Department of Computer Science

Self-Study Report

2022-2023

Table of Contents

Chapter 1: Introduction

Section 1.1. Lehman College Mission and Overview

> *Lehman College, an urban public institution and economic and cultural catalyst in the Bronx, is a national engine for social mobility and a vibrant center of discovery and creative work, providing a transformative educational experience while advancing equity, inclusion, and social justice.*
>
> *Lehman College Mission Statement*

Lehman College is one of the 25 colleges that collectively form The City University of New York (CUNY). CUNY's eleven senior colleges, seven community colleges, and seven honors, graduate, and professional schools make it the largest urban university in the United States of America. From 1931-1968 what is now the Lehman College campus was the Bronx campus of CUNY's Hunter College and known as Hunter-in-the-Bronx. Located on a 37-acre tree-lined campus, Lehman College was established as an independent senior college within the CUNY system on July 1st, 1968. It is named for Herbert H. Lehman, a former Governor and U.S. Senator from the state of New York, and a well-known internationalist. His values of dedicated public service, commitment to human rights, and support for immigrants are embodied in many of the College's programs and initiatives today.

In the years since its founding the campus grounds have evolved with the opening of Carman Hall (1971), the Leonard Lief Library (1980), the Lehman Center for the Performing Arts (1980), and the APEX gymnasium (1994). These structures were designed to complement the Art building (1959) and the original campus core which consisted of neo-Gothic towers and tree-lined walkways as it existed in 1946 when it served for six months as the interim headquarters for the newly formed United Nations. The College has made more recent additions with the Multimedia Center (2010), Science Hall (2012) and the Child Care Center (2013). The largest ongoing project is a $64 million, 50,000 square-foot Nursing Education, Research, and Practice Center, which will open in fall 2023.

Lehman College, the only comprehensive public four-year college in the Bronx, serves a predominantly low-income, underrepresented student population of approximately 14,500 drawing from 92 nationalities. The College provides an intellectual, economic, and cultural center for the Bronx and the surrounding region; it offers approximately 140 programs in undergraduate and graduate studies in the liberal arts and sciences as well as professional degrees. The college actively engages students in their academic, professional, and personal development, while embracing the diversity of its students, faculty and staff, and the surrounding community.

A top-ranked Hispanic-Serving Institution, Lehman's student population is a microcosm of the area it serves. Approximately 60% of undergraduates come from households with incomes below $30,000; 93% of undergraduates are from racialized communities; 88% of full-time freshmen receive financial aid (72% Pell); and 53% speak a language other than English at home. Approximately 60% of recent graduates were the first in their families to earn college degrees, approximately 60% were born outside the United States, and approximately 20% were parents while attending Lehman College. In 2020, the Brookings Institution recognized Lehman College as fourth in the nation among four-year universities for success in lifting low-income students into the middle class[1]. According to the American Council of Education, Lehman has the highest extended mobility rate, measured as the fraction of students who come from families in the bottom two income quintiles and end up in the top two quintiles as adults, among all Hispanic-serving institutions. Similarly, recent data from the National Bureau of Economic Research and the Equality of Opportunity Project ranked Lehman as having the third highest mobility rate in the nation among Hispanic-Serving Institutions. The Chronicle of Higher Education currently ranks Lehman College #3 for Highest Mobility Rate in the Nation[2]. U.S. News and World Report ranks Lehman College #1 in "Student Debt Load at Graduation - Least Debt" among regional universities[3] and Forbes recently set out to identify "the schools where grads owe the least relative to their likely mid-career income." Lehman was one of only four public colleges to rank in

---

[1] http://www1.cuny.edu/mu/forum/2017/01/24/lehman-college-ranked-no-4-in-mobility-rate-for-students-in-the-u-s/

[2] http://www1.cuny.edu/mu/forum/2017/01/24/lehman-college-ranked-no-4-in-mobility-rate-for-students-in-the-u-s/

[3] http://www1.cuny.edu/mu/forum/2017/01/24/lehman-college-ranked-no-4-in-mobility-rate-for-students-in-the-u-s/

the top 25 “low-debt/high-income schools.” Degree Choice ranks Lehman #1 among Hispanic Serving Institutions[4].

Lehman College offers 140 degree programs that are grounded in liberal arts and sciences, including 76 undergraduate majors and programs and 64 graduate degree programs, in addition to 33 graduate certificates. Lehman’s degree programs respond to the College’s core identity as a liberal arts institution committed to developing in students the critical skills and competencies they need for sustained, successful careers in the constantly changing 21st century workplace, as well as function as responsible global citizens. The faculty are committed to fostering students’ critical skills and competencies that they will need for successful careers in the constantly evolving twenty-first century workplace, and to participate in their communities, in their state and nation, and in the international community as responsible citizens. To this end, Lehman strives to balance its proactive investments in technology and multimedia resources – including equipment, support, and professional development and certification for web-enhanced instruction – with the highest quality in-person, hands-on traditional learning in our classrooms, laboratories, and studios.

Pivotal to providing these offerings are Lehman’s dedicated employees, consisting of nearly 1,000 faculty (full-time and part-time) and over 600 staff. The College’s nationally recognized faculty, including five distinguished professors, reside in the College’s six Schools – School of Arts and Humanities; School of Business; School of Continuing and Professional Studies; School of Education; School of Health Sciences, Human Services, and Nursing; and School of Natural and Social Sciences – where they serve as scholar-teachers, engaging with students in the latest disciplinary research and analysis. Lehman takes great pride in the diversity as well as productivity of its faculty.

[4] http://www1.cuny.edu/mu/forum/2017/01/24/lehman-college-ranked-no-4-in-mobility-rate-for-students-in-the-u-s/

The College's goals and institutional priorities are geared toward strengthening its contributions to the advancement of CUNY's stated mission, which reads in part: "...[T]he University will continue to maintain and expand its commitment to academic excellence and to the provision of equal access and opportunity ... The City University is of vital importance as a vehicle for the upward mobility of the disadvantaged in the City of New York." This commitment to the success of its students is evidenced in its offerings of advising, education, and opportunity for students across the enrollment spectrum – from Search for Education, Elevation, and Knowledge (SEEK) to Honors students, from first-year through transfer majors to graduate students.

Lehman College was experiencing a period of growth and development aligned with its mission and the mission of CUNY as a whole when the COVID-19 pandemic struck. While this has led to a small decrease in recent enrollment numbers, Lehman College is robust and growing in terms of access and opportunity. Excluding the recent decline, Lehman's enrollment has been expanding consistently for many years before and since 2017, when total student enrollment reached its largest point since 1976 when the City University of New York discontinued its policy of free tuition for New York City students. In particular, the number of STEM students at Lehman increased by 96% between 2013-2017, the highest rate of increase in the entire City University of New York system. These high rates of student engagement with science, technology, and mathematics have held steady since then.

The College's robust growth is also clear in its retention and graduation rates. Beginning in 2017, Lehman College's retention of first-time, full-time freshmen outpaced the CUNY senior college average by nearly 10%; the College's six-year graduation rate for first-time, full-time freshman has also increased and held steady, outpacing our fellow CUNY campuses significantly; Lehman's four-year graduation rate for transfer students from a CUNY Associate's Degree program also scored beyond the CUNY senior college average.

Lehman College opened the first Virtual and Augmented Reality Academy and Lab in the region with Eon Reality, and received a major grant from the Economic Development Corporation to provide additional workforce development as part of a three-year partnership with New York University.

Lehman College has in recent years received much more private financial support than in previous periods of its history. Recent investments (November 2018) include a $4.2 million award from the Robin Hood Foundation to support 250 students in the new Accelerate, Complete, Engage (ACE) completion program, starting in Fall 2019 with the goal of achieving 50% four-year graduation rates for freshmen and transfers with Associate degrees. In January 2021 Lehman College received a $30 million gift from philanthropist, author and activist MacKenzie Scott. This is the largest single gift in the college's history and one of the largest in the history of The City University of New York (CUNY).

Section 1.2. Vision, Values, and Goals of the College

In 2017, Lehman College launched the 90x30 challenge, a bold initiative designed to increase educational attainment in the Bronx by awarding 90,000 or more degrees and credentials from the beginning of the initiative through 2030. The 90x30 challenge is a call to action to help boost educational attainment rates in the Bronx, which ranks next to last of all 62 counties in New York State. Only 27.7% of Bronx residents have an associate's degree or higher.

In 2021, Lehman College published its new strategic plan, "Lehman 2025: Roadmap to the Future," which will guide the direction of the college as a catalytic institution of and national engine of inclusive excellence in the next five years and likely beyond. Developed through collaboration with campus and external stakeholders, the plan focuses on strengthening Lehman's long-term health and enhancing its visibility as one of the most innovative public colleges in the United States.

The new plan envisions Lehman College as a nationally recognized leader in educational attainment and the expansion of knowledge through innovative curriculum and pedagogy, original research and scholarship. and enhanced digital solutions. Furthermore, it asserts that the college will be a model of engaged citizenship and a leading force for a more just and sustainable world.

Beyond stating Lehman's vision, the 2025 roadmap also commits the college to five core values.

They are as follows.

Social Justice. Advocating for human rights, honoring differences, and working towards equality and equity for all.

Excellence. Pursuing innovative teaching, research, and scholarship to produce a diverse pipeline of leaders and change agents committed to novel and sustainable solutions and igniting new possibilities.

Ethics. Upholding the principles of academic freedom and demonstrating honesty, integrity, respect, and care in our interactions with others.

Diversity and Inclusion. Fostering a campus environment that respects and values diverse perspectives and identities and where all members of the Lehman community experience a remarkable sense of belonging.

Service. Empowering the local and international community through the increased engagement of faculty, staff, students, and alumni in outreach and partnerships contributing to individual achievement, economic development, and the common good.

Additionally, there are four key goals outlined and detailed in the new strategic plan. They are included immediately below.

Goal 1. Educate, Empower, and Engage Students to Participate in a Global Society and Enhance Career Advancement.

Lehman College combines excellence in curriculum and student experiences to prepare graduates who are educated, engaged, and empowered change agents and self-assured members of a global society. Our programs and curriculum will reflect the changing needs of the region and students to result in equitable outcomes. Lehman College will increase its impact through purposeful enrollment growth, innovation in curriculum and delivery, and an intentional approach to bridging theory and practice.

Goal 2. Enhance Faculty and Staff Success.

Lehman College is led by a diverse community of faculty and staff whose creativity and innovation intersect with teaching, coaching, mentoring, scholarship and academic support to achieve excellence, create knowledge, transform the lives of students and establish a more just society. The college encourages and supports faculty and staff at each stage in their professional career from the initial recruitment of outstanding individuals through the professional life cycle to achieve excellence. Faculty and staff are expected to engage students in the joys and transformative power of the pursuit of knowledge and are supported in their efforts to do so.

Goal 3. Sustain Growth, Vitality, and Institutional Effectiveness.

Lehman College will create an inclusive, welcoming campus climate, ensure sustainable financial and fundraising models, and increase grant submissions. We will facilitate a culture of continuous improvement, optimize campus facilities, technologies and related infrastructure, and develop integrated communications to keep our community informed and enhance our regional and national visibility.

Goal 4. Embrace the Spirit of Community Engagement.

Lehman College will increase educational attainment as a purpose-driven institution, serve as a catalyst for social justice and change, and contribute to the region’s vitality by expanding high quality educational programs to communities, creating strong talent pipelines that transform lives and ignite new possibilities for the region, and increasing collaborative opportunities and partnerships with local businesses, unions, cultural institutions, government agencies, and non-profit organizations.

The 2025 Roadmap calls on all campus stakeholders, including academic departments, faculty and staff, to participate in helping the college make progress towards fulfilling the stated vision, values and goals. As a vibrant community of teachers, learners, scholars, and activists, at the crossroads of a promise, dedicated to an ideal etched in stone more than 50 years ago by our founders, to work together to "enrich the human spirit and offer to as many as [could] realize their potential, the opportunity to be so enriched." That ideal remains our mission and the focus of Lehman College.

Section 1.3: The Department of Computer Science Overview

One can trace the origins of the Lehman College Department of Computer Science back to the early 1980s. In that period the Department of Mathematics at Lehman College began drawing up plans to offer computer science and computer information systems programs. The Department of Mathematics launched the first program in the fall 1983 semester, becoming the Department of Mathematics and Computer Science. Among the department faculty a core group of mathematicians dedicated themselves for many years to the computer science and computer information systems programs they developed in conjunction with the lone computer scientist on the faculty at that time, Julian Laderman (Emeritus since 2007). Four years later, in 1987, Victor Pan (Distinguished Professor) joined the department. He remains an active and vital member of the computer science faculty to this day and is the only remaining faculty member from this era.

In 1998 three computer scientists joined the fulltime faculty along with a mathematician who was hired to teach computer courses exclusively. Of these four only one, the department chair Brian Murphy, remains on the faculty today. In addition to Professors Pan and Murphy there are currently seven other regular fulltime computer science faculty: Itai Feigenbaum, Matthew Johnson, Sameh Fakhouri, Steven Fulakeza, Eva Sofianos, Liang Zhao, and Mingxian Zhong all of whom joined the faculty during the previous decade. All department faculty members are distinguished computer scientist and teachers. They along with dedicated staff are committed to a positive student experience. (See Chapter 5 for more details on faculty and staff.)

In fall 2017, Computer Science was extracted from the Department of Mathematics and Computer Science. The Department of Mathematics and Computer Science reverted to the Department of Mathematics and, a new department, the Department of Computer Science was formed. The Department of Computer Science at Lehman College is one of ten departments housed in the School of Natural and Social Sciences. Measured by declared majors it is the second largest department within the school behind the Department of Psychology. Along with the nine regular fulltime faculty the department has two temporary fulltime faculty, 16 adjunct faculty, and five staff members. (See Chapter 5 for more details.) Two of those staff members were hired this spring semester with funding from the new CUNY Inclusive Economy program. (See Section 7.4 for details.)

The department offers three undergraduate majors: a Bachelor of Science degree in Computer Science, a Bachelor of Science degree in Computer Information Systems, and a Bachelor of Arts degree in Computer Science the latter of which is limited to students taking a double major or a closely related minor. (At the time of the department split, a Bachelor of Science major in Computer Graphics and Imaging was handed off to the Department of Art.) The department also offers a minor in computer science and another in computer applications as well as a Certificate in Digital Technology and Electronics in conjunction with the Department of Physics and an online program to prepare K-12 teachers to teach computer science in the public schools. Each of our programs provides an enrolled undergraduate student some flexibility to tailor their study plans to best meet their interests and intended post-graduation plans. All of the programs seek to help prepare a student for further study and/or a career in industry. The department also offers two graduate programs one where students can earn a Master of Science degree in

Computer Science and another in conjunction with the School of Education provides teachers with the credentials to teach computer science in the New York public schools.

Section 1.4. Mission and Core Values of the Department

The mission of the Department of Computer Science is threefold:

1. to provide all participating students with a quality computer science or computer information systems education whereby they can learn and develop the quantitative and critical thinking skills they need to successfully pursue their individual course of study and become effective members of their greater communities,
2. to contribute to the world's body of knowledge through research, scholarship, and professional partnerships, and
3. to serve the neighboring community through advocacy for education, career preparation, the training of teachers, and partnerships in industry and with local schools/businesses.

The department is deeply committed to quality education, research, and service. We strive to balance our efforts to better serve our students, the computer science discipline, and the surrounding Bronx and New York City community. We contend that balancing these values supports Lehman's overall mission in promoting social mobility and providing a transformative educational experience.

With regards to education, the department works hard to meet Lehman students where they are preparation-wise when they enter the program. The department firmly believes in eliminating obstacles, bottlenecks, and roadblocks in an effort to better serve students in their academic paths toward graduation. At the same time, the department is committed to maintaining high academic standards and honoring articulation agreements across the CUNY system by setting a high bar. Therefore, we actively provide our students with the resources and skills that they need to ultimately demonstrate mastery in courses with uniform syllabi, clear learning objectives, and standardized assessments across multi-section courses. The more we demand and expect from our students, the more they can and will accomplish.

The department also expects a great deal from its faculty when it comes to original research and scholarship. A critical objective of the Department of Computer Science is to maintain a world-class research program – to continually contribute high-quality results to the world's body of knowledge through its research contributions in an effort to continually raise the department and school's research profile. All professors on the Department of Computer Science faculty are distinguished computer scientists that are well respected in their fields.

In an effort to serve the Bronx and the NYC community at large, the Department of Computer Science regularly participates in both Lehman and CUNY outreach program efforts. We have continued our involvement with the College Now program to welcome more local high school students on campus to take computer science and computer information systems courses. Additionally, we have developed a teacher training program to prepare teachers in the New York public schools to teach computer science. Further, one of our faculty members runs and many other participate in the Google Developer Group for the Bronx which holds its meetings and events on campus in Gillet Hall where the Department of Computer Science is housed.

Section 1.5. Department Leadership

<u>Chair</u>
Brian Murphy

<u>Deputy Chair (Not Elected - Selected by Chair)</u>
Eva Sofianos

<u>Personnel & Budget Committee</u>
Liang Zhao, Matthew Johnson, Victor Pan, Brian Murphy

<u>Senate Representative</u>
Steven Fulakeza

<u>CS Educational Policy Committee</u>
Eva Sofianos, Itai Feigenbaum, Liang Zhao, Brian Murphy

<u>CIS Educational Policy Committee</u>
Eva Sofianos, Mingxian Zhong, Steven Fulakeza, Brian Murphy

<u>Senators at Large</u> (Note: There are 51 faculty Senators in total. CS has 5 of them.)
Eva Sofianos, Liang Zhao, Sameh Fakhouri, Brian Murphy

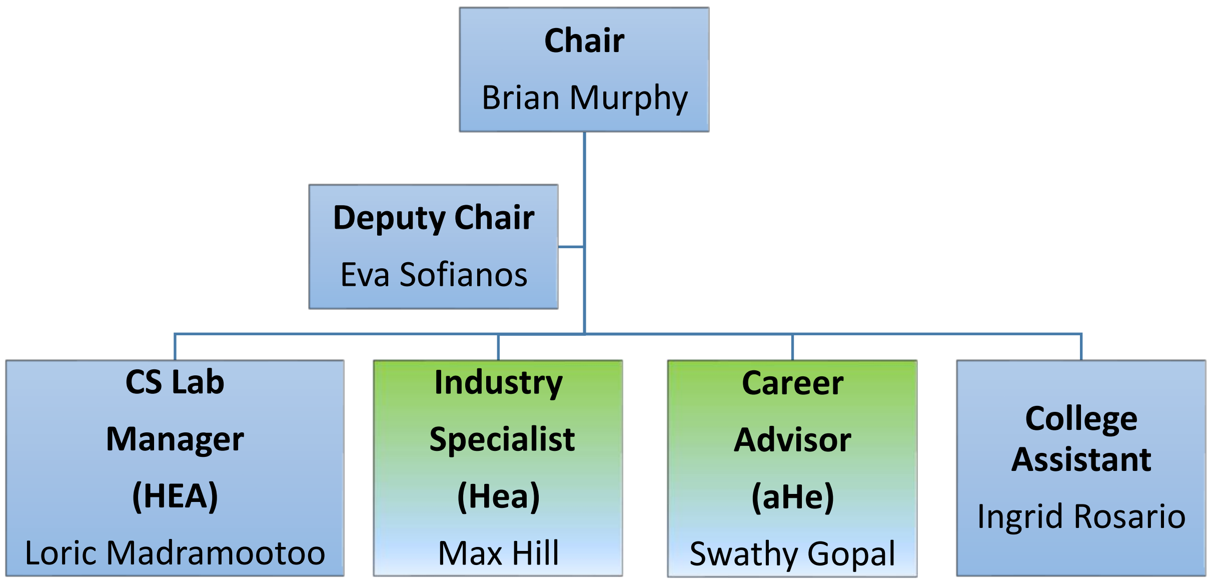
<u>Chairs since 2011 (last review)</u>
2011-2014 - Robert Schneider (Mathematics and Computer Science)
2014-Present - Brian Murphy (Mathematics and Computer Science until July 1st, 2017)

<u>Deputy Chairs since 2011 (last review)</u>
2010-2011 – Nancy Griffeth
2011-2012 – Anthony Cocchi
2012-2014 – Brian Murphy
2014-2017 – Joseph Fera (now chair of the Department of Mathematics)
2019-2019 – Sameh Fakhouri
2019-Present – Eva Sofianos

Note 1: Elected department positions have three-year terms that start July 1st.

Note 2: This May the department will hold elections for all positions.

Department of Computer Science Organization



Chapter 2: Assessment, Standardization, Online Tools, DWIF Studies

Section 2.1: Assessment

The Department of Computer Science has adopted a standard set of skills and knowledge that all Computer Science and Computer Information Systems majors will acquire along with strengthening their quantitative reasoning, critical thinking, and problem solving skills as they pass through the required program courses.

For the Computer Science majors these skills are as follows:

- Write programs; and be able to learn other languages easily.
- Use basic algorithms for sorting, searching, and organizing data.
- Use basic data structures such as lists, stacks, queues, tables, trees and graphs
- Know basic principles of computer architecture.
- Know basic principles of operating systems and how operating systems are organized.
- Know basic principles of database systems and how databases are organized.
- Know basic principles of computer networking using TCP/IP and write basic programs using sockets.
- Construct inductive proofs of programs and determine complexity of simple programs.

For the Computer Information major these skills are as follows:

- Understand the processes in an enterprise
- Understand how information systems support business processes to achieve the objectives of the enterprise, including both technical and organizational factors
- Understand the basic legal, professional, and ethical issues raised for individuals and society by various aspects of computing, and develop habits as a professional for addressing these issues

Each core department course has its own set of content-specific learning outcomes. These course outcomes are typically included in the class syllabus and state which of the general program learning objectives and goals they address. (See Appendix A for Syllabi.)

Section 2.2. Assessment and Course/Program Review

Formal assessment of Department of Computer Science courses and programs began in Spring, 2010. The assessment procedures and practices established then have remained in place (with only minor changes) for the past 12 years. A full-time faculty member serves as the department's assessment coordinator. This faculty member (currently Eva Sofianos) oversees all assessment efforts, works with the college's office of Assessment and Institutional Advancement on behalf of the Department of Computer Science, and receives 3 hours of reassigned time yearly from the institution.

Different courses are assessed each academic year on a yearly rotation. Assessment of introductory courses include a reevaluation of uniform syllabi, which provide a careful scheduling of course content chapter-by-chapter. Assessment of more advanced courses are done on an individual basis by the faculty instructing the course with support and guidance from the Computer Science Assessment Coordinator. Department programs (i.e., majors and minors) are also reviewed and assessed by the Computer Science Assessment Coordinator. Recommendations and changes to courses and/or programs are evaluated by two department subcommittee, the Computer Science Educational Policy Committee (CS EPC) and the Computer Information Systems committee (CIS EPC). The EPCs are each made up of 3 full-time faculty elected by department members to serve 3-year terms as well as the department chair.

For advanced courses assessment is mainly done using data collected on department final exams. The data consists of student performance on final exam questions, each of which is directly aligned with course learning outcomes. Instructors report how many of their students successfully answered each final exam question; this data is collected by the assessment coordinator and used to create their report/recommendations. Overall, the process moves from data collection, followed by analysis, and ultimately to recommendations to the EPCs for changes, if necessary. A critical step in this process is "closing the loop" – acting on the data collected in an effort to improve the course and advance student success. (See Appendix B for Program Assessment Reports.)

Section 2.3: Ancient History with Languages

In the 1980's Pascal was used as the introductory language and used across the curriculum, except in a few cases where languages such as C and Assembly as well as LISP were appropriate. Pascal was replaced by C++ in 1994, and C++ with Java in 1997. Java remained in place until 2011 when Python took its place, but only for the first programming course in the major. Other courses continued to rely on Java.

Section 2.4: The Past Decade

Over the past 10 years, the Department of Computer Science has gone through multiple phases in the setup and design of the introductory programming sequence (CMP 167 Programming Methods 1, CMP 168 Programming Methods 2, and CMP 338 Data Structures & Algorithms I). Each of these courses had multiple sections. At the time there was no coordination or joint planning between the various sections in each course. Each course had its own syllabus outlining the material that must be covered in each course and it was left to the instructors' discretion to teach the material and assess the students' performance. The department observed that even though there was a requirement of a B- in order to move from Programming Methods 1 to Programming Methods 2 there seemed to be a lack of grade integrity or consistency when evaluating the preparedness of incoming students in Programming Methods 2. This led the department to develop a strategy for unifying the curriculum across the sections of Programming Methods 1.

Section 2.5: Python and a Unified Curriculum for Programming 1

A unified curriculum was created that included a common syllabus, common assignments, common exams, as well as common assessment rubrics for all Programming Methods 1 sections. This led to better grade integrity and consistency across the sections of this course. In addition to all these changes, the department also changed the programming language used for instruction in Programming Methods 1, from Java to Python. The reasoning behind this decision derived from the ease of use of Python as an introductory language, coupled with its user-friendly teaching tools. Java remained the main language of instruction in the subsequent courses. The unified Python curriculum for Programming Methods 1 was in effect for several semesters and it produced the expected consistency across all the sections. However, we began observing that students were struggling to apply basic programming concepts in Programming Methods 2 when they moved to Java. To accommodate the students and enable them to succeed in Programming Methods 2, instructors were dedicating class time to re-teach students the basic programming concepts and how to implement them using the stricter rules of Java. The department decided to transition back to using Java as the language in Programming Methods 1 so that students would be familiar with Java upon entering Programming Methods 2.

Section 2.6: Return to Java

The return to Java in Programming Methods 1 was not a graceful one. Although there was a unified syllabus and learning objectives, common assignments and common assessment criteria were abandoned. Some instructors decided to remain teaching the course using Python while others converted to Java. After the Computer Science Department split from the Mathematics Department, all sections of Programming Methods 1 transitioned to Java as the language of choice. This phase reintroduced the inconsistencies across sections and lack of grade integrity. However, students entering Programming Methods 2 were better versed in Java rules as well as Java syntax than when Python had been the language used in Programming Methods 1. These observations led to the decision that Java should be used across all sections of Programming

Methods 1, but that unification of assignments and assessments needed to be put in place once again. In an attempt to support student success throughout the major and prepare them for subsequent courses with unified assessment and curriculum, the department decided to not only unify Programming Methods 1 but Programming Methods 2 as well.

Section 2.7: Restructuring and Unification

Starting in the spring semester of 2019 the department restructured and unified the course curriculum for the first two required courses in the Computer Science Major's learning sequence, Programming Methods 1 and Programming Methods 2 . These two courses had been selected for restructuring to assure students would build a solid foundation of knowledge that would allow them to succeed in the major. The main objective of the curriculum restructuring project was to reduce the percentage of Drop-out, Withdrawal, Incomplete, and Failure (DWIF) in the initial courses which would also result in a reduction of DWIF in subsequent courses of the Computer Science Major's programming sequence. The effort of curriculum restructuring, development, and unification, as well as content management and assurance of consistency among the multiple sections is led by Computer Science Lecturers, Sameh Fakhouri and Eva Sofianos. Course instructors are expected to follow the curriculum, as well as use the customized online textbook and data tracking tools they are provided for each of their course sections.

The unification effort for Programming Methods 1 and 2 involved choosing an online textbook that allows instructors to track student progress with regards to both reading the book and completing the programming assignments. While students are reading the book they have to complete Participation and Challenge (P&C) activities. This allows instructors to track the progress of students throughout the semester. In addition, all programming assignments are integrated into the textbook, allowing students to submit their code for testing and verification. Students are given instant feedback on their submission and are given the opportunity to repeatedly re-submit their code until they have earned full credit.

We continue to use the unified curriculum, customized online book and single syllabus for each course across all sections of CMP167 and CMP168. We continue to maintain grade integrity by hosting meetings where all instructors are present and review the scores prior to official grade submission.

The material covered in CMP 167 and CMP 168 matches across all sections. The same rubrics, exams, assignments, projects and problem sets are used in assessing student learning.

Section 2.8: Interactive Textbook and Tracking

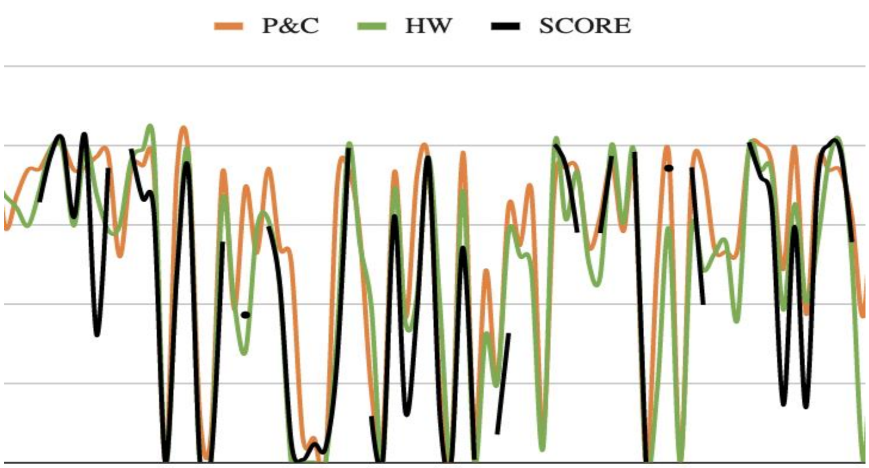
Through the interactive online book we have been able to track student reading and participation (P&C) as well as homework submission to determine how it affects their overall success in a single course. The calculated course grades for CMP 167 and CMP 168 use the same formula to calculate the weighted average: P&C 10%, HW 25%, Projects 15%, Midterm Exam 25%, Final Exam 25%. Upon reviewing the data there is a very strong positive correlation between P&C completion (interactive reading), successful homework correctness and completion, as well as overall course grade. Students who put in the effort by reading the assigned material are able to complete the homework assignments and proceed to correctly completing the projects as well as performing successfully on the exams, thus leading to overall success in the course.

Tracking of individual student success along with continued effort and commitment from one course to the next has not been inspected at this point in time. It is expected that students who earn a credit grade of “B-” or better will be prepared to move on to the next course and perform well in the second course given that their effort persists. Data is being collected so that we can track individual student success in the programming course sequence given their effort persists.

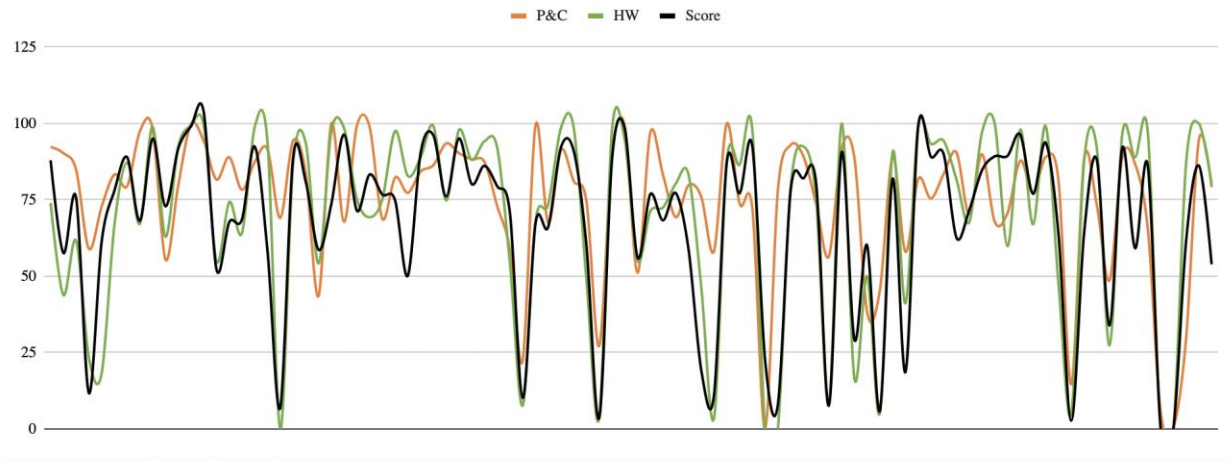
Section 2.9: Data Collection

We are still in the process of collecting data in order to be able to track students along the programming course sequence. That being said, it is evident that properly focused effort on assigned materials leads to student success in the course they are completing the work for. Below are the charts depicting the correlation between P&C, HW, and Course Grade for Programming Methods 1 and 2 during the Fall 2021 and Spring 2022 Semesters

CMP 167 Fall 2021

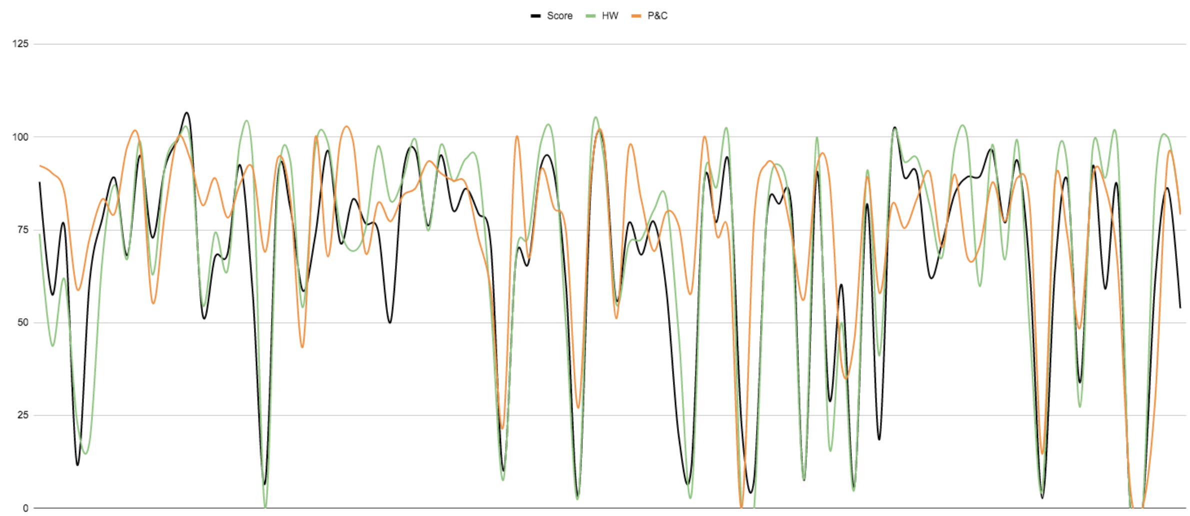


CMP 168 Fall 2021



CMP 167 Spring 2022

CMP 168 Spring 2022



We have unified the 3 programming courses and have changed the delivery structure for the first 2 from independent lecture-lab sections to one unified lecture with multiple coordinated lab sections. We continue to collect data surrounding student participation, assignment completion, exam scores, and overall course success for all 3 courses.

We will continue to use the shared unified content across course sections for each of the courses.

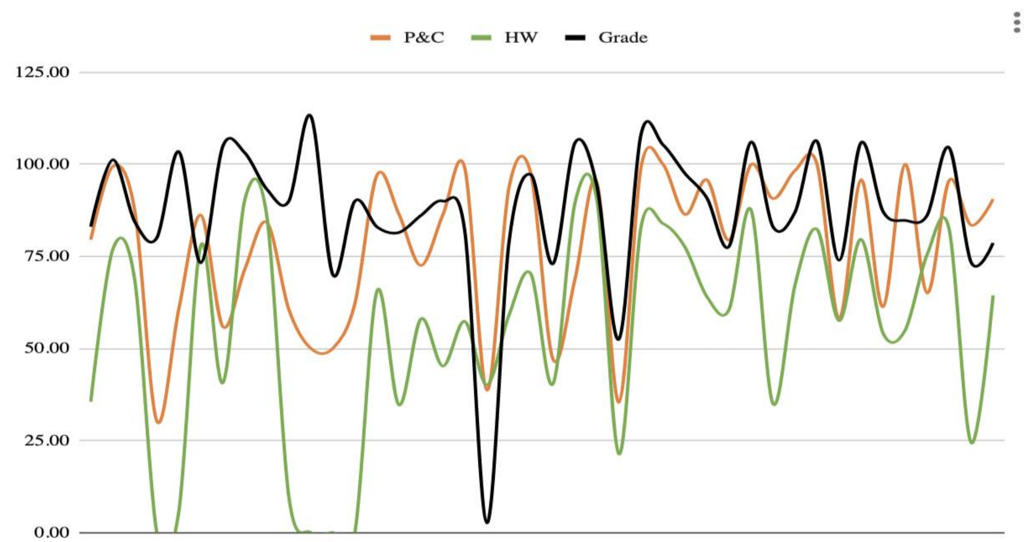
Section 2.10: Data Structures and Algorithms 1

With regard to CMP 338, Data Structures and Algorithms 1, an online interactive book has been adopted and customized so that student P&C as well as programming assignments can be tracked in an identical manner to Programming Methods 1 and 2. The same assignments, and problem sets in the book are being used to assess student learning across all sections of the course and track student success. The same strong positive correlation between P&C completion (interactive reading), successful homework completion, and overall course grade has been observed. However, the department chair who often teaches CMP 338 has used the online textbook once, was not satisfied, and reverted to his old textbook choice.

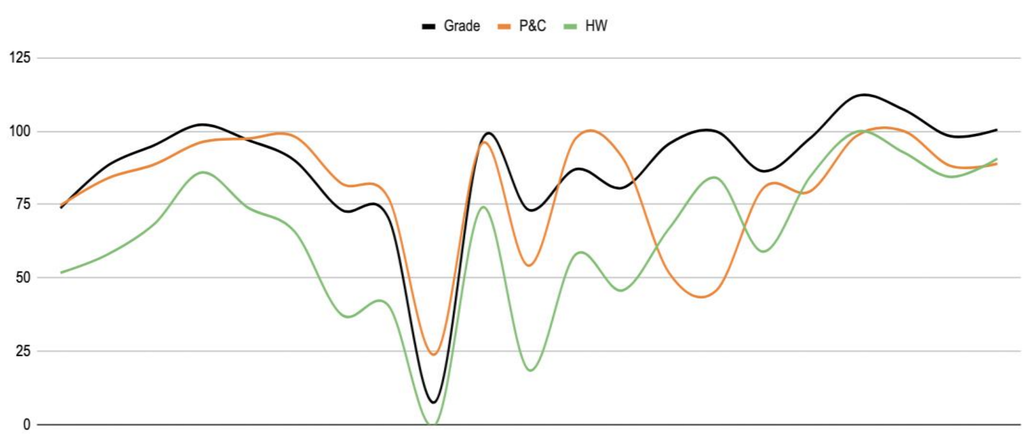
Section 2.11: More Correlating Squiggly Lines

Following are charts depicting the correlation between P&C, HW, and Course Grade for Data Structures & Algorithms 1 during the Fall 2021 and Spring 2022 semesters.

CMP 338 Fall 2021



CMP 338 Spring 2022



We plan to continue tracking student participation and success in each course as well as their progression through the programming sequence. We also plan to use the data to observe whether there are particular topics in the course sequence that may benefit from content adjustment if the data analysis shows students are struggling despite their effort.

Our course unification has coincided with the Covid-19 pandemic which resulted in classes shifting to online or hybrid modalities. We have found that in this situation student engagement has suffered greatly. Furthermore, the admittance criteria for students entering the college has changed due to the pandemic. One of the changes to the admission criteria that affects the Computer Science Program is the removal of Math Competency Scores. This directly affects student preparedness and ability to succeed in courses in the introductory programming sequence where problem solving skills acquired, practiced, and reliant on mathematics are necessary. Until now, we have not observed a reduction in the DWIF rate in Programming Methods 1 and 2. We are however optimistic that the course redesign and unification will begin to bear fruit as we switch back to in-person learning and restoration of math requirements in the admission criteria.

Chapter 3: Curriculum

Section 3.1: New Curriculum

Over the last few years, the Department of Computer Science's CS EPC and CIS EPC have undertaken major overhauls of the two BS programs housed in the department. The redesigns add various tracks within the Computer Information Systems major including programming and web development and they expand the Computer Science BS from 60 to 80 credits to ensure that Lehman College students are properly prepared for what comes after graduation. These changes are held up in the approval process. The documents submitted to effect these changes are provided in Appendix H.

The next few sections present the programs as they are currently constituted.

Section 3.2: BS in Computer Science (57-60 Credits)

To earn a BS degree in Computer Science, students must complete all of the following courses:

MAT 175 - Calculus I
MAT 176 - Calculus II
MAT 313 - Elements of Linear Algebra
CMP 167 - Programming Methods I
CMP 168 - Programming Methods II
CMP 232 - Elementary Discrete Structures & Applications to Computer Science
CMP 334 - Computer Organization
CMP 338 - Data Structures and Algorithms I
CMP 405 - Introduction to Networks
CMP 420 - Database Systems
CMP 426 - Operating Systems

Complete at least 4 of the following:
Advanced CMP courses

Complete at least 1 of the following:
Advanced MAT course

Additional Comments:
Four Advanced CMP courses: 300- or 400-level CMP courses are required. MAT 226 or PHY 305 can be substituted for one of these courses.

One Advanced MAT course: One 300- or 400-level MAT course is required, not including MAT 300, MAT 301, or MAT 348. CMP 332, CMP 416 can be used for this course.

All students, particularly those considering graduate work, are advised to take more upper-level Computer Science courses, particularly CMP 339 and CMP 416.

It is suggested that students follow the prescription that follows to complete their degree in four years:

FRESHMAN

FALL

ENG 111 3 CR
English Composition I
Required Core – Communication

MAT 172 4 CR
Pre-calculus
Required Core-Quantitative skills

LCR 3 CR
Flexible Core – Scientific World

CMP 167 3 CR
Programming Methods I

Elective 3 CR
LEH 100 (recommended)
The Liberal Arts - Freshman Seminar

SPRING

ENG 121 3 CR
English Composition II
Required Core – Communication

CMP 168 4 CR
Programming Methods II

CMP 232 4 CR
Elementary Discrete Structures & Applications to Computer Science

MAT 175-LCR 4 CR
Calculus I
Flexible Core – Any area [1]

MAT 155 1 CR
Calculus I Lab

16 FALL CREDITS + 16 SPRING CREDITS = 32 CREDITS

SOPHOMORE

FALL

LCR 3 CR
Foreign Language I
Lehman College Option

CMP 334 4 CR
Computer Organization

CMP 338 4 CR
Data Structures and Algorithms I

MAT 176 4 CR
Calculus II

MAT 156 1 CR
Calculus II Lab

SPRING

LCR 3 CR
Foreign Language II
Lehman College Option

CMP 405 3 CR
Introduction to Networks

CMP 426 4 CR
Operating Systems

MAT 313 4 CR
Elements of Linear Algebra

32 PRIOR CREDITS + 16 FALL CREDITS + 14 SPRING CREDITS = 62 CREDITS

JUNIOR

FALL

LCR 3 CR
LEH 352, 353, 354, or 355 [2]
Lehman College Option

LCR 3 CR
Flexible Core – World Cultures and Global Issues

LCR 3 CR
Flexible Core - US Experience in Its Diversity

CMP 3## or **4##** [3] 4 CR
Major Elective

Minor or Certificate Course I [5] 3 CR

SPRING

LCR 3 CR
LEH 352, 353, 354, or 355 [2]
Lehman College Option

LCR 3 CR
Required Core – Life and Physical Science

MAT 3## or 4## [4] 4 CR
Advanced Math Course

CMP 3## or **4##** [3] 4 CR
Major Elective

Minor or Certificate Course II [5] 3 CR

62 PRIOR CREDITS + 16 FALL CREDITS + 17 SPRING CREDITS = 95 CREDITS

SENIOR

FALL

CMP 3## or **4##** [3] 4 CR
Major Elective

CMP 420 4 CR
Database Systems

Elective 1-2 CR

Minor or Certificate Course III [5] 3 CR

SPRING

CMP 3## or **4##** [3] 4 CR
Major Elective

LCR 3 CR
Flexible Core – Creative Expression

LCR 3 C
Flexible Core – Individual and Society

Minor or Certificate Course IV [5] 3 CR

5 PRIOR CREDI TS + 12 FALL CREDITS + 13 SPRING CREDITS = 120 CREDITS

Section 3.3: BA in Computer Science (43-46 Credits)

To earn a BA degree in Computer Science, students must complete all of the following courses:

MAT 175 - Calculus I
MAT 176 - Calculus II
MAT 313 - Elements of Linear Algebra
CMP 167 - Programming Methods I
CMP 168 - Programming Methods II
CMP 232 - Elementary Discrete Structures & Applications to Computer Science
CMP 334 - Computer Organization
CMP 338 - Data Structures and Algorithms I
CMP 339 - Programming Languages OR CMP 426 - Operating Systems
CMP 405 - Introduction to Networks OR CMP 420 - Database Systems

Complete at least 2 of the following:
Two advanced CMP electives: At 300- or 400-level. MAT 226 can be used as one of these electives.

Notes:

All students, particularly those considering graduate work, are advised to take more upper-level Computer Science courses. (The list above is only the minimum required for graduation.)

It is suggested that students follow the prescription that follows to complete their degree in four years:

FRESHMAN

FALL

ENG 111 — 3 CR
English Composition I
Required Core – Communication

LCR — 3 CR
Flexible Core – World Cultures and Global Issues

MAT 104 or MAT 172 or Elective — 3-4 CR
(depending on your math placement)

CMP 167 — 3 CR
Programming Methods I

Elective — 3 CR
LEH 100 (recommended)
The Liberal Arts - Freshman Seminar

SPRING

ENG 121 — 3 CR
English Composition II
Required Core – Communication

MAT 172 -LCR or Elective — 4 CR
Pre-calculus
Flexible Core-Quantitative skills

LCR — 3 CR
Required Core – Scientific World

LCR — 3 CR
Flexible Core - US Experience in Its Diversity

CMP 168 — 3 CR
Programming Methods II

15 FALL CREDITS + 16 SPRING CREDITS = 31 CREDITS

SOPHOMORE

FALL

LCR — 3 CR
Foreign Language I
Lehman College Option

LCR — 3 CR
Flexible Core – Individual and Society

CMP 232 — 4 CR
Elementary Discrete Structures & Applications to Computer Science

MAT 175 -LCR — 4 CR
Calculus I
Flexible Core – Any area [1]

MAT 155 — 1 CR
Calculus Lab I

SPRING

LCR — 3 CR
Foreign Language II
Lehman College Option

CMP 334 — 4 CR
Computer Organization

CMP 338 — 4 CR
Data Structures and Algorithms I

MAT 176 — 4 CR
Calculus II

MAT 156 — 1 CR
Calculus Lab II

31 PRIOR CREDITS + 15 FALL CREDITS + 16 SPRING CREDITS = 62 CREDITS

JUNIOR

FALL

LCR 3 CR
LEH 352, 353, 354, or 355 [2]
Lehman College Option

CMP 339 or **CMP 426** 4 CR
Programming Languages or Operating Systems

MAT 313 4 CR
Elements of Linear Algebra

Minor or Certificate Course I [4] 3 CR

SPRING

LCR 3 CR
LEH 352, 353, 354, or 355 [2]
Lehman College Option

LCR 3 CR
Required Core-Life and Physical Science

CMP 405 or **CMP 420** 3-4 CR
Introduction to Networks or Database Systems

Elective 3 CR

Minor or Certificate Course II [4] 3 CR

62 PRIOR CREDITS + 14 FALL CREDITS + 15 SPRING CREDITS = 91 CREDITS

SENIOR

FALL

CMP 3## or **CMP 4##** [3] 4 CR
Major Elective

Minor or Certificate Course III [4] 3 CR

Minor or Certificate Course IV [4] 3 CR

Elective 3 CR

Elective 3 CR

SPRING

CMP 3## or **CMP 4##** [3] 4 CR
Major Elective

LCR 3 CR
Flexible Core – Creative Expression

Elective 3 CR

Elective 3 CR

1 PRIOR CRED ITS + 16 FALL CREDITS + 13 SPRING CREDITS = 120 CREDITS

Section 3.4: BS in Computer Information Systems (55-58 Credits)

To earn a BS degree in Computer Information systems, students must complete all of the following courses:

Earn at least 19 credits from the following:
CIS 166 - Computer Programming for Information Processing I
CIS 211 - Computer Information Systems
CIS 212 - Microcomputer Architecture
CIS 244 - Introduction to Database Management
CIS 331 - Network Introduction
CIS 344 - Database Design and Programming

Earn at least 6 credits from the following:
CGI 221 - Applied Imaging and Applications to the World Wide Web I
CGI 321 - Computer Modeling and Design I
CGI 421 - Computer Animation I
CMP 168 - Programming Methods II
One of the courses must be a 300- (or 400-) level CIS course

Mathematics
Earn at least 15 credits from the following:
MAT 132 - Introduction to Statistics
MAT 174 - Elements of Calculus
MAT 301 - Applied Statistics and Computer Analysis for Social Scientists
MAT 348 - Mathematical Methods for Management
MAT 174: Students considering graduate work should take MAT 175, MAT 176 instead of MAT 174.

Economics
Earn at least 9 credits from the following:
ECO 166 - Introduction of Macroeconomics
ECO 167 - Introduction to Microeconomics
ACC 185 - Introduction to Accounting for Non-Accounting Majors OR ACC 171 - Principles of Accounting I

Further Electives
Complete at least 2 of the following:
Complete at least 1 courses in the following Course Sets:
Computer Information Systems 200 Level
Complete ANY of the following Courses:
PHI 221 - Ethical Issues in Computing and Technology
POL 299 - Law, Computers and the Internet: The Politics of Information Technology

Additional Comments:
One 200-level CIS course: Or higher, for 3 credits.

At least one of PHI 221 and POL 299 must be chosen.

It is suggested that students follow the prescription that follows to complete their degree in four years:

FRESHMAN

FALL

ENG 111 3 CR
English Composition I
Required Core – Communication

LCR 3 CR
Flexible Core - World Cultures and Global Issues

MAT 104 (depending on your math placement) or 3 CR
Elective

ECO 166 3 CR
Introduction to Macroeconomics
Flexible Core – Individual and Society

Elective 3 CR
LEH 100 (recommended)
The Liberal Arts - Freshman Seminar

SPRING

ENG 121 3 CR
English Composition II
Required Core – Communication

LCR 3 CR
Required Core – Life and Physical Science

CIS 166 3 CR
Computer Programming for Information Processing

CIS 211 4 CR
Computer Information Systems

MAT 132- LCR 4 CR
Introduction to Statistics
Required Core – Quantitative Skills

15 FALL CREDITS + 17 SPRING CREDITS = 32 CREDITS

SOPHOMORE

FALL

LCR 3 CR
Foreign Language I
College Option - Foreign Language

LCR 3 CR
Flexible Core – Scientific World

LCR 3 CR
Flexible Core – Creative Expression

CIS 212 3 CR
Microcomputer Architecture

ECO 167 - LCR 3 CR
Introduction to Microeconomics
Flexible Core – Any area [1]

SPRING

LCR 3 CR
Foreign Language II
College Option - Foreign Language

LCR 3 CR
Flexible Core – US Experience and Diversity

CMP 167 3 CR
Programming Methods I

CIS 244 3 CR
Introduction to Database Management

MAT 171 4 CR
Problem Solving for Management, Economics, and Life Sciences

2 PRIOR CREDITS +

15 FALL CREDITS + 16 SPRING CREDITS = 63 CREDITS

JUNIOR

FALL

LCR **LEH 352, 353, 354, or 355** [2] *Lehman College Option*	3 CR
CIS 331 Network Introduction	3 CR
CIS 344 Database Design and Programming	3 CR
MAT 174 Elements of Calculus	4 CR
MAT 301 Applied Statistics and Computer Analysis for Social Scientists	4 CR

SPRING

LCR **LEH 352, 353, 354, or 355** [2] *Lehman College Option*	3 CR
ACC 171 or ACC 185 Principles of Accounting I *or* Introduction to Accounting for Non-Accounting Majors	3 CR
CIS 2##, 3##, 4## or CGI 221, 321, 421 or CMP 168 [3] Major Electives	3 CR
MAT 348 Mathematical Methods for Management	4 CR
Elective	3 CR

63 PRIOR CREDITS + 16 FALL CREDITS + 16 SPRING CREDITS = 95 CREDITS

SENIOR

FALL

CIS 2## or PHI 221 or POL 229 [4] Further Major Electives	3 CR
CIS 2##, 3##, 4## or CGI 221, 321, 421 or CMP 168 [3] Major Electives	3 CR
Elective	3 CR
Minor or Certificate Course I [5]	3 CR
Minor or Certificate Course II [5]	3 CR

SPRING

CIS 2## or PHI 221 or POL 229 [4] Further Major Electives	3 CR
Elective	3 CR
Minor or Certificate Course III [5]	3 CR
Minor or Certificate Course IV [5]	3 CR

96 PRIOR CREDITS + 13 FALL CREDITS + 12 SPRING CREDITS = 120 CREDITS

Section 3.5: Minor in Computer Science (12 Credits CMP courses only)

To earn a Minor in Computer Science, students must:

Earn at least 12 credits at the 200-level or above, with at least 6 at 300-level or above.

Complete all of the following courses:

CMP 334 - Computer Organization
CMP 338 - Data Structures and Algorithms I

Students majoring in Mathematics, who elect to have a minor in Computer Science
Complete at least 1 of the following Courses:
CMP 332 - Numerical Methods
CMP 338 - Data Structures and Algorithms I

Earn a minimum grade of C- in each course.

Section 3.6: Minor in Computer Applications (16 Credits)

To earn a Minor in Computer Science, students must complete ALL of the following:

CIS 211 - Computer Information Systems
Two CIS courses at the 200-level
One CIS course at the 300 level

Earn a minimum grade of C- in each course.

Section 3.7: Online Computer Science for Teachers

In addition to courses from the School of Education students must complete all of the following courses:

CMP 566 – Computer Thinking for Educators
CMP 567 – Programming Methods I for Educators
CMP 568 – Programming Methods II for Educators
CMP 569 – Data Structures and Algorithms for Educators

Section 3.8: MS in Computer Science (36 Credits)

To earn a MS degree in Computer Science, students must complete all of the following:

CMP 692 - Programming Languages
CMP 697 - Operating Systems
CMP 761 - Analysis of Algorithms
Six additional courses must be chosen from among all CMP courses numbered 683 and above.

Courses are chosen with permission of the Graduate Adviser

A master's thesis or a written comprehensive examination. The thesis option is subject to approval of the Graduate Adviser.

Section 3.9: Certificate in Digital Technology and Electronics

To earn a Certificate in Digital Technology and Electronics, students must complete all of the following courses:

At least 6 of the following Courses:
PHY 165 - Applied Physics
PHY 305 - Analog Circuits
PHY 315 - Digital Circuits and Systems
CMP 167 - Programming Methods I
CMP 168 - Programming Methods II
CMP 334 - Computer Organization
PHY 165: Students may take PHY 167 or PHY 169 in place of PHY 165.

At least 1 of the following Courses:
CMP 487 - Internship in Computer Science
OR PHY 487 - Internship in Physics
OR PHY 489 - Honors Course

Section 3.10: Computer Science Course Descriptions (Some Learning Objectives)

CMP 108 Programming for Non-Computer Science Majors

Implementation of basic programming constructs using robots, designing of simple video games, and creation of elementary web pages.

Learning objectives / syllabus unavailable

CMP 128 Programming through Web Development

Creation of websites using HTML, CSS, and Javascript.

Independently design and develop simple web pages Define and use variables Define and use functions Demonstrate the use of parameters and information passing in programs Understand and use arrays Understand and use control structures such as decision branching & iteration

CMP 157 Programming 1 Lab

Application of concepts learned in CMP 167 to develop programming solutions to problems as lab assignments. Code will be written in an Integrated Developer Environment.

Independently design, create, debug simple applications Define and use variables of various data types Define and use methods Demonstrate the use of parameters and information passing in programs Use existing libraries and their methods Manipulate strings Manipulate 1 dimensional arrays Use control structures such as decision branching & iteration Create classes to represent objects

CMP 158 Programming 2 Lab

Application of concepts learned in CMP 168 to develop programming solutions to problems as lab assignments. Code will be written in an Integrated Developer Environment.

Independently design, create, debug applications Create GUI (graphical user interface) for desktop applications Perform decision branching using if-else statements, switch cases Perform iteration using loops for, while, do-while Use recursion to solve problems Manipulate arrays (1 dimensional & 2 dimensional) Manipulate strings Use streams and perform file i/o Demonstrate use of exception handling Use popular sorting algorithms (bubble, selection, insertion, merge) Use popular searching algorithms (sequential, binary)

CMP 167 Programming Methods I

Structured computer programming using a modern high-level programming language. Includes console I/O, data types, variables, control structures, including iteration, arrays, function definitions and calls, parameter passing, functional decomposition, and an introduction to objects. Debugging techniques.

Independently design, create, debug simple applications Define and use variables of various data types Define and use methods Demonstrate the use of parameters and information passing in programs Use existing libraries and their methods Manipulate strings Manipulate 1 dimensional arrays Use control structures such as decision branching & iteration Create classes to represent objects

CMP 168 Programming Methods II

Continuation of parameter passing with a focus on devising function definitions and tracing recursive calls. Sorting and searching algorithms and a comparison of their performance. GUI programming. Threads, Exceptions and Exception Handling. Object Oriented Programming techniques.

Demonstrate OOP through proper use of encapsulation, polymorphism and inheritance. Independently design, create, debug applications GUI (graphical user interface) for desktop applications Perform decision branching using if-else statements, switch cases Perform iteration using loops for, while, do-while Use recursion to solve problems Manipulate arrays (1 dimensional & 2 dimensional) Manipulate strings Use streams and perform file i/o Demonstrate use of exception handling Use popular sorting algorithms (bubble, selection, insertion, merge) Use popular searching algorithms (sequential, binary)

CMP 232 Elementary Discrete Structures & Applications to Computer Science

Sets, relations, and functions; propositional calculus, Boolean algebras, and combinatorial circuits, counting methods; proof techniques; analysis of algorithms; graphs and trees, puzzles; finite machines, sequential circuits, and recognizers.

Use truth table to evaluate logic statements including compound propositions, predicates and nested quantifiers, and establish propositional equivalences. Understand basic concepts in set, function, sequences, big-O notations, elementary number theory and graph theory, and apply them to solve simple problems. Understand proofs and use induction or other basic proof techniques to write proofs. Solve counting problems by applying elementary counting techniques including product and sum rules, permutations, combinations and the pigeon-hole principle.

CMP 266 Cmp Prog Inf Proc II

Cmp Prog Inf Proc II

Learning objectives / syllabus unavailable

CMP 269 Programming Methods III

An In-depth exploration of Object Oriented programming with emphasis on inheritance, interfaces, multi-threading, I/O, GUI, recursion and unit testing. Programming projects to be implemented in different languages.

Demonstrate OOP through proper use of encapsulation, polymorphism and multi-level inheritance hierarchies. Implement multiple Interfaces as well as inheriting from classes that have implemented Interfaces Implement generics and their use in object declarations Demonstrate understanding and usage of Collections in each of the assigned programming languages Demonstrate understanding and usage of Enumerations Synchronize Threads using Semaphores in applications Design and Develop fully functional applications in the assigned programming languages Perform Serialization and Streaming of Objects Use File I/O for both text and object storage/retrieval in applications Development of GUI Make use of Debugging Techniques and Tools Make use of Testing Techniques and Tools (Unit testing and Integration testing)

CMP 270 Computer Programming in Visual Basic

Intensive introduction to Visual Basic, for students who have successfully completed a two-semester sequence in another programming language.

Learning objectives / syllabus unavailable

CMP 273 Computer Programming in Fortran

Intensive introduction to Fortran for students who have successfully completed a two-semester sequence in another programming language.

Learning objectives / syllabus unavailable

CMP 274 Computer Programming in C++

Intensive introduction to C++, for students who have successfully completed a two-semester sequence in another programming language.

Learning objectives / syllabus unavailable

CMP 277 Computer Programming in Java

Intensive introduction to Java, for students who have successfully completed a two-semester sequence in another programming language.

Learning objectives / syllabus unavailable

CMP 279 Topics in Computer Programming

(May be repeated, with change of language and Departmental permission.) Intensive introduction to a programming language, for students who have successfully completed a two-semester sequence in another programming language.

Learning objectives / syllabus unavailable

CMP 320 Quantum Computer Science

An elementary introduction to quantum information science and quantum computing for majors in computer science, physics, and mathematics. The linear algebra of quantum mechanics, bits versus qubits, quantum cryptography, quantum teleportation, quantum gates and quantum computing, the Grover search algorithm.

Learning objectives / syllabus unavailable

CMP 332 Numerical Methods

Computer methods of approximate calculation. Topics chosen from polynomial approximation and interpolation, orthogonal polynomials, finite difference methods, splines and least square approximations; numerical differentiation and integration; numerical solutions of equations by iterative methods and matrix inversion; eigenvalue computations; sparse matrix methods.

Learning objectives / syllabus unavailable

CMP 333 Data Management and Analysis

Introduction of Data handling tools and techniques, extracting and presenting information about data, and computational thinking processes.

Import data into Python from files in different formats. Scrape data from websites and databases. Process text, images, and date and times. Create data aggregation using pivot tables. Apply formulas to extract statistical information. Visualize the dataset and make inferences.

CMP 334 Computer Organization

Introduction to digital logic-expressions, gates, flip-flops, adders. busses, multiplexers Introduction to assembly language and assembly level organization - data representation, instruction formats, addressing modes, interrupts. Memory systems - caches (mapping and management policies) and memory hierarchies, latency and bandwidth, virtual memory (pagetables, TLB). Input/Output- busses, channels and DMA. Performance considerations- pipelining, RISC architecture, branch prediction, introduction to instruction level parallelism.

Understand unsigned and signed fixed-width binary arithmetic, including overflow conditions and comparisons. Understand the relationship between: Boolean algebra, truth-tables, and combinational circuits Design a 4-input combinational circuit from an informal description of its function. Understand the function of basic hardware components such as: flip/flops, registers, buses, adders, clocks, and control logic. Understand how these components are built from digital logic gates. Understand how the data-path and control of a processor are built from these components. Understand how computer pipelining can improve processor performance, the hazards that can impede such improvements, and the hardware and software techniques used to minimize the negative impacts of these hazards. Use computer performance equations and Amdahl's law to predict the performance impact of changes to hardware configurations such as faster clocks, better floating–point units, processor pipelining, hazard mitigation, cache organization etc. Understand the implications of today's multi-core processor chips, including synchronization and memory consistency. Create simple assembly language programs. Understand (in some detail) how computer hardware executes such programs.

CMP 335 Computer Organization Laboratory

A practical course applying principles taught in CMP 334. Construction and operation of the separate hardware components; system software installation.

See CMP 334 for Learning Objectives

CMP 338 Data Structures and Algorithms I

Abstract characterizations of data structures such as arrays, stacks, queues, trees and graphs will be studied along with algorithms that make use of such structures, including algorithms for sorting, searching, and memory management. Implementation issues will be considered, and students will write programs that embody these structures and algorithms.

Improve skills in object-oriented programming Improve understanding of recursive methods Understand a core group of basic data structures as enumerated in topics below Be able to conceptualize many programming issues at a higher level through data structures Know the tradeoffs of each studied data structure so as to employ the appropriate one for a given situation Be able to write parameterized data structures using generics Be able to design algorithms that incorporate data structures for efficient handling of data Be able to code algorithms involving data structures using an object oriented programming language Be able to analyze new data structures and their algorithms for asymptotic behavior Achieve a level of maturity in the subject so that further study of data structures can be pursued independently

CMP 339 Programming Languages

Design, evaluation and implementation of programming languages. Syntax and semantics of language constraints. Control structures, including branching, selection, iteration, procedure calls and recursion; data types and data structures, and operations on them; name structures, binding, environments, dynamic and static scoping; parameter passing.

Learning objectives / syllabus unavailable

CMP 340 Introduction to Probability for Computer Science

Introductory probability theory with applications to computer science. Axioms of probability, conditional probability, discrete and continuous random variables, expectation, variance and covariance, law of large numbers, central limit theorem.

Understand and apply basic probability theory, including random variables, distributions, and expectation. Understand and apply the law of large numbers and central limit theorem. Be prepared for learning advanced topics in Computer Science that rely on probability theory.

CMP 342 Internet Programming

Programming using languages for the Internet, such as Java or Visual Basic. Web server management, including administrative software tools.

Design and Develop event driven GUI based applications Design socket based communication systems to simulate asynchronous I/O. Design systems with multiple threads of execution. Determine the containment of a point in a circle and understand the math involved. Design and develop functional and well formatted web pages using HTML and CSS Use JavaScript to generate dynamic web content. Write code that responds to timers. Write code that responds to user generated events. Perform Encryption/Decryption via a One Time Pad System. Perform Encoding/Decoding using 1-bit error correcting Hamming Code.

CMP 343 Full Stack Web Development

Full stack web development through experiential project-based learning using apis, protocols, and popular frameworks.

Explain the concepts, implement the syntax, and know the benefits of using JavaScript frameworks on the client-side as well as server-side. Develop aesthetically pleasing web applications that include back-end constructs such as databases and application servers. Describe the benefits of: various architectures design patterns elements of production-readiness historical revisions and version adoption by browsers

CMP 346 Object Oriented Techniques

Study of design patterns and a virtual machine. Class file structure, the virtual machine computer architecture and instruction set, Just-In-Time compilation, different forms of garbage collectors and their merits, and performance enhancements.

Learning objectives / syllabus unavailable

CMP 405 Introduction to Networks

Introduction to network protocols and algorithms. Intensive study of the most important protocols at each layer. Examination of their strengths and weaknesses. Basic algorithms for identifying primary servers, constructing forwarding and broadcasting trees, and determining routing tables. Writing a simple networking service at the I.P. layer or higher. Lab exercises include building and testing small networks.

Define and use basic concepts and terminologies for networking. Describe the layers of the TCP/IP reference model and their purposes. Work with classful and classless internet addressing. Explain the Address Resolution Protocol. Identify the elements of segments, datagrams and Ethernet frames. Explain datagram fragmentation. Understand internet routing and routing protocols. Explain how routing protocols such as BGP, RIP, and OSPF work. Explain UDP, TCP and ICMP. Understand sliding window protocols.

CMP 406 Network Systems Lab

Introduction to installation and configuration of networking equipment, network monitoring and trouble-shooting, and network management. Use of typical networking equipment. Some programming in Java or C will be required.

Learning objectives / syllabus unavailable

CMP 407 Compiler Construction

Compilation of simple expressions and statements. Organization of a compiler, including compile-time and run-time symbol tables, lexical scan, object code generation, error diagnostics, object-code optimization techniques such as register allocation via graph coloring, and method in-lining. Use of compiler writing languages and bootstrapping.

Learning objectives / syllabus unavailable

CMP 408 Software Engineering

Classical and object-oriented software engineering. System life cycle, structured coding techniques, modularity, data encapsulation and generic facilities. Verification, validation and testing techniques: static analysis, input-output assertions, structured induction, and symbolic execution.

Software Engineering Introduction: Engineering Software Process Agile Software Development Requirements Engineering System Modeling Architectural Design Design and Implementation Software Testing Software Evolution System Dependability and Security: Dependable Systems Reliability Engineering Safety Engineering Security Engineering Resilience Engineering Advanced Software Engineering: Software Reuse Component-based Software Engineering Distributed Software Engineering Service-oriented Software Engineering Systems Enginnering Systems of Systems Real-time Software Engineering Software Management: Project Management Project Planning Quality Management Configuration Management

CMP 409 Security of Networks

Introduction to attack and defense in network security. Basic tools for both attacking and defending networks and their use.

Learning objectives / syllabus unavailable

CMP 410 Data Structures and Algorithms II

Design and analysis of algorithms: Worst and average case behavior. Design strategies, such as divide and conquer, the greedy principle, dynamic programming, and backtracking, are illustrated by examples chosen from sorting and searching, applications of graph theory, scheduling, pattern matching, matrix multiplication, and other topics. NP-complete problems. Parallel processing and algorithms.

Learning objectives / syllabus unavailable

CMP 412 Computer Graphics

Techniques of producing graphical displays. Two-dimensional transformations. World vs. Screen coordinates. Line drawing algorithms. Clipping. Motion. Use of devices such as light pens, graphics tablets, and mouses.

Learning objectives / syllabus unavailable

CMP 414 Artificial Intelligence

Survey of artificial intelligence with an introduction to LISP programming. Graph-searching algorithms applied to problem solving. Selected topics from pattern recognition, natural language processing, and expert systems.

Handle large volumes of data using python scientific libraries. Understand the concepts and procedures for commonly-used machine learning algorithms. Given a particular learning task, build a machine learning model and train the model on the training dataset. Tuning a machine learning model to improve its performance. Assess model fairness and identify data privacy leaks.

CMP 415 Machine Learning

Introduction to applied machine learning models and algorithms using a high level programming language and relevant scientific libraries.

Prepare data sets for training machine learning models. Apply the concepts and procedures for popular machine learning algorithms. Select proper methods to build and train machine learning models. Tune a machine learning model to improve its performance. Evaluate the performance of the machine learning model using the test dataset.

CMP 416 Computability Theory

Mathematical formulation of computability theory and abstract machine theory. Finite-state machines and Turing machines; Church-Turing Thesis; recursive functions and recursively enumerable sets; unsolvability and the Halting Problem.

Learning objectives / syllabus unavailable

CMP 417 Cryptography

Methods for transforming information into cryptic messages and for deciphering secret code. Review of selected topics in elementary number theory. Construction of linear, quadratic, and exponential ciphers, as well as key exchange protocols.

Learning objectives / syllabus unavailable

CMP 418 Systems Analysis and Design

Development of many systems will be studied using a structured approach to system design. Feasibility study, project cost estimation, top down design, Hierarchy/Input/Process/ Output (HIPO), data flow diagrams, data dictionaries, backup, security, maintenance and testing system.

Learning objectives / syllabus unavailable

CMP 420 Database Systems

Introduction to theory of database systems and database management: theory of relational, hierarchical, and network database organization, with emphasis on the first; views of data, data organization, data dependency and redundancy, normal forms, and query language.

1. Understand fundamental concepts of modern database systems. 2. Describe database systems concepts and architecture. 3. Perform conceptual data modeling by ER/EER (Entity Relationship/Enhanced Entity Relationship). 4. Understand the relational model. 5. Work with Structured Query Language (SQL). 6. Explain functional dependencies and normalization as a database design process. 7. Understand File/index structures. 8. Explain physical database design decisions, transaction processing concepts and theory, concurrency control techniques. 9. Discuss database recovery techniques.

CMP 425 System Programming Languages

Assembly language for one machine language. Projects to illustrate programming techniques in assembly language including, loops, procedure and subroutine call and return, parameter passing, call stacks, data structures. Introduction to the C language- including the C preprocessor, macros, and syntax. Data types, structures, pointers and pointers to functions.

Learning objectives / syllabus unavailable

CMP 426 Operating Systems

Operating systems and their role in various types of computer systems; the principles of multiprogramming; algorithms for resource allocation; multiple-computer systems.

1. Use the command line to interact with the operating system 2. Explain operating systems and their role in various types of computer systems 3. Describe the basic structure of an operating system and its components 4. Describe the concept of a process, threads, and how processes deal with scheduling, cooperation, and communication with other processes 5. Describe CPU scheduling algorithms and be able to compare them 6. Explain the classical problems in process synchronization and know several different ways to solve such problems, including semaphores, critical regions, and monitors 7. Explain how to characterize and handle deadlocks, including prevention, avoidance, detection, and recovery 8. Understand main memory and virtual memory and describe several different schemes for managing main memory, including swapping, virtual memory, paging, and segmentation 9. Describe file/storage management 10. Discuss how disks are structured and how their space is managed by the operating system

CMP 428 Video Game Programming

General game architecture, asynchronous input, animated sprites, action oriented A.I., collision detection, scrolling, sound clips, 3D graphics. Student projects involving development of several video games, both individually and in teams.

how to implement the drawing of polygons from the ability to draw a pixel what a real-time system is and be able to implement one in Java the features of a game loop and how to implement one how to read and respond to asynchronous input within a game loop how to move objects around on the screen based on user input, A.I., and classical mechanics how to detect collisions between objects represented on the screen how and why to use double buffering and page flipping how to implement an object to encapsulate frame based animation what translation and rotation are and know the underlying mathematics what a polygon model is and how to implement them how to implement scrolling and parallax scrolling and understand the underlying principle how to implement choreographed AI how to implement chasing/evade and targeting algorithms and understand the mechanisms behind them how to utilize finite state machines to implement game object behavior how to use path finding algorithms such as A* how to perform perspective transformations in order to display 3D scenes how to implement billboarding and understand the mechanisms behind it how to implement backface removal and understand the mathematics behind it how to use the painter's algorithms to ensure correct occlusion how texture mapping is performed and the mathematics behind it. 21) how to partition the game world for efficient processing of events

CMP 430 Mobile Programming

Design and implementation of applications running on a mobile platform such as smart phones and tablets. Learn programming languages and development tools for mobile SDKs. Writing code to exercise important features of mobile devices. (May be re-elected for credit as often as the topic changes.)

Demonstrate understanding of the essentials of mobile application development. Develop applications that demonstrate proper use of the application lifecycle. Create user interfaces, activities and handle events Work with threads, files, and databases on mobile devices.

CMP 431 Mobile Programming for Android

Introduction to Android Mobile Programming. Developing applications for devices running the Android Operating System using the Android Studio IDE to develop their Android Applications.

1. Develop Android mobile applications. 2. Develop applications that properly handle the Android application lifecycle. 3. Create user interfaces, activities and handle events. 4. Pass data between activities. 5. Use themes, layouts and styles. 6. Work with menus, tabs, preferences, and settings. 7. Work with threads and files on Android mobile devices. 8. Use intents, services, notifications, alarms, and broadcast receivers. 9. Store data using SQLite database 10. (If time permits) Work with Content Providers

CMP 432 Mobile Programming for iOS

Introduction to IOS programming using Apple's new language Swift. Developing applications for Apple's iPhones and iPads using the SwiftUI framework and Xcode

1. Demonstrate understanding of the essentials of iOS mobile application development. 2. UseApple'sIntegratedDevelopmentEnvironment(IDE)XCode. 3. Develop applications that demonstrate proper use of the iOS application lifecycle. 4. Create user interfaces, activities and handle events using SwiftUI. 5. Work with Queues (threads) and files on iOS mobile devices. 6. Use the Foundation library of iOS tools.

CMP 436 Introduction to Enterprise Computing

Technical issues and principles for building distributed enterprise systems. Applications of these principles using the Java EE framework. Server-side distributed component model such as Enterprise Java Beans and Web Services.

Learning objectives / syllabus unavailable

CMP 438 Communicating Robots

Techniques and principles for building communicating robots. Programming on resource-limited systems, designing communications protocols, and testing distributed algorithms. Project to involve building a robot to work/compete with other robots.

Learning objectives / syllabus unavailable

CMP 444 Modeling Cellular Networks

Mathematical models and computational tools for studying regulatory networks in the cell.

Learning objectives / syllabus unavailable

CMP 446 Computational Tools for Bioinformatics

Foundational tools, techniques and algorithms for Bioinformatics and Molecular Biology.

Learning objectives / syllabus unavailable

CMP 447 Linear Programming and Operations Research

Introduction to linear programming and other optimization techniques in Operations Research with applications of practical problems and theoretical computer science.

Articulate and apply the basic methods of linear and nonlinear programming problems, including duality. Formulate linear programming models. Solve linear programming problems using simplex method. Conduct sensitivity analysis of linear programming problems.

CMP 464 Topics in Computer Science

Various sections on topics in computer science. (For specific topics and sections, consult the Department.)

Learning objectives / syllabus unavailable

CMP 475 Combinatorial & Graph Algorithms

Demonstrate understanding of basic concepts in Graph Theory. Represent a given graph as adjacency lists or an adjacency matrix. Describe and implement the breadth first search and the depth first search algorithm. Apply Prim and Kruskal algorithm to solve minimal spanning tree problems Apply algorithms to solve shortest path problems. Apply algorithms to solve maximum problems

CMP 476 Parallel Algorithms & Architecture

Know the advantages and disadvantages of using parallel versus serial hardware. Know use cases of various parallel hardware configurations. Apply the concept of Thread Level Parallelism. Apply the concept of Instruction Level Parallelism. Be cognizant of the inherent difficulties in devising parallel algorithms and their implementations and know how to overcome these issues. Understand how to navigate memory hierarchies of parallel hardware to efficiently move data to where it is needed. Perform intelligent subdivisions and agglomerations. Recognize components of implementations of parallel primitives such as Prefix Scan and Fast Fourier Transform. Explain how parallel primitives are used in devising parallel algorithms. Be prepared for further study of parallel algorithms and architecture so as to facilitate targeted independent study in the area.

CMP 485 Independent Study

Independent study on a specific topic under faculty supervision.

Various topics

CMP 485 Independent Study - Technical Interview Preparation

Technical Interview Preparation: This course is designed to help you prepare for software engineering and related interviews in the tech industry.

1. Build a professional portfolio to help you land an interview at your dream company 2. Sharpen your technical skills to help you nail your technical interview 3. Refine your soft skills to make a better impression during your behavioral interview 4. Gain the resources, comfort and confidence to move your career forward

CMP 486 Independent Study

Not allowed as elective for Mathematics or Computer Science concentrations.

Learning objectives / syllabus unavailable

CMP 487 Internship in Computer Science

On-the-job training in a public or private institution.

CMP 487 is only available during the Fall and Spring semesters. The course does not have to be taken at the same time as the internship. CMP 487 can be taken up to two times, to earn credit for two separate internships. To earn credit for CMP 487 and the internship a 7-10 page essay style report must be submitted before the end of the semester that you are registered for CMP 487. The specific deadline is provided each semester.

Section 3.11: Computer Information Systems Course Descriptions

CIS 106 Computer Literacy

Use of software packages for word processing, database manipulation, and spreadsheet calculation. Introduction to the Internet and the World Wide Web.

Learning objectives / syllabus unavailable

(Withdrawn) CIS 166 Computer Programming for Information Processing I

Structured computer programming using a modern high-level programming language. Includes console I/O, data types, variables, control structures, including iteration, arrays, function definitions and calls, parameter passing, functional decomposition, and an introduction to objects. Debugging techniques.

Write programs using looping and decision constructs. Design and use methods in programs. Understand the use of parameters and information passing in programs. Write programs with system and file I/O. Write and debug programs independently.

(Withdrawn) CIS 211 Computer Information Systems

Survey of the technology and applications of computer- based information systems in business.

This course gives students a solid and concise foundation in the fundamentals of information systems through the most recent research, references, and examples in the field. Students will explore topics such as multimedia in today's business; application development for the iPhone, iPad, and similar devices; cloud computing; forecasting; and environmental design and green computing. Business-related examples of supply chain management (SCM) and customer relationship management (CRM) are provided as well. Finally, students will discuss communities and work structures, including how social networking sites, such as Facebook and Twitter, are assisting virtual teams and how companies are effectively using virtual organizational structures with mobile workers.

CIS 212 Microcomputer Architecture

Architecture of microcomputer systems and its supporting system software. Various microprocessor systems, expansion bus design, memory design and management, secondary storage technologies and management, peripherals, and telecommunication technologies.

Define systems architecture and related terms Describe the components and functions of computer networks Describe numbering systems and their use in data representation Describe CPU instruction and execution cycles Describe the distinguishing characteristics of primary and secondary storage Describe how the CPU and bus interact with peripheral devices Describe basic concepts of text and image representation and display Explain communication protocols Describe logical and physical network topologies Describe the application development process and the role of methodologies, models, and tools

CIS 213 Microcomputer Architecture Lab

Practical application of concepts learned in CIS212 - Architecture of microcomputer systems.

1. Define systems architecture and related terms 2. Define and identify the components andfunctions of computernetworks 3. Describe numbering systems and their use in data representation 4. Describe CPU instruction and execution cycles 5. Describe the distinguishing characteristics of primary and secondary storage 6. Demonstrate how the CPU and bus interact with peripheral devices 7. Describe basic concepts of text and image representation and display 8. Explain communication protocols 9. Describe logical and physical network topologies 10.Describe the application development process and the role of methodologies, models, and tools

CIS 216 Computer Group Productivity Tools

Use and characteristics of basic group-ware and productivity tools such as electronic mail and messaging, presentation creation, group calendaring and scheduling, electronic meeting systems, desktop and realtime data conferencing, gourp document handling, work-flow and work-group utilities and group-ware developmental tools.

Learning objectives / syllabus unavailable

(Withdrawn) CIS 228 The Internet

Website creation, coding HTML documents, tags of the HTML language, style sheets. Writing and embedding simple java script code.

• Learn HTML, CSS, and JavaScript 1.8 • Create cross-browser webpages • Write source code in a widely accepted format • Develop error-free HTML, CSS and JavaScript codes • Understand and utilize web developer tools

CIS 234 Introduction to Spreadsheet Analysis

Use of spreadsheet software for elementary data analysis, simple modeling and forecasting. Worksheets, files, graphs, and formatted output and screen presentation. Macro programming.

Describe the concepts and capabilities of spreadsheet software Create, format, revise and print a spreadsheet Use simple and advanced functions Manipulate data in an excel database Analyze list data Use financial functions Solve complex problems with goal seek, solver, and scenario manager Create and use 2 variable data tables Create charts

CIS 241 Computer Graphics in Business

Applications of graphics in business as a reporting and analysis tool: presentation packages, project management packages, desktop publishing, CAD/CAM. Overview of the methods (software and hardware) used for generating graphics, including algorithmic image manipulation, interactive processing, input tools and display tools.

Learning objectives / syllabus unavailable

CIS 242 Introduction to Systems Analysis and Design

Study of a computer system life cycle via a structured approach: problem definition, feasibility study, cost estimation, analysis, design, implementation, testing and maintenance. Hardware and software organization. Several case studies will be considered.

Learning objectives / syllabus unavailable

CIS 243 Introduction to Operating Systems

Overview of operating systems (OS) from both a theoretical and a systems manager point of view. Process management and multitasking, memory management, resource management, file management, I/O management, command interpreter/shell and shell scripts will be covered. Large networked systems will be discussed from a systems point of view.

Learning objectives / syllabus unavailable

CIS 244 Introduction to Database Management

Fundamental concepts of database organization: fields, records, tables, indexes, queries, forms, reports. Use of a relational database system for loading, modifying and querying a data base. Programming in a database language.

A. Create MySQL databases using the windows command line or MacOSX terminal B. Design different types of databases using MySQL C. Understand fields, tables, Keys, indexes and data types and apply them in MySQL D. Create different types SQL Queries and Relational Algebra (Inner joins) using

MySQL E. Understand normalization 1NF, 2NF and 3NF F. Understand the types of relationships in a database 1:1, 1:M and M:M

(Withdrawn) CIS 246 E-Commerce

Introduction to electronic commerce on the Internet. Designing an e-commerce site including web page content and development, site marketing and advertisement, legal and security considerations, shopping cart management, credit card and other debit transactions.

Learning objectives / syllabus unavailable

CIS 247 Practical Unix: Programming and System Administration

Topics chosen from the following: text editors, file system, utility programs, pipe and filter paradigms, shell language programming; tools for maintenance of normal system operation, security, hardware and software configuration management and network connections.

Learning objectives / syllabus unavailable

CIS 249 Introduction to Local Area Networks (LAN's)

An overview of LAN's from the point of view of a network administrator as well as a hands-on introduction to a popular network operating system. General topics will include LAN media, topologies, protocols, multi-platform connectivity, remote access, and rudimentary internetworking.

Learning objectives / syllabus unavailable

CIS 252 Topics in Computer Applications

(May be repeated for credit as the topic changes up to a maximum of 6 credits.) (For specific topics, see the Department.) Various sections on topics in Computer Applications.

Learning objectives / syllabus unavailable

CIS 266 Computer Programming for Information Processing II

Techniques of business information processing using object-oriented programming. Randon access files, data management and control, variable arrays, object variables. Introduction to advanced techniques.

Learning objectives / syllabus unavailable

CIS 299 Elective Credit

Elective Credit

Learning objectives / syllabus unavailable

CIS 324 Spreadsheet Design and Programming

Advanced features of spreadsheets, including graphics, database manipulation, built-in functions, and macros, applied to a variety of business-oriented problems.

Learning objectives / syllabus unavailable

CIS 329 Local Area Networks (LAN's)

An overview of LANs as well as hands-on introduction to a popular network operating system. General topics will include LAN media, topologies, protocols, multiplaform connectivity, remote access, and rudimentary internet working.

Learning objectives / syllabus unavailable

CIS 331 Network Introduction

Introduction to network technologies (Ethernet, ATM, WiFi, Bluetooth, ZigBee), network architectures (telephone, OSI, and Internet), and standard tools for administering and monitoring networks. Evaluation of network technologies and designs for supporting some services; design and configuring networks for those services.

Define and use basic concepts and terminologies for networking. Describe the layers of the TCP/IP reference model and their purposes. Work with classful and classless internet addressing. Explain the Address Resolution Protocol. Identify the elements of segments, datagrams and Ethernet frames. Explain datagram fragmentation. Understand internet routing and routing protocols. Explain how routing protocols such as BGP, RIP, and OSPF work. Explain UDP, TCP and ICMP. Understand sliding window protocols.

CIS 333 Network Security

Introduction to securing networks, with emphasis on firewalls, intrusion detection, and monitoring tools. Monitoring and improving the security of an organizations network. Building firewalls and configuring intrusion detection systems. Detecting some well-known attacks.

Learning objectives / syllabus unavailable

CIS 341 Computer System Fundamentals

Examination, removal, and reassembly of computer hardware components, such as processors, disks, memory, and buses. Installing and operating the following computer system components: operating systems, user interfaces, subsystems (such as Web servers), development environments, communications, and distributed file systems. Performance characteristics also will be discussed and measured.

Learning objectives / syllabus unavailable

CIS 344 Database Design and Programming

Programming in a database language. Emphasis on programming ideas and techniques and user interfaces in a modern database system. Review of elementary relational database concepts, with emphasis on programming rather than theory.

A. Properly design databases B. Understand and create ER Diagrams, EER Diagrams and Database Schemas C. Apply cardinality rules to database structures D. Create databases using SQL with the Command Line or Mac OSX terminal E. Create tables, relationships, indexes, views, stored procedures, and apply appropriate data types with the corresponding field using SQL F. Create different types of SQL queries and subqueries using SQL G. Understand relational algebra (Inner joins, left join, Right join etc.) H. Understand Normalization 1NF, 2NF, 3NF, BCNF, and 4NF I. Understand the different types of relationships that exist in a database 1:1, 1:M & M:M J. Understand the use of DDL, DML and DCL K. Understand and apply conditional statements in MySQL

CIS 345 Introduction to Operating Systems

Overview of operating systems (O.S.)from both a theoretical and a systems manager point of view. Process management and multitasking, memory management, resource management, file management, I/O management, command interpreter/shell, and shell scripts will be covered. Large networked systems will be discussed from a systems point of view.

Describe the purposes and structure of the major subsystems of an Operating System . Explain Processes and Threads, their relationship and the advantages and drawbacks of each Describe the various CPU scheduling algorithms and their characteristics and uses. Understand the various types of Process synchronization, critical sections, and where they are used   Explain Real and Virtual Memory management, including the algorithms and the justification. Understand the File System Interface and Implementation including the internal structure required for implementation. Understanding Virtualization and the cloud, what is its purpose and the advantages. Create and Amazon Linux Virtual Machine and be able to understand basic OS concepts Explore important concepts of an OS such as processes, threads, security etc.

CIS 346 E-Commerce

Introduction to electronic commerce on the Internet: Designing an e-commerce site including web Server installation, configuration, and tuning; web page content and development, site marketing and advertisement, legal and security considerations, shopping cart management, credit card and other debit transactions.

Understand e-commerce concepts and the processes that are involved in launching, operating, and managing business activity on the World Wide Web. Use of Apache Web server, PHP, MySQL, and Zen cart Develop store catalog, set taxation, shipment options, payment options, and advertisement Backup Database

CIS 349 Data Communications and Distributed Networks

Data communications: standard models, system operations, major components, digital transmission (including some current schemes such as SONET, ISDN and ATM). Hands-on introduction to local area network architectures, link-layer protocols and their design and analysis.

Learning objectives / syllabus unavailable

CIS 356 Intermediate-Level Topics in Computer Applications

(May be repeated for credit as the topic changes up to a maximum of 8 credits.) (for specific topics, see the Department). Various sections on topics in Computer Applications at the intermediate level.

Learning objectives / syllabus unavailable

CIS 488 Internship in Computer Information Systems

On-the-job training in a public or private institution.

CIS 488 is only available during the Fall and Spring semesters. The course does not have to be taken at the same time as the internship. CIS 488 can be taken up to two times, to earn credit for two separate internships. To earn credit for CIS 488 and the internship a 7-10 page essay style report must be submitted before the end of the semester that you are registered for CIS 488. The specific deadline is provided each semester.

Chapter 4: Students, Services, and Associated Supports

Section 4.1: Characteristics of the Lehman College Student Body

One might say that the student body at Lehman College is diverse and in so doing one would be understated. The simple statistics only hint at the wealth of cultural, linguistic, intellectual, professional, emotional, and social diversity of our students, who hail from 95 countries. Across the past five years, our students are approximately 70% female. Six in ten undergraduates are enrolled in full-time study, reflecting the pressures on our students to work in order to contribute financially to the economic stability of their families. Approximately half of Lehman College students are first-generation college students; these graduates provide a crucial catalyst for their own descendants and family to pursue college-level studies. Nearly 60% of Lehman undergraduates, and 43% of graduate students, live in the Bronx; as the only public, four-year college in the Bronx, Lehman College provides a crucial engine for educational advancement for residents of this borough. The remaining students at Lehman travel to the Bronx from other boroughs in New York City, as well as from surrounding communities in Westchester, New Jersey, and Long Island.

The average age of Lehman College undergraduates is 26 years old; for graduate students, the average age is 34 years old. Because our students tend to be slightly older than college students who move directly from high school into a four-year undergraduate college, our students come with significant professional experience as well as personal/domestic. The average age of Lehman College undergraduates is 26 years old; for graduate students, the average age is 34 years old. Because our students tend to be slightly older than college students who move directly from high school into a four-year undergraduate college, our students come with significant professional experience as well as personal/domestic experience, often with their own children. Twenty percent of Lehman College students are parents; the preponderance of our students live in the context of their families, either as adults living with their parents, siblings, and other relatives, functioning as part of the economic unit of their families, or living in intergenerational settings that often also include their own children. As a result, our students are deeply connected to their families and communities.

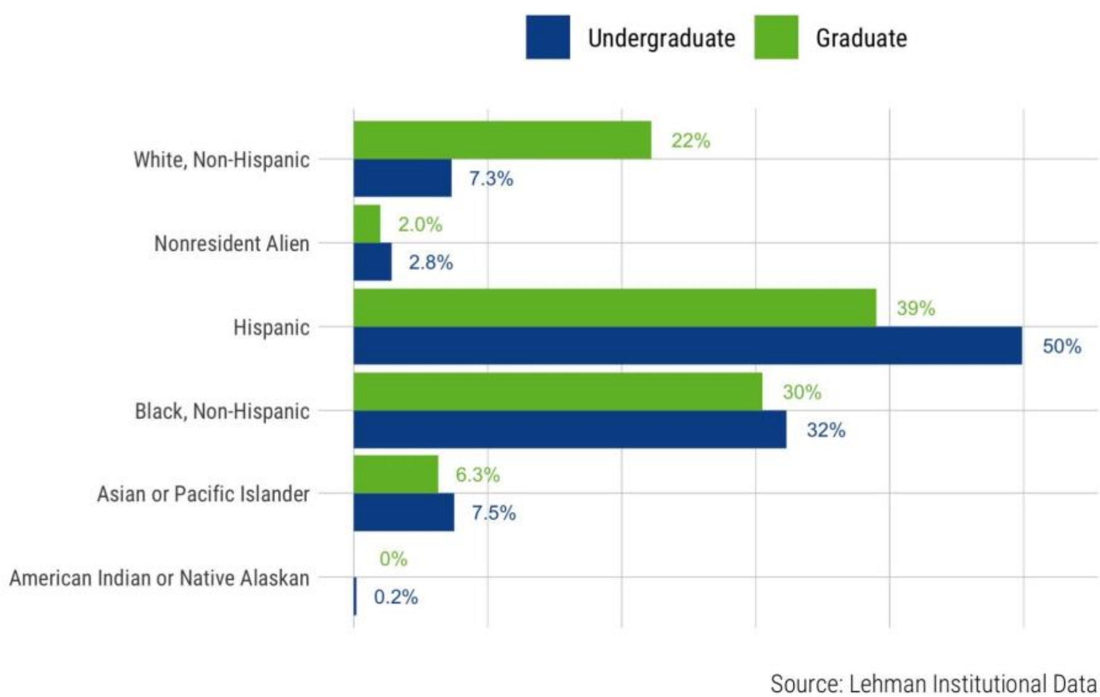


Figure 4-1: Race, ethnicity, or immigration status for Lehman students in Fall 2020

As a commuter-campus, the College strives to ensure that students have access to the extra-curricular activities that are foundational to a vibrant college life, even as students have the flexibility that they need to balance their studies and student experiences with their responsibilities as parents, family members, and workers at their (often full-time) jobs. Since 2013, the College has expanded the number of students enrolled as well as total degrees granted. Nearly 15,000 undergraduate and graduate students have registered in recent years, reaching the College's largest enrollment in over four decades. The College has graduated more than 81,000 alumni, over half of whom remain in the Bronx and surrounding areas, giving back and strengthening their communities through their chosen professions. In 2019-2020, 3,545 degrees and certificates were awarded, an increase of 28% over the number of degrees and certificates awarded in 2013-2014 (2,769).

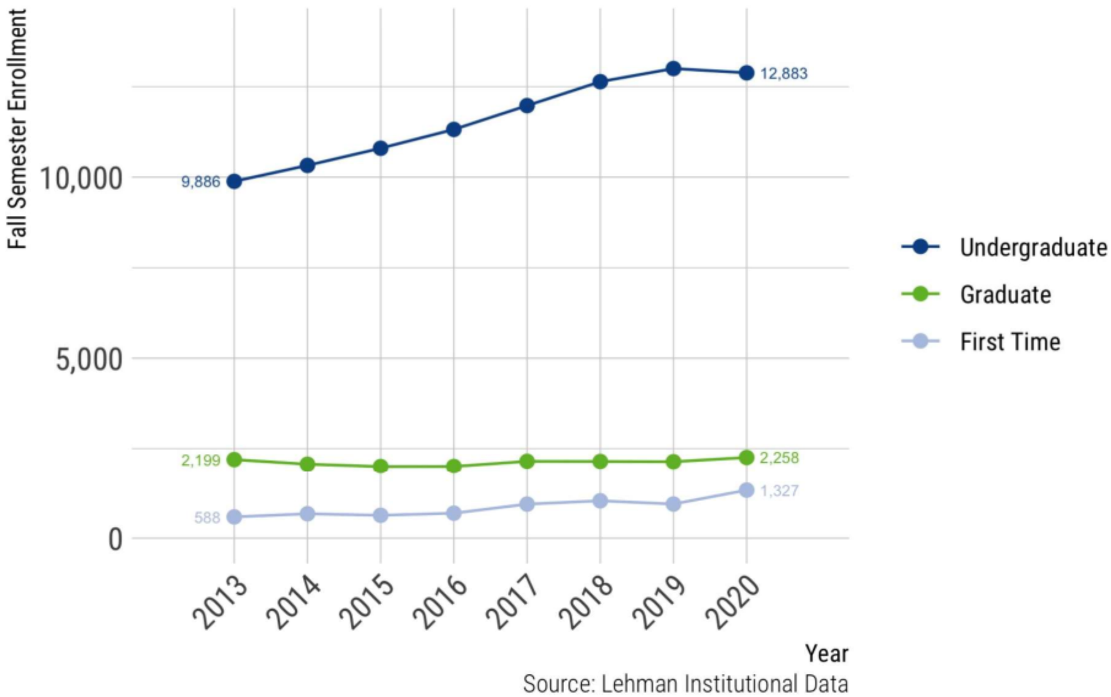


Figure 4-2: Overall enrollment at Lehman College from Fall 2013 through Fall 2020. While graduate enrollment has been essentially flat through this period, undergraduate enrollment has increase by approximately 30% and the number of first-time full-time enrollees has more than doubles.

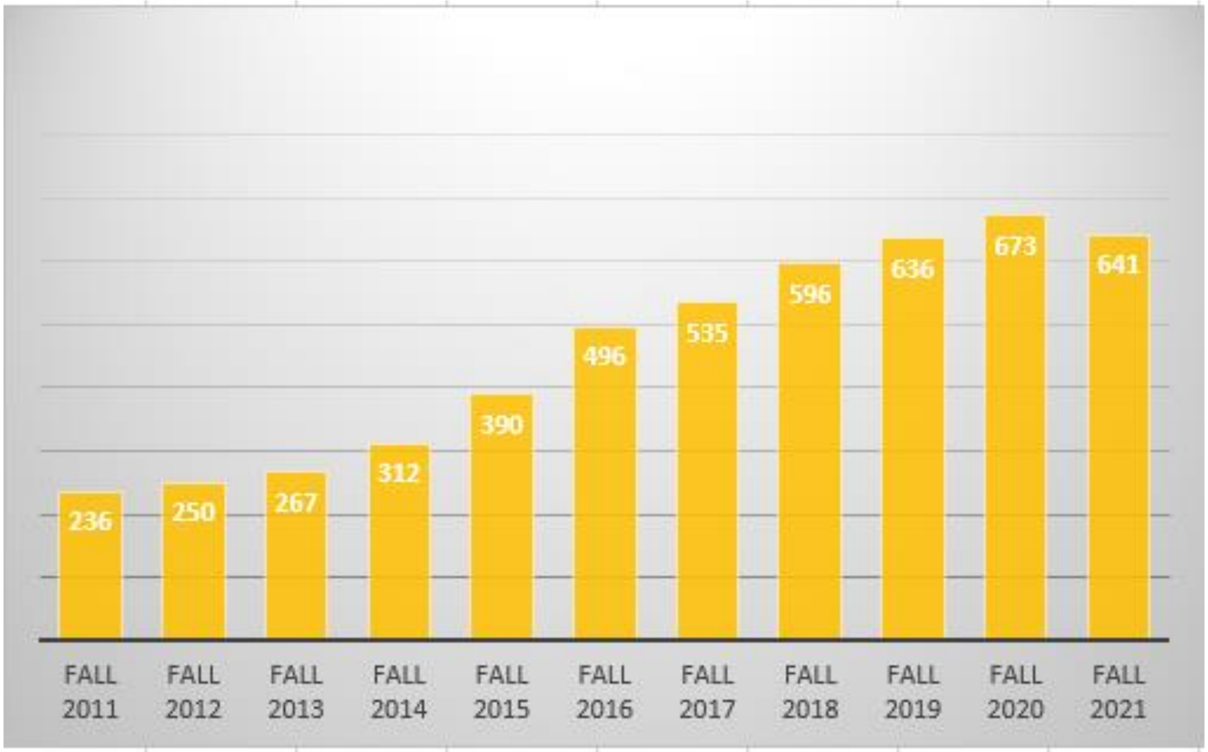


Figure 4-3: Number of declared Computer Major from Fall 2011 to Fall 2021. Source: Institutional Research

Section 4.2: Computer Science Students

With the growth of the technology industry, the number of students declared as majors in the Department of Computer Science has grown rapidly in the last ten years. The number of students declared as a major offered in the computer science department was 236 in Fall 2011. The number nearly tripled by Fall 2020 with 673 students. CUNYFirst queries which track closely with the data from Institutional Research for the past semesters presented in Figures 4.3 and 4.4 indicate that at the close of the Fall 2022 semester the department had over 700. We can reasonably expect continued growth in the next ten years based on projections by the U.S. Bureau of Labor Statistics indicating that there will be a 25% growth rate for software developer jobs from 2021-2031. This means 411K more industry positions in the next ten years. Such positions typically requires B.S. as an entry-level education.

Figure 4-4: The number of students enrolled in different majors offered by the Computer Science department. Source: Institutional Research

Both our undergraduate majors, Computer Information Systems and Computer Science, have seen tremendous growth over the last ten years. The number of undergraduates majoring in Computer Science grew from 94 in Fall 2011 to 406 in Fall 2021, with a growth rate of over 300% and the number of Computer Information Systems majors almost doubled over the last ten years. On the other hand, our master's program stayed roughly the same level across the years. The reason that the master's program did not increase with the undergraduate majors is two-fold. First, many jobs in the industry do not require a master's degree which matches with the desires of most of our undergraduate graduates who prefer an immediate job rather than going off to graduate school right away. Secondly, the number of faculty in the department limits the number of graduate courses the department can offer, which makes the program less competitive compared to other programs.

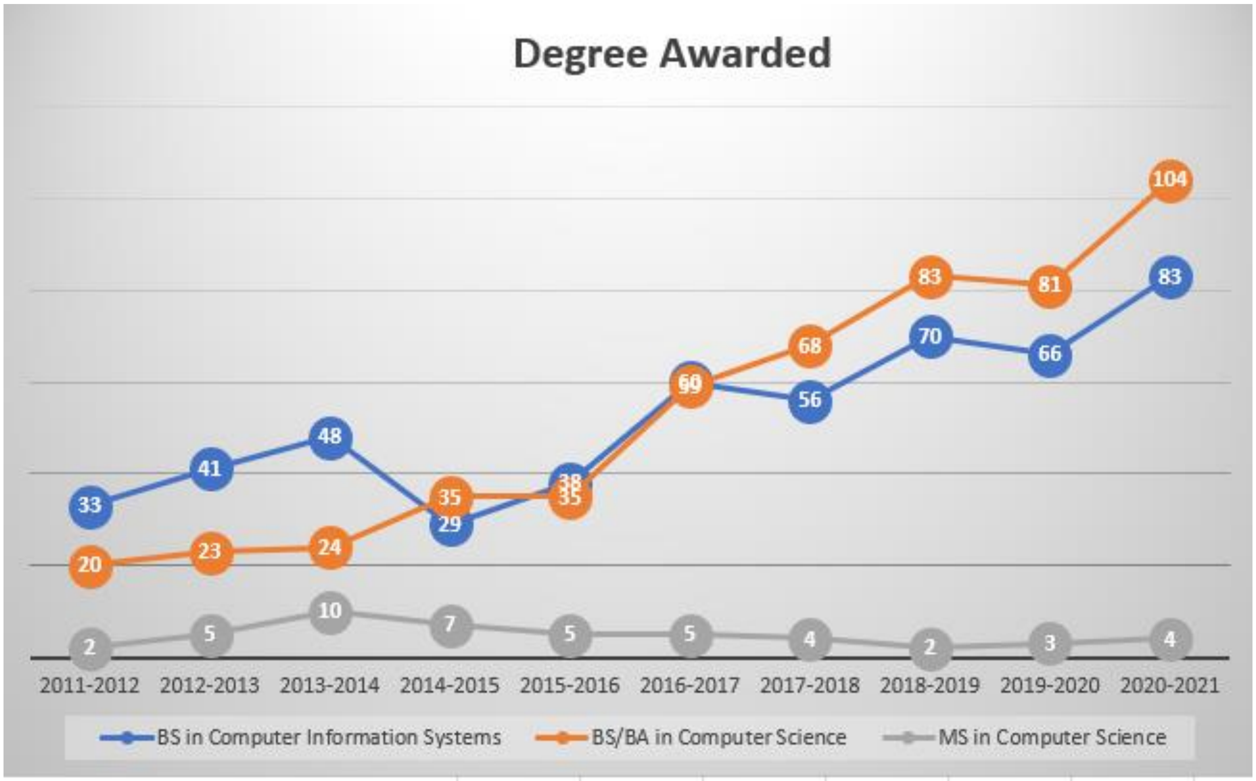


Figure 4-5: The degrees awarded in the past 10 years. Source: Institutional Research

The number of bachelor's degrees awarded by the department also increased from 53 in 2011-2012 to 187 in 2020-2021, which is consistent with the increase rate of the number of majors.

Section 4.3: Survey on Current Students

As part of our internal review, the department performed an anonymous student survey in Fall 2022 to assess how the Computer Science department and the courses that it offers are viewed by its students. The survey received 51 responses, with 60% of them coming from seniors and 31.4% from juniors. Respondents were asked "How satisfied are you with your experience with the Computer Science Department?" Most respondents were satisfied with 31.1% extremely or very satisfied and 51% satisfied. A total of 15.7% responded as not so satisfied and 7.8% as not at all satisfied. When asked to rate how much they learn in CMP/CIS classes, 80.4% students were satisfied with what they learned and 13.7% chose that they learned much more than expected. 76.5% of respondents believe that what they learned from the department is relevant to what they want to do in the future and 25.5% believe it is extremely relevant. 74.5% of students who took this survey believe they are on the right path to graduation and 31.4% are extremely confident. Another question in the survey asked about the future plans of the students: 80.4% of respondents want to find Computer Science-relevant jobs and 17.6% plan to apply for graduate schools.

As an optional question, students were asked, if they were willing to provide a short bio to better understand who our students were. Those bios are included below:

- *Currently a senior looking for an internship or new grad role, not sure whether at a startup or medium or bigger company. I've been at Lehman all 3 years so I mainly know java, I feel like the classes most relevant to a cs career have been CMP414 and CMP420 because I actually learned technical skills like python/ML and MySQL in them.*
- *My name is Charles James I am pursuing a BS in Computer Science at the rip old age of 33. My passion for Computer Science particularly Natural Language Processing started only 3 years ago and that's when I decided I need to acquire a degree in the field. Because the filed of NLP requires a graduate degree to be successful, I play on continuing my education after graduating from Lehman.*
- *I am a transfer student of Hispanic heritage with an interest in machine learning, software engineering, and robotics. I am considering attending graduate school for machine learning.*
- *I transferred to Lehman from another CUNY school and I wasn't majoring in computer science, so I'm only taking the starting classes of computer science like mat 175 and Cmp 128.*
- *As someone who got into programming at an early age, computer science as a degree and curriculum taught me the underlying concepts of software engineering. As a pre-requisite for my masters, taking comp science mad me realize what I want to do going forward and that's along the track of machine learning / ai. Taking an ai course allowed me to build a vin scanner for a Fortune 500 and I have Lehman to thank for that.*
- *Many students like myself want to pursue their career in the field, but I think that they are often overwhelmed with work outside of school. Providing more ressources for to allow students to cross the finish line, would be very helpful. Wether it's trough job opportunities on/off campus, info regarding scholarships...*
- *I study computer science at Lehman college. My career goal is to become a data scientist expert*
- *I'm passionate about becoming a full stack software engineer at any of the companies like Meta(Facebook), Amazon, Apple, Netflix, or Google.*

Another optional question asks students share an experience in the computer science department and their responses are included below:

- *We're not given much internship/post-grad/work resources. Most of the things my peers and I have learned about working in tech have all been from resources outside of school like:*

  -jobs we can get with a cs degree

  -startups vs big companies?

  -knowing what skills we actually need (what languages/frameworks/libraries etc. people actually use in the industry)

  -what technical interviewing is and how to prep

  -finding mentors/educational organizations

- *I understand school should teach the fundamentals and logic of CS, but at the same time how can the department also make the things we learn in school more relevant after we graduate/start to work?*
- *For a school that doesn't focus on Computer Science some of the professor are top notch.*
- *Please do more real life related tasks*
- *The teachers are so kind and helpful. They make learning so seamless if you do your part. It has gotten to the point where I enjoy learning and get a little excited for an upcoming lecture or the classes I'm going to take in future semesters.*
- *Ziou is an outstanding teacher. Kelly has engaging lectures and is a lot of fun, while also being really competant. Johnson is brilliant but lack of structure negates that. Fakhouri is a God tier lecturer. Fulakeza is fantastic. Bring back cornichiulo, his industry knowledge is outstanding. He could be the backbone of the department. Overall I greatly enjoyed the experience here. The one thing I take issue with is dumbing down exams and lectures to accommodate people who stand no shot of getting a job in industry. The teaching expertise is there but it seems like passing and graduation rates are prioritized over rigor. Although this is good in the short run, long term it's disastrous. Also, more coding assignments in upper level courses such as Networks, databases and operating systems. If a student can't handle that they might as well major in finance.... without solid programming skills it's impossible to get a swe job. On a final note, teaching at least one class with c++ would be transformative. Thanks for listening.*
- *Class availability had been an issue for me. For the database system course for spring for example we only have 1 section which is at night. With such restrictions in terms of number of sections, I think the course should be hyflex to allow us to participate.*
- *Professor Brian Murphy lectures are excellent and extremely helpful.*

- *I feel like my CMP 167 has been a bit rushed but I understand how it must move fast the way it does. I feel like I really have to take my time to learn everything and I've been struggling with CMP 167.*

Section 4.4: Presidential Scholars from the CS and CIS Programs

2012

Radhika Ramnauth, Andrae Raymond, Abdulrahman Bah, Said Broud, Juan Castillo, Raphael Correa E Silva, Steven Fulakeza, Carole Anne Hannon, Bradley Jensen, Dodi Katsan, Ngoc Kim, Miles Leon, Anthony Obando, Amos Omokpo, Reyane Oukpedjo, Hakim Oumarir, Ismail Rashid, Edwin Sanchez, Carmine Secchiano, Imran Wadud, Shamsudeen Yusif

2013

Igal Katan, Tesfaye Bogale, Robert Cavalluzzo, Piotr Gauza, Alfredo Gonzalez, Kevaughn Gordon, Albaro Guzman, Ariel Landron, Sylvester Nimako, Melvin Pena, Andrae Raymond, Dana Smith, Manuel Decena, Raymond Ferranti, Pablo Guerrero, Bradley Jensen, Miles Leon, Elio Mita, Davis Montero, Dmitry Mushkatinsky, Gene Nakagaki, Roger Ramdhan, Richard Roeung, Carmine Secchiano, Katherine Suazo, Victor Vasquez, Mirna Znait, Blen Bayissa, Said Broud

2014

Ahmed Ashraf, Tesfaye Bogale, Gaston Borda, Julissa Briones, Kevaughn Gordon, Albaro Guzman, Elio Mita, Riffat Mitul, Vanessa Ortiz, Zain Raza, Luis Rodriguez, Daniel Tapias, Daniel Vargas, Forest White, Elhadj Balde, Ahamed Abbas, Abdulrahman Bah, Gustavo Figueiredo, Tashieka Graham, Genaya James, Amber Lewis, Sandra Mamani, Edison Mejia, Edgar Morales, Gene Nakagaki, Ozurumba Onwuegbuchulam, Rebecca Ramnaresh, Pedro Ravel, Michael Rivera, Dana Smith, Katherine Suazo, Marlon Thompson, Andrea Vazquez, Mirna Znait, Igal Katan

2015

Deneice Allen, Dinelsi Almonte, Abdulrahman Bah, Dwight Beckett, Rafael Calderon, Ivan Gonzalez, David Gross, Bruno Guidugli, Mahedi Kaiser, Aurora Koch-Pongsema, Sandra Mamani, Edison Mejia, Edgar Morales, Rebecca Ramnaresh, Sandra Rivera, Richard Roeung, Dana Smith, Mirna Znait, Ahamed Abbas, Misha Alyas, Benjamin Ankomah, Ahmed Ashraf, Blenda Belliard, Cesar De Leon, Ibrahima Diallo, Christopher Fitzpatrick, Cesar Garcia, Reymy Gomez Hidalgo, Tashieka Graham, Genaya James, Juancarlos Manon, Andrew Marion, Ruber Montes de Oca, Leeana Persaud, Zain Raza, Amanda Rodriguez, Eddie Rodriguez, Faris Shehadeh, Forest White

2016

Ahamed Abbas, Aysmel Aguasvivas Velazquez, Misha Alyas, Anthony Beltran, Jasmeet Chawla, Hasan Chowdhury, Derian Escobar, Christopher Fitzpatrick, Kevin Gonzalez, Aurora Koch-Pongsema, Edgar Lucero, Keri Mallari, Sandy Maracallo, Andrew Marion, Carlos Marrero, Riffat Mitul, Rebecca Ramnaresh, Pedro Ravel, Luis Recio, Charlie Ung, Danielle Vasquez, Forest White, Angie Williams, Jonathan Feigenson, Thiry Ngin, Ahmed Ashraf, Abdul Aziz, Noranyi Brito Frometa, Elaine Burchman, Chanice Collins, Jessi Coplin, Jose Escudero, Sohan Ganesh, Raphy Gutierrez, Genaya James, Samuel Lin, Michael Nti, Luis Paulino, Yanilda Peralta Ramos, Timothy Rogers, Sumaiyah Syed, Yorranshell Wattson

2017

Dinelsi Almonte, Misha Alyas, Madelin Arias Bueno, Anik Banerji, Noranyi Brito Frometa, Ashley Broome, Elaine Burchman, Indira Castillo, Melissa Castillo, Edith Escamilla, Sean Feliciano, Younes Fellahi, Juan Gamarra, Huma Hack, Yasmin Lorenzo Vidal, Keri Mallari, Dalida Martinez, Andre-Ann McMillan - Tarnaguida, Lorre Mcintosh, Edison Mejia, Michael Nti, John Perdomo, Rebecca Ramnaresh, Orlando Roche, Lourdes Rodriguez, Michael Rodriguez, Tomas Rosendo, Mubtasim Sawonto, Charlie Vargas, Joel Vargas, Forest White, Md Alam, Rehan Anwar, Tahara Biva, David Broadbelt, Jose Cantres, Chanice Collins, Emmanuel Delgado, Robert Ellis, Manuel Emiliano, Peter Ferreira, Jamil Gafur, Johnn Garcia, Ismael Garrido Rodriguez, Jasmin Hernandez, Hanh Hoang, Julian Marin, Andrew Marion, Anthony Munoz Cid, Ikuseghan Pryce, Bryan Ramirez, Jose Mariz Salva, William Smith, Angie Williams, Ramon Feliz

2018

Patrick Abreu, Joseph Angeletti, Maurice August, Yisell Avila, Abdul Aziz, Elizabeth Bocs, Noranyi Brito Frometa, Jose Cantres, Indira Castillo, Melissa Castillo, Joshua Cordero, Peter Farquharson, Leomar Grullon Mendez, Christian Guerrero, Alaury Jorge, Derek Li, Samuel Lin, Tony Lu, Alexander Luciano, Nicholas Luckhai, Sabrina Mahjabin, Xhesika Malecaj, Andrew Marion, Jhony Maurad, Emeka Mbazor, Edison Mejia, Ali Mohamed, Ba Nguyen, Ifeatu Okaneme, Rayshon Patterson, Vantroi Perez, Jonathan Rivera, Orlando Roche, Amanda Rodriguez, David Rodriguez, Richard Rodriguez Urbaez, Jose Mariz Salva, Brandon Seda, Joel Tavarez, Jonathan Zacsh, Jose Zamora Orellana, Jorge Goris, Ricardo Robinson, Faris Shehadeh, Raymond Adjetey, Oumou Balde, Hydeia Blakey, Justin Castronuovo, Rene Clever, Harold De Leon Veras, Erika Dibenedetto, Sean Feliciano, Tommy Garcia, Ismael Garrido Rodriguez, Jonathan Guzman, Dhanraj Hiralal, Hanh Hoang, Yu Jen Hsieh, Shadel James, Raymond Martinez, Mahboba Mim, Mame Aissatou Ndiaye, Domagoj Ozimec, Brandon Quezada, Starlyn Reyes, Christopher Roberts, Joshua Roman, Cohomlan Sahossi, Estiber Santana, Ramon Santos, Ramish Saqib, Sabri Sonmez, Amado Suarez Rosario, Jose Toribio, Pamela Vargas, David Broadbelt, Matthew Dennis, Maximiliano Martinez

2019

Maurice August, Hector Cadeaux, Rene Clever, Sebastian Cruceanu, Harold De Leon Veras, Djibril Diallo, Erika Dibenedetto, Luisiana Dominguez, Muhamed Elmeetknany, Sean Feliciano, Ismael Garrido Rodriguez, Jonathan Guzman, Jacinto Herrera, Israt Jahan, Sulaiman Jalloh, Shadel James, Marc Johnson, Asifa Khan, Derek Li, Jayke Lim, Nicholas Luckhai, Gabrielle Lugo, Sabrina Mahjabin, Waleed Malik, Keri Mallari, Raymond Martinez, Mahboba Mim, Alvin Motieram, Ba Nguyen, Jarin Niha, Stanley Nunez Martinez, Rayshon Patterson, Bradly Perez, Vantroi Perez, Giovanne Quezada, Joan Quintero, Michael Rodriguez, Cohomlan Sahossi, Jose Mariz Salva, Tony Sanchez, Estiber Santana, Daniel Santiago, Mubtasim Sawonto, Edward Serrano-Rodriguez, Arezouma Solly Ngobou, Amado Suarez Rosario, Raihan Tanvir, Anthony Valerio, Danielle Vasquez, Rafael Velazquez, Gilberto Veras, Tyrell Whitfield, Aissa Zouhayr, Daniel Mejia, Temistocles Perez, Ricardo Robinson, Kofi Duodu, Abdul Malik Ahmed, Abdul Alim, Jeffrey Almanzar, Mark Ammari, Madelin Arias Bueno, Leah Benitez, Joshua Cordero, Janina Dizon, Eddie Dee Dulce, Brianna Duvivier, Christopher Estevez, Tanya Genao Matias, Kevin Gonzalez, Zinnia Gail Gotico, Kymarly Henry, Dhanraj Hiralal, Dana Jackson, Michael Lieberman, Kevin Lopera, Alexander Luciano, Wilibiely Lugo, Bryan Lynch, Omar Maldonado, Emeka Mbazor, Pawel Misiak, Juan Morillo, Denis Nezaj, Anapatricia Olvera Medina, Ashadur Omith, Justin Pagan, Ayesha Qureshi, Jonathan Rivera, Luis Rojas, Joshua Roman, Bryant Saavedra, Marianny Soto Sanchez, Sumaiyah Syed, Juliona Velazquez, Hanh Hoang, Amara Malik, Dardan Meha, Khadijah Mohammed, Steven Rumph

2020

Janissa Abreu, Imtiaz Ahmed, Umar Ali, Jeffrey Almanzar, Madelin Arias Bueno, Leah Benitez, Josue Bueno, Richard Camacho, Luis Candelario, Albert Cedeno, Xujuan Chen, Chanyu Choung, Rene Clever, Joshua Cordero, Eduardo De La Cruz, Crystal Diaz-Maldonado, Geidy Dorviny Molina, Brianna Duvivier, Andy Estevez, Natalie Gonzalez, Rafael Grullon, Adem Gutic, Dhanraj Hiralal, Zahra Khan, Alix Leon, Jayke Lim, Milton Long, Nicholas Luckhai, Sabrina Mahjabin, Diego Marcelino, Emeka Mbazor, Nicholas Morales, Ba Nguyen, Jarin Niha, Andrew Ohakam, Soombe Wenne Edmond Ouedraogo, Wandy Peralta, Thi Phan, Umair Qureshi, Adelin Rivas, Bryant Saavedra, Jose Mariz Salva, Tony Sanchez, Marianny Soto Sanchez, Tyrell Whitfield, Emile Zounon, Mohammed Abdul Khaliq, Kelvin Abraham, Alan Abraja, Nafiz Ahmed, Tanbir Ahmed, Syed Ali, Juan Alvarez, Chris Kevin Andrade, Kazi Ashraf, Thierno Bah, Stephen Bailey, Anik Banerji, Nicholas Bonet, Aaron Bryant, Lizabeth Calderon, Sebastian Castano, Md Atik Chowdhury, Francisco Cuesta, Khadijah Dabo, Arnab Das, Hawa Dia, Aminata Diaby, Samuel Diaz, Ibrahim Doumbouya, Muhammad Drammeh, Muhamed Elmeetknany, Luis Enriquez, Lennen Escolastico, Christopher Estevez, Francis Figueroa, Wilper Germosen, Kevin Gonzalez, Cruz Gonzalez Garcia, Nathanael Gutierrez, Ariel Guzman, Jeremy Hernandez, Arman Hossain, Sabrina Hossain, Usama Ikram, Ibnat Iqbal, Saiful Islam, Dana Jackson, Juliana Jaku, Momodou Alpha Jallow, Marc Johnson, Brian Jones, Asifa Khan, Michael Lieberman, Omar Maldonado, Abdul Malik, Abdul Maradi, Nikolin Mele, Jayna Menard, Karimu Mohammed, Nicol Monegro, David Moran-Silva, Steve Morocho, Alvin Motieram, Boshra Mourin, Shahran Navid, Michael Nti, Yones Obaid, Anapatricia Olvera Medina, Ashadur Omith, Wilfy Ortega, Vanessa Ortiz, Bladimir Osorio, Lessie Quezada, Ayesha Qureshi, Sohel Rana, Francis Regalado, Christy Remigio, Sabier Richards, Jonathan Rivera, Jasir Romero, Mohammed Sameer-Uddin, Ramish Saqib, Alejandro Sarasky-Guzman, Stephen Sarfo, Kevin Savinovich, Peter Sayegh, Ala Sobhan, Sabri Sonmez, Abdur-Rasheed Syed, Iliya Tynan, Stephanie Vasquez, Ingrid Villanueva Garcia, Sonya Wong, Ashley Woods, Elvis Zelaya, Samuel Asiedu, Bryant Law, Falilou Ouro-Djobo

2021

Mohamed Abdulkarim, Alan Abraja, Kevin Alec Acquaah, Mathias Adikpon, Elija Agyapong, Kauser Ahmed, Allison Alava, Mohamed Albasuony, Syed Ali, Yasin Alkaifi, Chris Kevin Andrade, Shada Assaidi, Derick Ayimadu, Mohamed Bah, Shashwata Baidya, Anik Banerji, Leah Benitez, Richard Benitez, Nicholas Bonet, Miguel Bravo, Bryan Brugal, Josue Bueno, Robert Cabrera, Luis Candelario, Nicholas Cercos, Dylan Chan, Khairul Choudhury, Tahmina Choudhury, Chanyu Choung, Rene Clever, Khadijah Dabo, Aliyah Daisie, Hawa Dia, Karidia Diakite, Luisiana Dominguez, Brianna Duvivier, Muhamed Elmeetknany, Javier Enriquez, Lennen Escolastico, Christopher Esquivel, Andy Estevez, Jason Facey, Rahat Fahim, Istiaque Ferdous, Ilearys Fernandez, Angjelo Gjoni, Willy Gomez, Tavon Grant, Mccaulley Granville, Isabelle Greene, Josseline Guardado, Nelson Guzman, Ikram Hussain, Omar Jimenez, Brian Jones, Souleymane Kafaba, Andy Kan, Zahra Khan, Kiyatou Konate, Donya Lewis, Jayke Lim, Milton Long, Tony Lu, Nicholas Luckhai, Omar Maldonado, Abdul Malik, Jean Martinez, Emeka Mbazor, Nikolin Mele, Michelle Michaca, Mia Minaya, Abdul Mohith, Alejandra Morales, Raony Nadal, Jazmin Naranjo, Aisha Ndamuzanye, Jarin Niha, Joshua Nunoo, Andrew Ohakam, Tosin Olowolafe, Jose Patrone, Thi Phan, Kharla Ramirez, Sohel Rana, Joshua Reyes, Yismel Rodriguez, Mohammed Sameer-Uddin, Ramish Saqib, Alejandro Sarasky-Guzman, Zaid Shaikh, Charlton Smith, Jan Smith, Richard Smith, Ala Sobhan, Marianny Soto Sanchez, Bassira Souare, Sajia Sulaiman, Edrissa Sumareh, Abdur-Rahman Syed, Abdur-Rasheed Syed, Steven Tejada, Guevara Torres, Cong Tran, Blessing Uhunmwangho, Michael Waller, Sonya Wong, Ashley Woods, Edwin Acosta, Doral Elvy, Terrence Hayes, Md Islam, Yafira Martinez, Ana Monique Micieli, Luis Acosta, Umar Ali, Eric Buddy, Reynaldo Calcano, Braulio Calderon, Maria Chimbo, Jannatul Chowdhury, Ziaul Chowdhury, Nixa Cribas Banegas, Bryan De Los Santos, Aminata Diaby, Shuaib Essaedi, Alberto Garcia, Xinwei Ge, Izaac Gibson, Marcos Gonzalez, Cruz Gonzalez Garcia, Lisangel Herra, Enrik Hoxha, Usama Ikram, Kemar Johnson, Gerardo Juarez, Audom Khen, Andrew Lopez, William Mckinley, Alvin Motieram, Jose Nunez, Anapatricia Olvera Medina, Jonathan Rivera, Jose Rodriguez, Benjamin Rosado, Jonathan Rosario, Tenzing Sherpa, Sira Sissoko, Faraz Tariq, Iliya Tynan, Junior Valerio Perdomo, Emile Zounon, Wandy Peralta

2022

Mohamed Irsath Abdul Azeez, Paula Aidoo, Alex Amin, Ruhul Amin, Chris Kevin Andrade, Richmond Ankomah, Shada Assaidi, Simon Aytes, Leah Benitez, Miguel Bravo, Nolan Bruney, Eric Buddy, Raphael Camilo, Borana Ceci, Nicholas Cercos, Maria Chimbo, Chanyu Choung, Mahdi Chowdhury, Md Robiul Chowdhury, Aliyah Daisie, Bryan De Los Santos, Crystal Diaz-Maldonado, Shaid Dinar, Luisiana Dominguez, Brianna Duvivier, Muhamed Elmeetknany, Lennen Escolastico, Christopher Esquivel, Jason Facey, Alberto Garcia, Angjelo Gjoni, Karolin Gomez, Isabelle Greene, Nathanael Gutierrez, Enrik Hoxha, Usama Ikram, Md Islam, Richard Kabore, Mesuara Kaleziq, Zahra Khan, Audom Khen, Abigail Kwakye, Ana Karen Leonardo Garcia, Donya Lewis, Abdul Malik, Jean Martinez, Carlos Medina, Kaher Miah, Nicol Monegro, Mariano Moran, Truman Nguyen, Anapatricia Olvera Medina, Emily Portalatin-Mendez, Lessie Quezada, Bibata Rabba Idi, Jasson Rivera, Jose Rodriguez, Robert Romero, Yalitza Rosario, Mohammed Sameer-Uddin, Samantha Sanchez, Jan Smith, Mahajabin Sultana, Nadia Sultana, Abdur-Rasheed Syed, Wenkai Tan, Faraz Tariq, Aboulaye Tchakoura, Brianna Torres, James Bosch, Jeffrey Mathew, Sara Abdorab, Alan Abraja, Alphy Alvarez-Hernandez, Richard Ametepey, Justin Castronuovo, Zaker Choudhury, Ziaul Chowdhury, Ali Cruz, Joseph Deleon, Mamadou Diallo, Rahat Fahim, Ashley Febus, Josseline Guardado, Anthony Guidice, Ilyasu Jalloh, Charles James, MD Khan, Aly Kourouma, Benicio Ladines, Milan Limbu, Mia Minaya, Alexander Quezada, Andrianina David Raharijao, Jonathan Rosario, Janet Victoria, Aurelio Ndilou, Malachi Rogers, Bryant Saavedra

Lehman College Computer Science students have won numerous awards, both within the CUNY system and nationally. Additionally, some of our students have been invited into national and international honor societies, including Alpha Sigma Lambdas, the largest and oldest national honor society for students that aren't "traditional" students.

Section 4.5: Alum Employment and Graduate School

Many of our students have started successful careers after graduation. For example, the average starting salary for our 2019 graduates was $87,500, while the average starting salary for college graduates is $55,260, according to the National Association of Colleges and Employers. Our alumni are working in top companies all around the country, including Google, Goldman Sachs, Capital One, JP Morgan, General Motors, Twitter, Amazon, IBM, and more. In addition, many of our students have gone to top graduate schools such as Columbia University, Georgia Tech, and Carnegie Mellon University. Below are some examples of our recent undergraduate alumni who have shared news of their success with us:

- Elvis Boansi (B.S. 2014), Senior Software Developer at John Jay College of Criminal Justice
- Aisha Green (B.S. 2014), Software Engineer at Stae
- Dana Smith (B.S. 2015), Software Engineer, Goldman Sachs
- Daryl Shy (B.S. 2015), Software Engineer at Warner Bros. Discovery
- Pedro Ravel (B.S. 2016) Software Engineer - Twitter
- Mirna Znait (B.S. 2016), Vice President, Goldman Sachs
- Carson Brown (B.S. 2016), Development Lead – AI and Machine Learning, General Motors
- Jonathan Feigenson (B.S. 2017), Senior Application Support at City National Bank
- Manuel Emiliano (B.S. 2017), Software Engineer at JPMorgan Chase & Co.
- Min Zhang (B.S. 2017), Senior embedded software engineer at Amazon
- Angel Morales (B.S. 2018), Software Engineer FactSet, Formerly Full Stack Engineer at Floreo Labs | Tech Advisor at Lehman College | Web Development Instructor at Hunter College
- Chayanne Paulino (B.S. 2018), Software Developer at Kyndryl
- James Cervantes (B.S. 2018), Software EngineerSoftware Engineer
- Angie Williams (B.S. 2018), Asset Management Administrator at Alvarez & Marsal
- Keri Mallari (B.S. 2019), currently a Ph.D. student at the University of Washington
- Emeka Mbazor (B.S. 2019), Software Engineer at Apple
- Peter Farquharson (B.S. 2019), Financial Model Developer Freddie Mac
- Eddy Vittini (B.S. 2019), Software Engineer at Infor for Rhythm
- Mahboba Mim (B.S. 2019), Sr. Quality Service Associate Amazon
- Nehemias Miranda (B.S. 2019), Software Engineer (Android) at Weight Watchers
- Amado Suarez (B.S. 2019), Associate Software Engineer, JPMorgan Chase & Co.
- Maximiliano Martinez (B.S. 2019), IT Analyst at The Levy Group
- Patrick Abreu (B.S. 2019), Software Developer Intern at PlateRate
- Sumaiyah Syed (B.S. 2019), Engineering Manager, InfraOps, Floreo Labs
- Michael Nti (B.S. 2019), Software Engineer at American Express
- Pawel Misiak (B.S. 2019) Software Engineer II at Insider Intelligence
- Nathanael Gutierrez (B.S. 2020), currently a Ph.D. student at Georgia Tech
- Muhammad Asghar (B.S. 2020), currently an M.S. student at Carnegie Mellon University
- Alix Leon (B.S. in 2020), currently an M.A. student at Columbia University
- Jeffrey Almanzar (B.S. in 2020), Software Engineer, RubensteinTech
- Jayna Menard (B.S. in 2020), Software Engineer II at BestBuy
- Ikuseghan (Anna) Pryce (B.S. in 2020), currently an M.S. student at Fordham University

- Ibrahim Bocoum (B.S. in 2020), Software Engineer at Lockheed Martin, also pursuing an M.S. in Software Engineering at Penn State university
- Abraham Lara (B.S. in 2020), Software Developer at IBM
- Ezequiel Leon (B.S. in 2020), Cybersecurity Incident Response Specialist, Boeing
- Daniel Encarnacion (B.S. in 2020), Software Development Engineer, Zillow
- Mahmudul Alam (B.S. in 2020), System Engineer II Peraton
- Ralphie Perez (B.S. in 2020), Jr. Automation Engineer, Cigna
- Christopher Estevez (B.S. in 2020), currently an M.S. student at the New Jersey Institute of Technology
- Xheve Mulliqi (B.S. in 2020), Junior UI Developer at LzLabs
- Ba Nguyen (B.S. in 2020), Software Engineer at FPT Software
- Sean Feliciano (B.S. in 2020), IT Support Analyst, Technology Infrastructure at New York
- MetsJuan Morillo (B.S. in 2020), Corporate Operations Engineer at Google
- Xujuan Chen (B.S. 2021), currently an M.S. student at City College
- Ramish Saqib (B.S. 2021), Software Engineer ex-Twitter
- Sharmin Tamanna (B.S. 2021), Software Engineer at Target
- Lemuel Benitez Hernandez (B.S. 2021), Team Lead Verizon, Software Engineer at Tata Consultancy Services, and an M.S. student at Pace University
- Chris Espinal (B.S. 2021), Software Engineer at The Home Depot
- Juan Alvarez (B.S. 2021), L2 Information Technology AssociateL2 Information Technology Associate Valon
- Rene Clever (B.S. 2021), Frontend Engineer at Chartbeat
- Ifeatu Okaneme (B.S. 2021), Network Engineer at Verizon
- Waleed Malik (B.S. 2021), Software Engineer at Tata Consultancy Services
- Kwadwo Adusei-Poku (B.S. 2022), Software Engineer at Cadre
- Brianna Duvivier (B.S. 2022), Frontend Engineer at Pineapple Workshop
- Brandon Quezada (B.S. 2022), Software Engineer at Snap Inc
- Bora Ceci (B.S. 2022), Field Services Engineer at Alvarez & Marsal
- Zounon Emile Giovanni (B.S. 2022), currently an M.S. at Harvard Graduate School of Education
- Eric F. Aragundi (B.S. 2022), currently an M.S. at Columbia University
- Micheal Almandeel (B.S. 2022), currently an M.S. at the University of Delaware
- Daniel Mejia – American Express Software Engineer
- Patrica Cavereta – American Express Software Engineer
- Alex Flores – American Express Software Engineer

Chapter 5: Faculty and Staff

Section 5.1: Overview

The Department of Computer Science currently has 9 full-time regular faculty members, (See Tables 5-1 and 5-2), including 1 distinguished professor, 2 associate professors, 3 untenured assistant professors, (the 6 professors each hold a Ph.D.) and 3 lecturers. Additionally, there are 2 substitute lecturers. Substitute lecturer positions are temporary (upper limit of two years) fulltime positions. All professors are researches and all of them are also faculty members of the Ph.D. program at CUNY Graduate Center. The curriculum vitas of full-time faculty members are included in the appendix.

The Department of Computer Science adjunct faculty members are experienced instructors with most of them having strong industry background (Table 5-3). They form an indispensable part of the department's teaching force.

Table 5-1. Full-time Research Faculty Members in the Department of Computer Science

Name	Position	Office	Teaching	Research Area
Itai Feigenbaum	Assistant Professor	GI-137C	Discrete Math, Game Theory, Operations Research	Game Theory, Operations Research
Matthew Johnson	Associate Professor	GI-137D	Algorithms, Networking, Programming	Algorithm design in networking and sensor networks, power plants, batteries and environmental optimization, and security. Algorithmic Game Theory, Computational

				Geometry, Massive Datasets.
Brian Murphy	Associate Professor	GI-209	Programming, Video Game Programming, Data Structures, Algorithms, Parallel Algorithms, GPU Programing, Computer Organization, Networking, Web Development, Linear Algebra	Parallel Algorithms, Symbolic and Numerical Computations, Structured Matrix Computations, Design and Analysis of Algorithms, GPU Computing
Victor Pan	Distinguished Professor	GI-330	Cryptography, Algorithms, Parallel Algorithms, Graph Algorithms	Symbolic and Numerical Computations, Design and Analysis of Algorithms, Linear and Multilinear Algebra, Data Sparse Matrices, Polynomial Root-finding
Liang Zhao	Assistant Professor	GI-101A	Programming, Data Management and Analysis, Artificial Intelligence	Numerical Algebra, Machine Learning
Mingxian Zhong	Assistant Professor	GI-200B	Discrete Math, Graph Theory	Graph Theory, Scheduling, Algorithms, Combinatorial Optimization.

Table 5-2. Full-time Non-Research Faculty Members in the Department of Computer Science

Name	Position	Office	Teaching
Sameh Fakhouri**	Lecturer with C.C.E.	GI-102	Data Structures, Networking, iOS Programming
Steven Fulakeza	Lecturer with C.C.E.*	GI-232	Programming, Web Development, Operating Systems, Databases, Computer Organization, Networking
Geoffrey Kelly	Sub-lecturer	GI-137A	Programming, Computer Organization, Operating Systems, Cybersecurity
Emira Shano	Sub-lecturer	GI-100C	Programming, Web Development
Eva-Maria Sofianos	Lecturer with C.C.E.*	GI-304	Programming, Data Structures, Android Development, Web Development

Note: C.C.E. is essentially Tenure for Lecturers

* - C.C.E. will be in effect as of fall 2023

** - Retiring in 2023

Table 5-3. Part-Time Faculty Members in the Department of Computer Science

Name	Starting Semester	CS/CIS Courses Taught	Working Experience
Francis Cornacchiulo	Spring 2011	CMP 408	Technology Manager at Merrill Lynch
Bella Frankel	Spring 1994	CIS 234	NYC High School
Cameron Flowers	Fall 2019	CMP 464	Software Engineer at Floreo Labs and SwayBrand
Jeffery Gitlin	Spring 2015	CIS 166, CMP 168	Corporate Training at Bell Labs, Software Developer at several financial services
Priyadharsini Krishnakumar	Summer 2002	CIS 106, CMP 230, CIS 228, CIS 344, CIS 346	Associate Director of Web Services at Web MD
Ariel Maduro Gomez	Fall 2019	CIS 228	Continuing Education Instructor at Lehman College
Loric Madramootoo	Spring 2013	CMP 167, 168	HEOa at Lehman College
Alfredo Mirambeaux	Spring 2013	CIS 211	Software Engineer at IBM
Yanilda Peralta Ramos	Fall 2017	CIS 166, 331, 344, CMP 167	Instructor at CUNY Queens College, Data analyst intern at AOL
Safiul Roni	Fall 2011	CIS 211	Lehman IT
John Solano	Fall 2022	CMP 420	Software Engineer at Soft Turbine
Ronen Samson	Fall 2005	CMP 167, 168, 232, CIS 166	Instructor at CUNY City College

Jehan Sanmugaraja	Fall 2015	CIS 345, CMP 405	Global Technical Lead at IBM
Hendrix Tavarez	Fall 2017	CMP 167, 168, 326, 334, 405, CIS 211	Software Engineer at IBM
Zihni Ugurbil	Spring 2017	CIS331,CIS212,CMP326,CIS345	Software Engineer at IBM, Visa, KLM, Worldspan, and Turkcell
Aron Wolinetz	Spring 2011	CIS211, CIS244	Lehman College

Table 5-4. Staff Members in the Department of Computer Science

Name	**Position**	**Office**
Ingrid Rosario	College Assistant	GI-211C
Marvin Florin*	HEOA**	GI-218
Loric Madramootoo	HEOa***	GI-222B
Max Hill*****	HEOa***	GI-137C
Swathy Gopal*****	aHEO****	GI-137C

* Affiliated with the Math department, but serves as an IT technician for the CS department.

** Associate Higher Education Officer

*** Assistant Higher Education Officer

**** Assistant to Higher Education Officer

***** Hired this semester as part of the CUNY Inclusive Economy Initiative

Table 5-5. Former Regular Fulltime Faculty (21st Century Only)

Name	Memo	Position
Charles Berger*	Retired in 2011	Professor
Anthony Cocchi	Retired in 2017	Distinguished Lecturer
Melvin Fitting*	Retired in 2012	Professor
Nancy Griffeth*	Retired in 2015	Professor
Yves Jean	Left in 2012	Associate Professor
Gwang Jung	Retired in 2022	Associate Professor
Julian Laderman*	Retired in 2007	Associate Professor
Javier Lopez	Left in 2015	Assistant Professor
Megan Owen	Went with Math in 2017 Split	Associate Professor
Esther Phillips*	Retired in 2006	Professor
Ryhs Rosholt	Passed away in 2013	Lecturer
Robert Schneider*	Retired in 2018	Professor
Katherine St. John	Left in 2017	Professor
Boris Yamrom	Left in 2006	Associate Professor

* Professor Emeritus

Note: Several mathematicians regularly taught Computer Science and Computer Information Systems courses and CMP 232 was almost exclusively taught by mathematics faculty prior to the department split. (See Appendix I for more details.)

Section 5.2: Faculty Research Highlights

Itai Feigenbaum

Dr. Feigenbaum's research interests are in the areas of game theory, mechanism and market design, combinatorial optimization, machine learning and data science. Much of Dr. Feigenbaum's research involves the design of optimization algorithms for multi-agent systems, where agents behave in a self-interested manner, which is often not aligned with the optimization objective. Some examples of Dr. Feigenbaum's work include:

1. *Strategyproofness in kidney exchange with cancellations (WINE 2022):* Patients requiring kidney transplant may have proxy donors: people who want to donate a kidney to the patient, but cannot due to medical incompatibility. However, patients can swap proxy donors, so that each swapping patient ends up with a compatible donor. Some patients, called overloaded, have multiple proxy donors. A matching is a collection of planned transplants resulting from swaps: we consider both balanced matchings, which restrict overloaded patients to swapping just one of their proxy donors, and unbalanced matchings which do not. In practice, many planned transplants get canceled, and so we want matchings to maximize the expected number of actually executed transplants. Maximization of executed transplants introduces perverse incentives for overloaded patients, who can increase the probability they receive a kidney by hiding some of their proxy donors. We design the SuperGreedy Algorithm, which provably incentivizes patients to fully reveal their proxy donors. When cancellation probabilities are uniformly constant, we prove that SuperGreedy O(1)-approximates the maximum number of executed transplants; we also implement SuperGreedy and show via simulation that it performs well on realistic data.
2. *Dynamic matching in school choice: Efficient seat reassignment after late cancellations (with Irene Lo, Yash Kanoria, and Jay Sethuraman, Management Science 2020)*: In the school choice market, where scarce public school seats are assigned to students, a key operational issue is how to reassign seats that are vacated after an initial round of centralized assignment. Practical solutions to the reassignment problem must be simple to implement, truthful, and efficient while

also alleviating costly student movement between schools. We propose and axiomatically justify a class of reassignment mechanisms, the permuted lottery deferred acceptance (PLDA) mechanisms. Our mechanisms generalize the commonly used deferred acceptance (DA) school choice mechanism to a two-round setting and retain its desirable incentive and efficiency properties. School choice systems typically run DA with a lottery number assigned to each student to break ties in school priorities. We show that under natural conditions on demand, the second-round tie-breaking lottery can be correlated arbitrarily with that of the first round without affecting allocative welfare and that reversing the lottery order between rounds minimizes reassignment among all PLDA mechanisms. Empirical investigations based on data from New York City high school admissions support our theoretical findings.

3. *Strategic facility location problems with linear single-dipped and single-peaked preferences (JAAMAS 2020):* We consider the design of mechanisms for locating facilities on an interval. There are multiple agents on the interval, each receiving a utility determined by their distances to the facilities. The objectives considered are maximization of social welfare (sum of utilities) and egalitarian welfare (minimum utility). Agents can misreport their locations, and so we require the mechanisms to be strategyproof—no agent should be able to benefit from misreporting; subject to strategyproofness, we attempt to design mechanisms that are approximately optimal (have small worst-case approximation ratios). The novelty of our work is the consideration of models in which single-dipped and single-peaked preferences exist simultaneously. We consider two models. In the first model, there is a single facility, and agents may disagree about its nature: some agents prefer to be near the facility, while others prefer to be far from it. In the second model, there are two facilities: a desirable facility that all agents want near, and an undesirable facility that all agents want far. We design a variety of approximately optimal strategyproof mechanisms for both models, and prove several lower bounds as well. For the social welfare objective, we provide bestpossible deterministic strategyproof mechanisms in the first model and the second model. We then provide improved randomized strategyproof mechanisms for each model, as well as a non-tight lower bound on the worst-case approximation ratio attainable by such mechanisms for the first model. For the egalitarian welfare objective, we provide a lower bound on randomized strategyproof mechanisms for the first model, as well as

an optimal (non-approximate) strategyproof mechanism for the second model. All of our mechanisms are also group strategyproof: no coalition of agents can unanimously benefit from misreporting.

4. *Approximately optimal mechanisms for strategyproof facility location: Minimizing L_p norm of costs (with Jay Sethuraman and Chun Ye, Mathematics of Operations Research 2017):* This paper is concerned with the problem of locating a facility on a line in the presence of strategic agents, also located on that line. Each agent incurs a cost equal to her distance to the facility whereas the planner wishes to minimize the L_p norm of the vector of agent costs. The location of each agent is only privately known, and the goal is to design a strategyproof mechanism that approximates the optimal cost well. It is shown that the median mechanism provides a $2^{1-1/p}$ approximation ratio, and that this is the optimal approximation ratio among all deterministic strategyproof mechanisms. For randomized mechanisms, two results are shown: First, for any integer p larger than 2, no mechanism—from a rather large class of randomized mechanisms—has an approximation ratio better than that of the median mechanism. This is in contrast to the case of $p = 2$ and $p = \infty$ where a randomized mechanism provably helps improve the worst case approximation ratio. Second, for the case of 2 agents, the Left-Right-Middle (LRM) mechanism, first designed by Procaccia and Tennenholtz for the special case of infinity norm, provides the optimal approximation ratio among all randomized mechanisms.

5. *Selfish knapsack (With Matthew P. Johnson, AAAI 2017):* We consider a strategic variant of the knapsack problem: the items are owned by agents, and agents can misrepresent their sets of items---either by hiding items (understating), or by reporting fake ones (overstating). Each agent's utility equals the total value of her items included in the knapsack. We wish to maximize social welfare, and attempt to design mechanisms that lead to small worst-case approximation ratios at equilibrium. We provide a randomized mechanism with attractive strategic properties: it has a price of anarchy of 2 for Bayes-Nash and coarse correlated equilibria. For overstating-only agents, it becomes strategyproof, and has a matching lower bound. For the case of two understating-only agents, we provide a specialized randomized strategyproof 1.522-approximate mechanism, and a lower bound of 1.09. When all agents but one are honest, we provide a deterministic strategyproof 1.618-approximate mechanism with a matching lower bound. The latter two mechanisms are also useful in problems beyond the one in consideration.

Matthew Johnson

Dr. Johnson's research is in algorithms: on applying efficient algorithms to (mostly) discrete optimization problems, in such a way that provable things can be said about what results, and when—something like a kind of quantitative epistemology: what can we know, and when can we know it?

Approximation Algorithms. Dr. Johnson's approach focuses on continuing to require a worst-case guarantee on solution quality, but relaxes that guarantee's desideratum from optimal to approximate. More precisely, the aim is to design (polynomial-time) algorithms with the property that the cost (or value) of the solution produced will always be within some multiplicative factor c of optimal. Any such c will thus be > 1 for minimization problems and < 1 for maximization, and in both cases one would like for c to be as close as possible to 1 (which would mean optimal). How close one can get the guarantee to 1 tells something about the difficulty of the problem, relative to running time. That is, it tells that one can obtain a polynomial running time in exchange for allowing degradation in solution cost by as much as factor c; or, it upper-bounds how much one must sacrifice solution quality in order to meet a polynomial-time limit.

Other Handicaps. Stepping back a little, the study of approximation algorithms can be seen as just one instantiation of the following research schema: studying what minimum "price" must be paid (in worst-case harm done to solution quality) for imposing a restriction or handicap on the class of permitted candidate algorithms, and approaching that price by trying to design algorithms that will perform nearly as well as one could've done without that handicap. Besides (a) polynomial-time vs. not, other prominent examples of such paid-for handicaps include (b) distributed vs. centralized, (c) online vs. offline, and (d) dependence on the choices of selfish players (in Nash equilibrium) vs. having full control. Constrained algorithms of these four types are judged by analogous performance measures: approximation ratio (for a & b), competitive ratio (for c), and price of anarchy (for d). Much of Dr. Johnson's work concerns the design and analysis of handicapped algorithms that (more or less) overcome their handicaps (...or else proving that no such algorithm could). His research interests focus on when strong results or interesting analysis derive from dead-simple algorithms, especially when they're not just point solutions, but in some sense apply universally, unifying a range of problem settings, e.g.:

• Red-blue partitioned pairs algorithm in "Red-blue-partitioned MST, TSP, and matching" In CCCG. It provides the best-known approximation guarantees for both "2-MST" and (slightly altered) "2-TSP," for both min-sum and min-max objectives,

• Multicast algorithm in "Minimum-cost network-wide broadcast over reliable MAC-layer multicast" IEEE Transactions on Mobile Computing. It provides a smoothly varying c-approximation, where c goes from O(log(n)) down to 1 (optimal) as a cost function parameter goes from 0 to 1,

• Secluded path DP in "Secluded path via shortest path". In SIROCCO. It provides a smoothly varying c-approximation, where c goes from 1 (optimal) up to $\sqrt{\Delta} + 1$ as the algorithm's "memory" parameter goes from Δ down to 0, where it converges to the same static algorithm that has approximation guarantees on unit-disk and planar graphs, and

• Achieving O(log(n))-approximations for the seemingly unrelated problems of sensor duty cycle scheduling in "Joint sensing duty cycle scheduling for heterogeneous coverage guarantee" In IEEE INFOCOM, and simultaneously-fresh data transmission scheduling in "Gathering information in sensor networks for synchronized freshness" In IEEE SECON, both by interpreting the problem as an implicit (exponentially large) set system, and running set cover greedy (in polynomial time).

Brian Murphy

Dr. Murphy's research interest lies in the area of algebraic and numerical computations, with a focus on computations involving polynomials and structured matrices. His efforts are geared toward finding faster and/or more accurate methods for computing the results for calculations in these areas with considerations of parallelism and new affordable hardware options.

Dr. Murphy's research highlights:

The papers "Linking the TPR1, DPR1 and Arrow-Head Matrix Structures" and "Eigen-solving via Reduction to DPR1 matrices" present some novel techniques for the transformation of matrix structures, namely of TPR1 structure into DPR1 and arrowhead matrix structures (here "TPR1" and "DPR1" are the acronyms for "Triangular+rank-1" and "Diagonal+rank-1", respectively). These nontrivial transformations are motivated by application to the matrix eigenvalue problem, which is one of the most fundamental problems of modern matrix computation. Efficient similarity transformations are known that transform a general matrix into a TPR1 matrix. A further similarity transformation into a DPR1 or an arrowhead matrix is highly desirable. This reduces the eigenvalue problem to the cases of the latter classes of highly structured matrices, for which the eigenproblem can be solved readily by recent highly efficient algorithms using quadratic rather than cubic time. A number of these advanced algorithms are proposed and analyzed in Chapter 14, "Root-finding with Eigensolving" in Symbolic Numerical Computation, Birkhäuser, Basel/Boston (2007). The algorithms accelerate the known methods for univariate polynomial root-finding, which is a venerable classical subject and still important for modern computations in algebra and geometry as well as in Signal and Image Processing and financial mathematics. The reduction to matrix computations is beneficial because techniques for the numerical computation of matrix eigenvalues are well developed. The proposed algorithms are also efficient in determining the solution of the secular equation, which is important for mechanics.

The paper "Real Root-Finding" in Proceedings of the Third International Workshop on Symbolic--Numeric Computation is devoted to the important special case of univariate polynomial root-finding where only the real roots are sought and where the number of real roots, r, is only a small percentage of the n complex roots. For this case, the paper develops some novel techniques, combines them with advanced known algorithms, and thus achieves acceleration of the known methods by a factor of n/r. These algorithms have been revisited and further advanced in the paper “On the Boolean Complexity of the Real Root Refinement” by V.Y. Pan and E. Tsigaridas, in Proc. International Symp. on Symbolic and Algebraic Computation (ISSAC 2013), (M. Kauers ed.), Boston, Massachusetts, June 2013, which achieves a nearly optimal solution of this highly important computational problem.

The papers “Nearly Optimal Symbolic-Numerical Algorithms for Structured Integer Matrices and Polynomials” in Proc. International Symposium on Symbolic-Numerical Computations, ACM Press, New York and “Unified Nearly Optimal Algorithms for Structured Matrices” in Operator Theory: Advances and Applications, Birkhauser, Basel proposed two distinct methods for solving structured linear systems of equations. The first introduced generalized Hensel’s lifting. The second combined the MBA algorithm by Morf 1980 and Bitmead & Anderson 1980, adjusted to symbolic computations, with the Chinese remainder algorithm. In both cases the proposed algorithms are unified for all matrices having displacement structure. This includes the structures of Toeplitz, Hankel, Vandermonde and Cauchy types. The algorithms are nearly optimal for matrices with all of these structures and for the associated fundamental computations with polynomials, such as computation of the gcds, lcms, and Pade approximation as well as for the Berlekamp-Massey problem for linear recurrence computations.

In the paper “Acceleration of the Inversion of Triangular Toeplitz Matrices and Polynomial Division” (No coauthors), in Lecture Notes in Computer Science Vol. 6885, pg. 321-332 (2011) I study the special but highly important problems of the Inversion of Triangular Toeplitz Matrices and Polynomial Division which are closely linked. Algorithms for the general and banded cases are presented which improve upon the asymptotic record bounds for arithmetic operations. The algorithms lend themselves to parallel acceleration due to an underlying FFT kernel. However, where the system is banded and the bandwidth is much smaller than system size the number of arithmetic operations is dramatically lowered and the degree of actionable parallelism rises.

The paper "Solving Tridiagonal Systems on a GPU" (No coauthors) in Proceedings of the IEEE 20th International Conference on High Performance Computing (2013), presents a parallel tridiagonal solver for NVIDIA GPUs under CUDA. There have been many efforts to implement fast tridiagonal solvers that can take advantage of the massive parallelism of such GPUs in recent years. Not unlike other attempts at accelerating this computation on a GPU, my effort foregoes ground up design of a new algorithm for a careful matching of known techniques to the idiosyncrasies of the GPU. My implementation is a hybrid of the LU decomposition, cyclic reduction, and parallel cyclic reduction. Unlike other tridiagonal solvers this implementation throws the CPU into the unusual role of coprocessor for the GPU computation. The LU decomposition is performed on the CPU via the serial Thomas Algorithm so that it overlaps a required asynchronous data transfer from the CPU to the GPU thereby "hiding" the latency of the computation. On the GPU the acceleration as compared to other GPU based solvers is attributable to a significant lowering of the communication effort during the computation phase and the ability to pack a larger segment of the linear system into the register files of each of the GPU's thread processors thanks to the essentially free preprocessing on the CPU.

The papers "Embarrassingly Parallel Butterflies Solve Diagonally Dominant Tridiagonal Toeplitz Systems" (No coauthors) in Proceedings of the 2014 International Conference on Parallel and Distributed Processing Techniques and Applications (2014) and "Butterflies Solve Bidiagonal Toeplitz Systems" (with student coauthors) in Proceedings of the 2015 International Conference on Parallel and Distributed Processing Techniques and Applications (2015), present new algorithms for the parallel acceleration of the respective computations on FFT processors. These systems are at the heart of problems as diverse as cubic spline and B-spline curve fitting, preconditioning for iterative linear solvers, computation of photon statistics in lasers, computational fluid dynamic, solution of multidimensional diffusion computations, solution of neuron models by domain decomposition, and much more. Both papers present algorithms based on FFT kernels. While the FFT is quite effectively parallelized, the reality is that parallel FFT processors are most efficient when the size of the input is relatively small and of course limiting communication between processing units is a top priority in the design of parallel algorithms in general. In consideration of these facts the algorithms presented in these two papers exploit a rectangular block matrix decomposition that both maximizes the utility of the FFT processes to lower the operation count and isolates computations for each block to eliminate the need for communication between partitions and therefore between cooperating FFT processors at any stage. An array of small FFT processors can then be applied in an embarrassingly parallel manner to tackle a large system. Of great significance, the number of parallel steps is a function of the two parameters defining the matrix entries and not the system size. Further, unlike other parallel algorithms that solve such systems these algorithms are fine-grained and yet do not require agglomeration to combat communication latencies while the algorithms are eminently scalable. They can take advantage of readily available hardware to match processor count to input size thereby maximizing parallelism.

Victor Pan

Dr. Pan solved a number of long-standing problems in Computational Math and Computer Science, introduced new insights and novel methods, revealed unexpected links among some seemingly distant

subjects, and proposed new research directions and new areas of study. His research helped create fields of Algebraic Computational Complexity, Symbolic-Numerical Computing, and Algebraic Multigrid and establishing synergistic links among Symbolic Computations, Numerical Computations, Theoretical Computer Science, and Numerical Linear Algebra.

Some concepts and definitions introduced in Dr. Pan's papers, techniques and insights are widely adopted and commonly used, sometimes as folklore. Google Scholar and DBLP list his four books (1623+LXXIV pages overall), over 20 surveys in journals and book chapters, over 180 research articles in journals, over 90 in refereed conference proceedings, and over 12,000 citations of his work. His book with Dario Bini "Polynomial and Matrix Computations", Birkh¨auser, Boston, 1994, over 700 Google Scholar citations, covers its title subjects in depth and presents a wealth of new research results. Reviewers in ACM SIGACT News (June 1995) and ACM SIGACT Bulletin (Sept. 1996) found it "outstanding" and recommended it as "must-have book" and "a reference for a graduate course in symbolic computations or computer algebra." Polynomial and matrix computations are also the subjects of my three other books, a dozen of my book chapters, and most of my research and survey papers, including my long surveys in SIAM Reviews in 1984, 1992, and 1997.

Dr. Pan's research highlights:

I. Univariate polynomial root-finding has been the central problem of Mathematics and Computational Mathematics throughout four millennia and is still important for the theory and practice of computing. Dr. Pan's algorithms in STOC 1995, CAMWA 1996 and J. Symbolic Comp. 2002 run in record low Boolean time, which is optimal up to polylogarithmic factors. The algorithms approximate all roots almost as fast as one can read input coefficients. His algorithm in J. of Complexity 2000, extending Weyl 1924, Henrici 1974 and Renegar 1987, was in turn extended into another nearly optimal root-finder by Becker et al. in J. Symbolic Comp. 2018. Already its implementation reported in ICMS 2018 by R. Imbach, himself, and C. Yap and supported by NSF Grant AF: Medium: Collaborative Research: "Numerical and Algebraic Differential Equations", CCF-1563942, $608,205, 2016-2021 (joint with A. Ovchinnikov and Yap) was fastest known for root-finding in a complex disc with fewer roots, but in CASC 2019 and arxiv:1805.12042 Dr. Pan proposed significant improvements of this algorithm as well as numerical multipoint polynomial evaluation, which is fundamental for polynomial root-finding and for many other subjects of numerical algebraic computations. His new root-finders efficiently handle polynomials defined by subroutines for their evaluation rather than by their coefficients, which is important novel feature for both theory and practice. His other polynomial root-finders have been well-recognized as well. A few examples are his papers in SICOMP 1995 by myself, in J. of Complexity 1996 and CAMWA 2004 with D. Bini, in ETNA 2004 and Numer. Math. 2005 with D. Bini and L. Gemignani, in ISSAC 2013 and J. Symbolic Comp. 2016 jointly with E.P. Tsigaridas, and in Symbolic-Numerical issue of TCS 2017 with L. Zhao. This includes nearly optimal real root-finding, nontrivial use of matrix methods, and technical novelties commonly used afterwards.

II. Since 2010 Dr. Pan published in LAA a number of algorithms for the solution of nonsingular and homogeneous singular linear systems of equations with randomized preprocessing. E.g., in LAA

2013, 2015 and 2017. He studied randomized multiplicative preprocessing as a means of numerical stabilization of Gaussian elimination instead of communication intensive pivoting. He proved that this produces accurate solution with a high probability for random inputs and proposed new classes of efficient sparse and structured multipliers.

III. Since 2016 Dr. Pan has extended the study of linear systems of equations to the hot research subjects of Low Rank Approximation (LRA) of a matrix, which is highly important for numerical linear and multilinear algebra and Big Data mining and analysis. Realistically one can only access a tiny fraction of all entries of input matrices representing such data, e.g., unfolding matrices of multidimensional tensors, and so one must devise sublinear cost algorithms, which use much fewer flops and memory cells than the input matrix has entries. The known algorithms, however, use superlinear time because no deterministic or randomized sublinear cost algorithm can output accurate LRA for the worst case input and even for a specified small family of matrices that admit rank-1 approximation. In his papers in Procs. of MACIS 2019 and arxiv 1906.04223, 1906.04327, 1906.04929, and 1907.10481, submitted for publication, Dr. Pan, however, proposed and formally supported a remedy: achieve sublinear cost performance by means of trivializing the bottleneck stages of some known algorithms: pre-process input matrices with new sparse multipliers rather than known random dense (Gaussian, SRHT, or SRFT) ones or compute LRA by means of random sampling under trivial choice of sampling probabilities (called leverage scores), thus avoiding their costly computations. He proved that the resulting sublinear cost algorithms still output reasonably accurate LRA whp for a random input matrix that admits LRA. He proposed deterministic sublinear cost LRA algorithms for a Symmetric Positive Semidefinite matrix. Furthermore he proposed novel extension of the popular sketching (aka subspace sampling) randomized LRA algorithms. Originally they output LRA at superlinear cost, but his recursive extension of these algorithms converges at sublinear cost under some mild assumptions about an input. Dr. Pan also applies that algorithm to iterative refinement of a crude but reasonably close LRA, which is a task of independent importance. The results of our extensive tests with both synthetic and real world inputs (some from SIAM Review paper of 2011 by Halko et al) are in good accordance with that formal study of LRA. This progress implies acceleration of various known algorithms linked to LRA4 and should motivate further effort by researchers towards sublinear cost matrix computations and synergistic combination of the techniques developed in the Applied Linear Algebra and Computer Science communities. Dr. Pan's work on LRA was largely motivated by its expected contributions to Deep Learning Networks, which is supported by NSF Grant AitF CCF 1733834, $448,086, 2017-2021 (joint with Bo Yuan and Xue Lin).

Liang Zhao

Dr. Zhao has a broad research interest in numerical algebra, algorithm design, and artificial intelligence. He proposed a novel method of constructing neural network models with their weight matrices/tensors following explicit patterns. This method can greatly reduce the number of parameters required for the model, and low-complexity algorithms can be applied to its structured weight matrices so that the training time and inference time for such models can be significantly improved. He proposed the method and showed the approximation potential of such neural network structures in paper "Theoretical Properties for Neural Networks with Weight Matrices of Low Displacement Rank" in ICML, and illustrated that it is a superior solution to other structural remedies to the over-parameterization problem with deep neural networks in paper "On the Universal Approximation Property and Equivalence of Stochastic Computing-based Neural Networks and Binary Neural Networks" in AAAI. Besides providing theoretical guarantees, Dr. Zhao studied a particular implementation of structured neural networks with circulant matrices, and indicated that this model can overcome the huge demand of computation resources and weight storage of typical neural networks, while providing similar prediction accuracy. The circulant structure and models can be either trained from scratch or re-trained from a pre-trained non-circulant model, thereby making it very flexible for different training environments. Through extensive experiments, such a strong structure-imposing approach is proved to be able to substantially reduce the number of parameters of convolutional layers and enable significant saving of computational cost by using fast multiplication of the circulant tensor. These works attracted a lot of interests in the field of edge computing, and led to the design of several deep learning acceleration units. Besides accelerating the computations related to neural networks, Dr. Zhao developed several other methods in this area. For example, he proposed a hybrid method for video object detection that combined techniques from deep learning and particle filtering, a novel transfer learning scheme for data imputation that is adaptable to new data formats, and a speech generation method using topology-enhanced generative adversarial networks.

Dr. Zhao is also interested in designing fast randomized algorithms for matrix computation. Randomized algorithms can provide close approximations with high probability to the exact results of matrix computation tasks such as multiplication and factorization, while using much less time than the theoretical minimum cost required by the exact calculation. In the age of big data, the data matrices are often too large to be stored in a single computer, and thus it is often impossible to obtain fully accurate computation results within the limitation of the available computation resources. Luckily, it is also often unnecessary to ask for results that are accurate to the last digit, where a good approximation is just as useful. Dr. Zhao has developed several fast randomized matrix computation algorithms, among which include a fast algorithm for the general matrix inverse in "An efficient computation of generalized inverse of a matrix" in Applied Mathematics and Computation, a numerically safe Gaussian elimination algorithm that does not require row/column pivoting in "Numerically Safe Gaussian Elimination with No Pivoting'' in Linear Algebra and Its Applications, and an efficient algorithm for approximating a matrix with a low-rank matrix in "Primitive and Cynical Low Rank Approximation" in Preprocessing and Extensions.

Mingxian Zhong

Dr. Zhong's research interests lie in the areas of graph theory and theoretical computer science. His research interests in graph theory focus on analyzing the structural properties of H-free graphs and how those properties help to make problems that are hard on general graphs easier on H-free graphs, for some special H. Dr. Zhong was mostly working on the problems of coloring graphs with forbidden induced subgraphs and the problems of finding the obstructions for 3-coloring and list-3-coloring H-free graphs. Dr. Zhong's research in theoretical computer science focuses on the design and analysis of combinatorial algorithms and he was mostly working on scheduling problems.

Coloring graphs with forbidden induced subgraphs. A coloring of a graph is a function that assigns each vertex a color such that no two adjacent vertices share the same color. Many problems in practical areas, such as register allocation, scheduling and pattern matching, can be modeled as a graph coloring problem. Dr. Zhong gave a polynomial-time algorithm to determine whether there exists a 3-coloring for a P7-free graph, which answers positively to a question first posed in 2002 by Randerath and had been open for more than one decade. Dr. Zhong also proved that the problem of 4-coloring P6-free graphs is polynomial-time solvable in paper "Four-coloring P6-free graphs", and gave an explicit construction for all triangle-free P6-free graphs in paper "Triangle-free graphs with no six-vertex induced path".

Obstructions for coloring and list-coloring H-free graphs. Dr. Zhong worked on the case of P6-free graphs and proved that there are exactly 24 4-critical P6-free graphs in paper "Obstructions for three coloring graphs without induced paths on six vertices". He gave a dichotomy theorem of the problem: for a connected H, there are finitely many 4-critical H-free graphs if and only if H is a subgraph of P6, and extended the above dichotomy theorem by removing the requirement that H is connected. In paper " Obstructions for three-coloring and list three-coloring H-free graphs", Dr. Zhong proved that there are only finitely many H-free minimal list-obstructions if and only if H is an induced subgraph of P6 or P4 + kP1 for some k ∈ N.

Scheduling with uncertainty on the number of machines. Often in a scheduling problem, there is uncertainty about the jobs to be processed. The issue of uncertainty regarding the machines has been much less studied. Motivated by the need to understand how to make scheduling decisions without knowing how many machines one will have, Dr. Zhong considered a different notion of uncertainty – a scenario in which one doesn't know how many machines to have, but one still has to commit (partially) to a schedule by making significant decisions about partitioning the jobs before knowing the number of machines. In order to evaluate algorithms in such an environment, Dr. Zhong introduced the idea of an α-robust algorithm, one which is guaranteed to return a schedule on any number m of machines that is within an α factor of the optimal schedule on m machines, where the optimum is not subject to the restriction that the sets cannot be separated. Under such environment, we give a (5/3 + ϵ)-robust algorithm for scheduling on parallel machines to minimize makespan, and show a lower bound 4/3 in paper "Scheduling When You Don't Know the Number of Machines". For the special case when the jobs are infinitesimal, Dr. Zhong gave a 1.233-robust algorithm with an asymptotic lower bound of 1.207. The previous results were subject to the objective of minimizing makespan on identical parallel machines.

Faculty Grants (2012 - 2022):

CUNY 2X and Tech Talent Pipeline grants (~2 million)

- Itai Feigenbaum, Six PSC-CUNY Research Awards, $4530 each (2017-2022)
- Steven Fulakeza (Co-Principal), "CUNY - NYC Tech Talent Pipeline Career Success Course Innovation Grant," Sponsored by CUNY2X, Local, $10,000.00. (August 25, 2022 - January 31, 2023).
- Steven Fulakeza (Co-Principal), Brian Murphy (Co-Principal), Eva-Maria Sofianos (Co-Principal), Sameh Fakhouri (Co-Principal), Rafael Gonzalez (Co-Principal), "Center for Inclusive Computing," Sponsored by Northeastern University, Other, $60,000.00. (February 1, 2021 - December 31, 2022).
- Steven Fulakeza (Co-Principal), Sameh Fakhouri (Co-Principal), Eva-Maria Sofianos, "CUNY - NYC Tech Talent Pipeline Career Success Course Innovation Grant," Sponsored by CUNY2X, Local, $10,000.00. (March 24, 2022 - June 30, 2022).
- Steven Fulakeza, "CUNY - NYC Tech Talent Pipeline Career Success Course Innovation Grant," Sponsored by CUNY2X, Local, $10,000.00. (March 24, 2022 - June 30, 2022).
- Matthew Johnson, PSC-CUNY Research Award 69778-00 47, $6,000, 2016–2017
- Matthew Johnson, CUNY Junior Faculty Research Award in Science and Engineering (sponsored by the Sloan Foundation), $50,000, 2016
- Matthew Johnson, NSF grant "INSPIRE: Optimization Algorithms for Regional Thermoelectric Power Generation with Nonlinear Interference" (CCF-1547205), $316,000, 2015-2018
- Matthew Johnson, CUNY ASRC Joint Seed Program Grant, $10,000, 2015-2016
- Matthew Johnson, PSC-CUNY Research Award 68818-00 46, $6,000, 2015–2016
- Matthew Johnson, Research in the Classroom Idea Grant, $7,500, 2014–2015

- Matthew Johnson, CUNY Collaborative Incentive Research Grant (CIRG 21) 2153 (with co-PI Felisa Vásquez-Abad), $30,000, 2014–2015
- Matthew Johnson, PSC-CUNY Research Award 67665-00 45, $6,000, 2014–2015
- Matthew Johnson, NSF REU supplement to grant DUE-1060598, $20,000, 2013-2014
- Victor Pan, NSF Grant (individual) CCF1116736, Novel Methods for Fundamental Matrix and Polynomial Computations", $350,000 (from 8/1/2011 to 12/31/2016)
- Victor Pan, NSF Grant (joint: A. Ovchinnikov PI, V. Pan co-PI, and C. Yap co-PI) CCF - 1563942 AF: Medium: Collaborative Research: Numerical Algebraic Differential Equations, $608,205 (from 7/1/2016 to 6/30/2021)
- Victor Pan, NSF Grant (joint: Bo Yuan PI, V. Pan co-PI, and Xue Lin co-PI) AitF, CCF-1733834, Medium: Collaborative Research: A Framework of Simultaneous Acceler- ation and Storage Reduction on Deep Neural Networks Using Structured Matrices, $448,086.00 (from 9/15/2017 to 12/31/2021)
- Victor Pan, PSC CUNY AWARD 69813 00 48: New Progress in Matrix and Polynomial Compu- tations, $11,998.92 (from July 1, 2017 to December 31, 2018)
- Victor Pan, PSC CUNY AWARD 62797-00 50: $11,999.74 (from June 30, 2019 to December 31, 2020)
- Victor Pan, PSC CUNY AWARD 63677-00 51: $5,999.87 (from July 1, 2020 to June 30, 2021)
- Victor Pan, PSC CUNY AWARD 6886200 46: Advancing Matrix and Polynomial Computations", $11,998, 7/1/201512/31/2016
- Victor Pan, PSC CUNY AWARD 6769900 45:Advancing Matrix and Polynomial Computations", $6,000, 6/30/2014 7/1/2015
- Victor Pan, PSC CUNY AWARD 657920043: Matrix and Polynomial Algorithms", $11,998.92, 6/30/2012 7/1/2013
- Victor Pan, PSC CUNY AWARD 645120042: Matrix and Polynomial Computations", $6,000, 6/30/2011 7/1/2012
- Eva-Maria Sofianos (Co-Principal), Lawrence Fauntleroy (Co-Principal), Brian Murphy (Co-Principal), Rafael Gonzalez (Supporting), "COMPUTER SCIENCE TRANSFER STUDENTS INITIATIVE Strengthening Confidence through Data Structures & Algorithms," Sponsored by CUNY 2x Tech, State, $20,000.00. (July 1, 2021 - December 31, 2021).
- Eva-Maria Sofianos, Center for Inclusive Computing Northeastern University $60,000.00. (January 28, 2021 - December 25, 2022)
- Eva-Maria Sofianos, NYC TTP CSCI Grant $10,000.00. (December 31, 2021 - June 30, 2022)
- Eva-Maria Sofianos, SSCRI/DWIF $5,000.00. (June 30, 2020)
- Eva-Maria Sofianos, WitNY $20,000.00. (January 1, 2019 - December 31, 2019)

- Eva-Maria Sofianos, CUNY Tech Consortium - CEWP JPMorgan Chase $12,000.00. (August 31, 2018)
- Liang Zhao, PSC-CUNY Research Award, $3,500, 2019.
- Liang Zhao, PSC-CUNY Research Award, $6,000, 2020.
- Liang Zhao, "CUNY - NYC Tech Talent Pipeline Career Success Course Innovation Grant," Sponsored by CUNY2X, Local, $10,000.00. (March 24, 2022 - June 30, 2022).
- Mingxian Zhong, PSC-CUNY Research Award, $3,000, 2020.

Research Facilities

Gwang Jung - Dell Server

The equipment is a Dell Server without any OS installed to get the server with lower price. The first estimated price (I guess it was in 2009) was 7,500. It was ITR tech money to purchase equipment to use for teaching research purposes. ITR delayed the purchase almost 1 year, so I guess the price was down (but not revealed). The server has 8 core Intel Xeon processors with 32 gigabytes DDR 3 DRAM.

Brian Murphy - CUDA Workstation

The 2 workstations include 8th Gen Intel i7 CPU, 32 GB DDR4 RAM, 4 x Nvidia GTX Titan graphic cards, and 2TB hard drive. It was purchased with Prof. Murphy’s research grant.

Liang Zhao - Deep Learning Workstation

The workstation was built in 2019. It includes 10th Gen Intel i7 CPU, 32 GB DDR4 RAM, 4 x Nvidia RTX 2080 Ti graphic cards, and 2TB hard drive. It was purchased with Prof. Zhao’s start-up funds in 2019, and it has been used for deep learning research.

Chapter 6: Facilities and Resources

Section 6.1: Offices

The Department of Computer Science office is located in Gillet Hall (GI) room GI-211C. GI-211C is one of three spaces accessible from within GI-211. Opposite GI-211C is GI-211A where the Department of Mathematics office is located. Between the two department offices is a common area including GI-221B, which is a faculty lounge shared by the two departments. Prior to fall 2017 the Department of Mathematics and the Department of Computer Science were one. The former Department of Mathematics and Computer Science was housed in the full space of GI-211. Individual faculty offices are spread out across Gillet Hall and are located on the first, second and third floors. Part time faculty share a large office on the third floor of Gillet Hall in room 303.

Section 6.2: Classrooms and Teaching Laboratories

Whenever possible computer science classes are held in Gillet Hall. Classes are scheduled in Carman Hall and other buildings when Gillet Hall cannot accommodate the class-size and when there are not enough classrooms in Gillet Hall to accommodate demand.

The Department of Computer Science currently has 6 designated teaching laboratories:

Room	Operating System	Number of Computers
GI 207	Windows	23
GI 217	Windows	22
GI 219	Windows	22
GI 221	Windows	23
GI 231	Mac OSX	30

All of these laboratories are on the second floor of Gillet Hall. The rooms are equipped with computers and software such as Java Development Kit (JDK), Eclipse, MYSQL, Microsoft Access, Microsoft Excel, Python IDLE, Android Studio, Visual Studio Code, and XCode needed for teaching our classes.

On the 3rd floor of Gillet Hall the department accesses 3 classrooms, the largest holding approximately sixty students.

Gillet Lecture Hall, room GI-024, with 200 seats is available for the larger CMP 167 - Programming Method I courses. These large classes have also been held in Lovinger Theater, Carman Hall B039 and Carman Hall B04.

All classrooms, teaching laboratories, and lecture halls are equipped with projectors.

Section 6.3: The Computer Science Lab

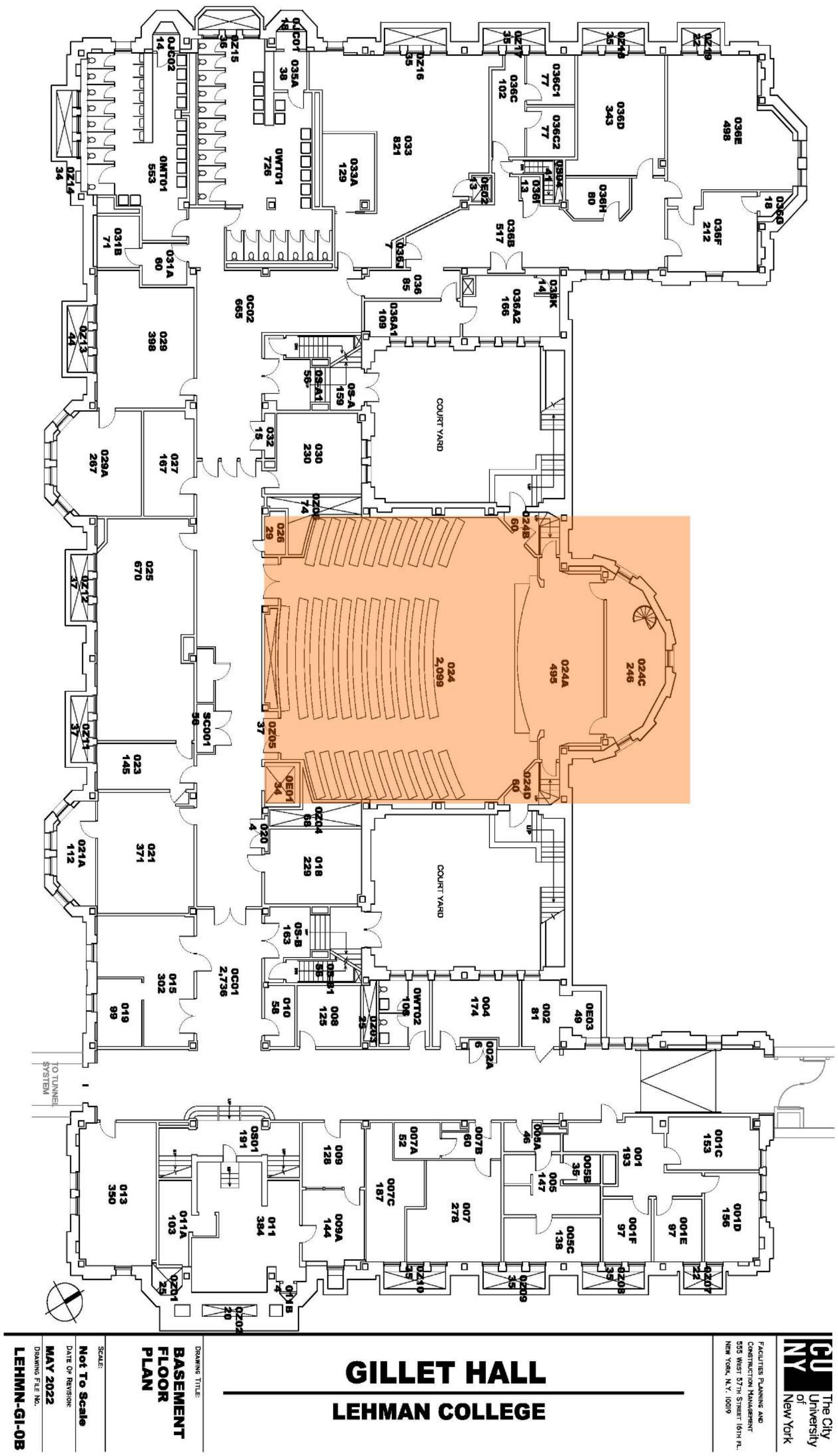
Located in GI 220 & 222, the CS Lab offers students the opportunity to participate in organized peer tutoring. Students actively engage in learning by asking questions, solving problems, applying concepts, and practicing skills. There is an opportunity for students to ask questions, express their understanding verbally, make mistakes and work them through, and learn by doing. A tutor helps students understand concepts and improve their problem-solving skills; they are careful not to do their homework for them. As students come to the CS Lab, they see peers who are committed to learning and who take the time to understand the course material. Students who use the CS Lab report that tutoring helped them gain confidence, improve their study habits, and become more aware of their learning style.

The CS Lab has suffered from underfunding and understaffing for many years, which has been impacted further by the loss of tutoring space. For tutoring services to be maintained and expanded to students taking challenging courses, adequate staffing and funding need to be provided on a timely basis for students to succeed academically.

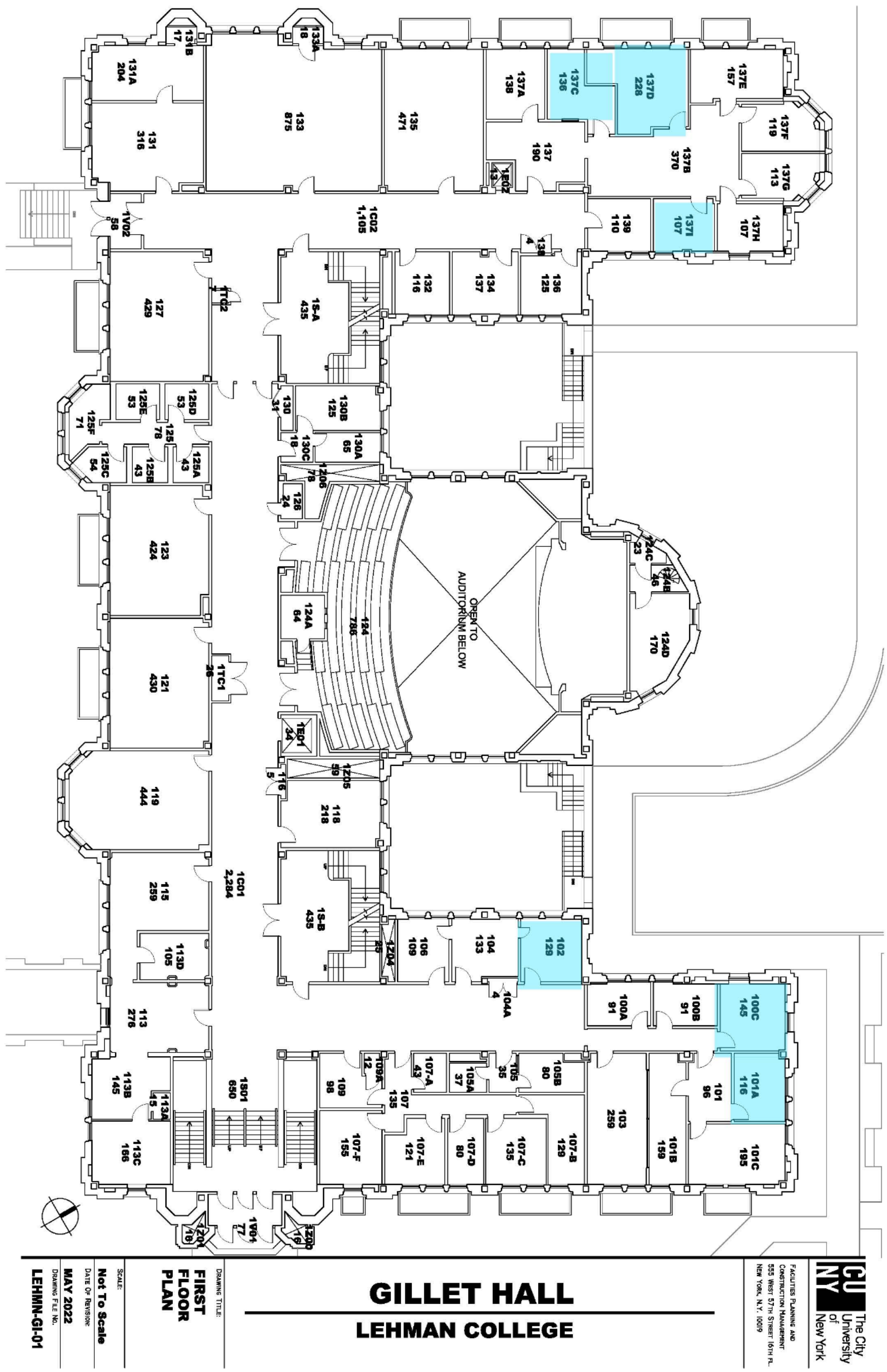
The CS Lab is equipped with 15 Windows computers for students’ use during their visit to the center. The CS Lab was formally called the Math and Computer Science Learning Center (MCSLC) until the summer of 2022. MCSLC provided tutoring services to both mathematics students and computer science students.

The floor plans for these areas are present on the following pages:

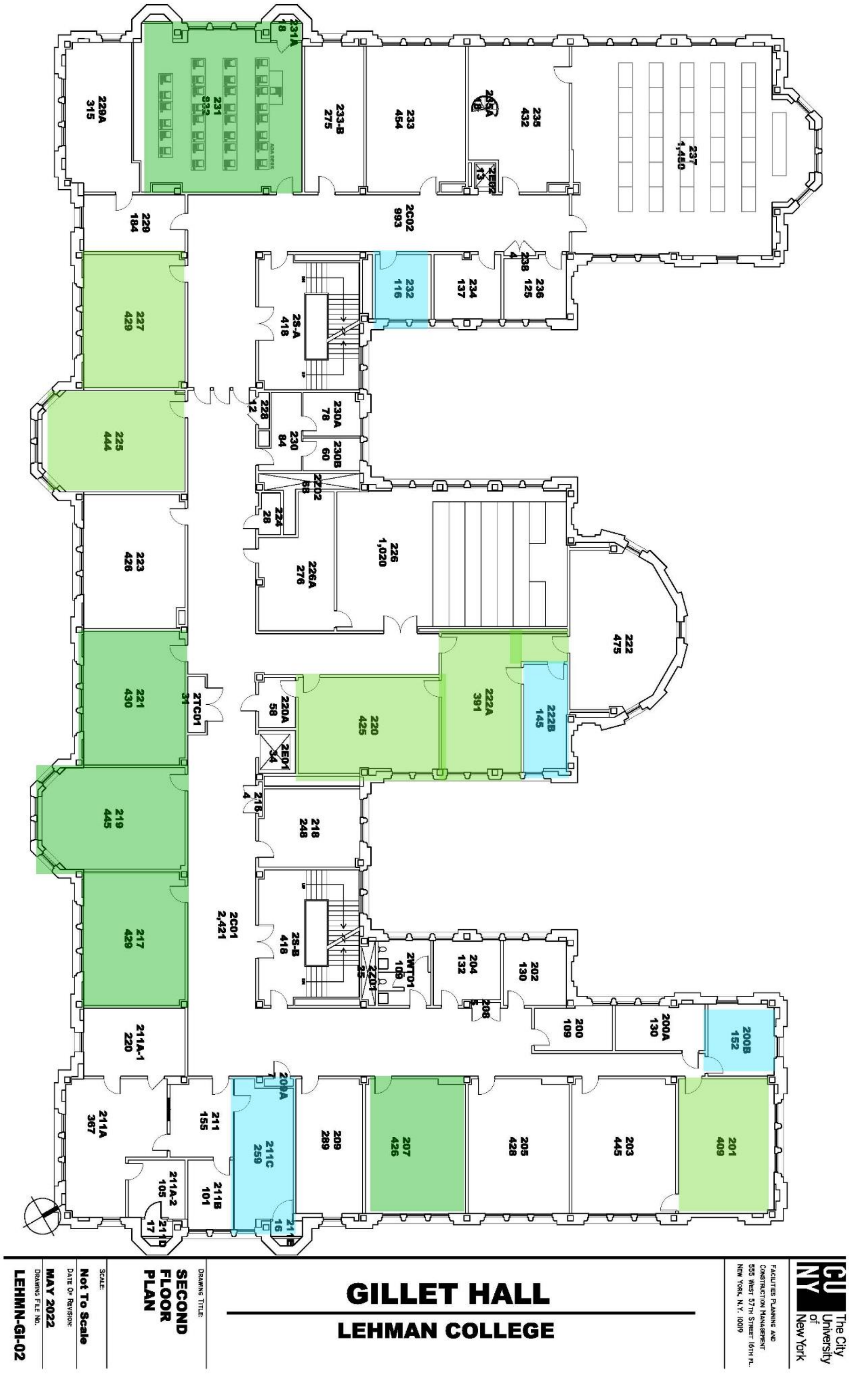
Floor plans for Gillet Hall, Basement



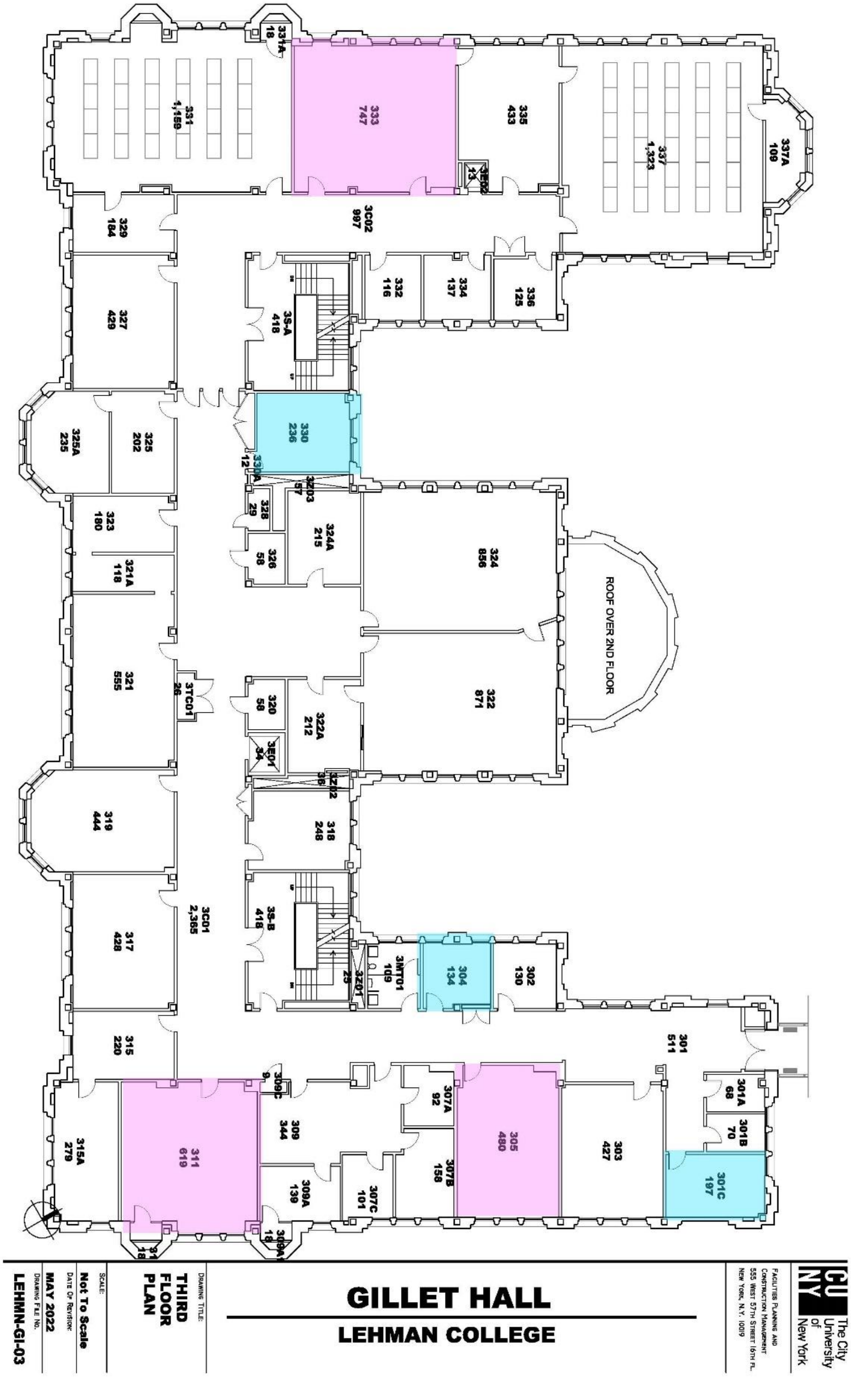
Floor Plans for Gillet Hall, 1st Floor:



Floor Plans of Gillet Hall, 2nd Floor



Floor Plans of Gillet Hall, 3rd Floor



Chapter 7: Special Initiatives

Section 7.1: Computer Science Lab Workshops

The Computer Science Department has conducted various workshops in collaboration with the CS Lab. The workshops were designed to help students develop their skills and knowledge in computer science and math. The workshops were interactive and hands-on, allowing students to apply what they learned in real-world scenarios.

The workshops were led by the experienced Computer Science Lab coordinator, faculty members, tech industry experts, and tutors. They provided guidance and support to participating students, helping them understand complex concepts and solve challenging problems.

The workshops covered a wide range of topics, including:

- Introduction to coding: Students learned the basics of coding, including how to write simple programs in languages such as Python and Java.
- Data analysis and representation: Participants learned how to use tools such as Python to collect, analyze, and visualize data.
- Algorithms: Students learned how to design and implement algorithms to solve complex problems.
- Web development: Participants learned how to create and design websites using HTML, CSS, and JavaScript.
- Git and GitHub
- Data Science
- Robotics: Participants learned how to program for robots

- White boarding and Interview preparations
- Mobile development: Using Google applied computer science resources on mobile development and data structures.

In addition to the workshops, the Department of Computer Science and the CS Lab also provided resources and support for students who wanted to continue learning and developing their skills. These resources included online tutorials, practice problems, and study materials.

Overall, the workshops were a valuable learning experience for students, helping them gain a deeper understanding of computer science and giving them the skills and confidence to succeed in their academic and professional careers.

Section 7.2: The Lehman College - IBM Computer Science Internship Program

Lehman College - IBM Computer Science Internship was a program designed to provide students with hands-on experience in the field of computer science at the IBM Watson Research Laboratories. In this program, graduate school oriented Lehman College Computer Science students earned credit by working at the IBM Watson Research Laboratories under the supervision of an IBM mentor on a project. The IBM mentor coordinated the student's project and progress with a Lehman faculty mentor. Throughout the internship, students received guidance and support from IBM mentors and other professionals. They also had the opportunity to attend workshops and seminars to further their learning and development.

Students were paid by IBM and were expected to take a full program load (12-16 credits) during these semesters and to engage in no other employment. During semester breaks and summers, students were offered full (or part) time employment at IBM. In the final semester, the student had to prepare a summary of the project to IBM and Lehman College Faculty mentors. On recommendation of an appropriate faculty committee, if all other requirements were fulfilled, a student received departmental honors in computer science at graduation. During the semester students spent 16 hours (2 full days) at IBM.

Students were able to work in a high-level work-learning environment. They acquired skills and experience suitable for them to compete for jobs at the good bachelor's level computer science. In addition, there were contacts and letters to help them get acceptance and fellowships at good graduate programs. Students had exposure to a real computing environment with an excellent credential for their resumes. A real benefit of the summer component of the program was that Lehman College students had the opportunity to meet and interact with IBM's summer interns who were among the very best graduate students in the US.

The program ran from 1998 to 2014. Some of the students who were offered full time jobs at IBM, are still working there.

Section 7.3: Tech Talent Pipeline (TTP) Residency & CUNY2X

CUNY 2X Tech was a five-year initiative of the City of New York and the Mayor’s Office that aimed to double the number of graduates from the City University of New York (CUNY) with tech bachelor’s degrees ready to compete for tech jobs in NYC by 2022.

NYC Tech Talent Pipeline (TTP) is a $20 million public-private partnership, launched by the former mayor of New York City, Mayor Bill de Blasio in May 2014 to support the growth of the NYC tech sector and prepare New Yorkers across the five boroughs for 21st century jobs. Lehman College received approximately $2 million from the funds. The other funds were distributed to other CUNY colleges. The Tech Talent Pipeline works with public and private partners to define employer needs, develop, and test training and education solutions, and scale what works throughout the city, delivering quality talent for the City’s businesses and quality jobs for New Yorkers.

The CUNY Tech Talent Pipeline (TTP) Residency was a program designed to provide training and support opportunities for students from underrepresented groups to gain the skills and experience necessary for careers in the technology industry. Participants gained skills and experience they needed to succeed in the competitive field of technology. The program aimed to increase diversity and inclusion in the tech workforce by providing participants with hands-on training and mentorship from industry professionals. The TTP provided CS majors with an opportunity to apply for internships that are subsidized by TTP.

In partnership with The NYC Tech Talent Pipeline (TTP), Lehman College connected qualified, upper-level computer science candidates with NYC businesses looking to host short-term, on-site opportunities for emerging technical talent in roles like:

- Front-end development
- Back-end development
- Mobile development
- Test/QA development

In addition to the training and experience gained through the CUNY Tech Talent Pipeline (TTP) Residency, participants will also have access to a network of industry mentors and professionals who can provide guidance and support as they begin their careers in technology.

Hunter College and Lehman College were the first colleges in CUNY to run TTP. At Lehman College, we had five TTP Students Cohorts over a period of 5 years. The first cohort was in the 2016 – 2017 annual year. The last cohort was in the summer of 2021.

TTP Students, Bootcamp Instructors, and TTP /CUNY2X staff in 2019

The following are some of the companies that we partnered with via TTP:

IBM, ESPN, ADP, Honeywell, RubensteinTech, Macy's Tech, BNY Mellon, American Express, Major League Baseball, Vimeo, Northrop Grumman, Zillow, NYU Langone Health, JPMorgan Chase, Capital One, Infor, Tata Consultancy Services, Wells Fargo, Goldman Sachs, Bank of America, Strongarm Tech.

The following is a summary of data metrics for the five cohorts:

TTP Cohort One

BOROUGH	COUNT	PERCENTAGES
Bronx	16	64%
Brooklyn	1	4%
Manhattan	3	12%
Outside NYC	5	20%
Grand Total	25	100%

GENDER	COUNT	PERCENTAGES
Female	3	12%
Male	22	88%
Grand Total	25	100%

AGE	COUNT	PERCENTAGES
20	1	4%
21	3	12%
22	1	4%
23	2	8%
24	7	28%
25	1	4%
26	3	12%
28	4	16%
29	1	4%
33	1	4%
50	1	4%
Grand Total	25	100%

ETHNICITY/RACE	COUNT	PERCENTAGES
Asian / Pacific Islander	5	20%
Black / African American	3	12%
Hispanic / Latino	13	52%
Other	2	8%
White/Caucasian	2	8%
Grand Total	25	100%

TTP Cohort Two

BOROUGH	COUNT	PERCENTAGES
Bronx	16	64%
Brooklyn	1	4%
Manhattan	3	3%
Outside NYC	5	20%
Grand Total	25	100%

GENDER	COUNT	PERCENTAGES
Female	12	48%
Male	13	52%
Grand Total	25	100%

AGE	COUNT	PERCENTAGES
21	4	16%
22	3	12%
23	3	12%
24	3	12%
25	2	8%
26	2	8%
27	4	16%
28	2	8%
29	1	4%
30	1	4%
Grand Totals	25	100%

ETHNICITY/RACE	COUNT	PERCENTAGES
Asian / Pacific Islander	2	8%
Black/African American	9	36%
Hispanic/ Latino	9	36%
Other	1	4%
White/ Caucasian	4	16%
Grand Totals	25	100%

Cohort Three

BOROUGH	COUNT	PERCENTAGES
Bronx	21	80%
Brooklyn	2	8%
Manhattan	1	4%
Queens	2	8%
Grand Total	26	100%

GENDER	COUNT	PERCENTAGES
Female	7	27%
Male	19	73%
Grand Total	26	100%

AGE	COUNT	PERCENTAGES
18	1	4%
20	3	11%
21	1	4%
22	3	12%
23	7	27%
24	2	8%
25	1	4%
26	2	7%
27	1	4%
28	1	4%
29	1	4%
33	2	7%
40	1	4%
Grand Totals	26	100%

ETHNICITY/RACE	COUNT	PERCENTAGES
Asian / Pacific Islander	3	12%
Black/African American	6	23%
Hispanic/ Latino	6	23%
Other	3	11%
White/ Caucasian	8	31%
Grand Totals	26	100%

Cohort Four

BOROUGH	COUNT	PERCENTAGES
Bronx	24	73%
Brooklyn	5	15%
Manhattan	1	3%
Yonkers	2	6%
Westchester`	1	3%
Grand Total	33	100%

GENDER	COUNT	PERCENTAGES
Female	11	33%
Male	22	67%
Grand Total	33	100%

AGE	COUNT	PERCENTAGES
20	1	3%
21	3	9%
22	7	21%
23	2	6%
24	1	3%
25	3	9%
26	2	6%
27	2	6%
28	5	15%
30	1	3%
31	3	9%
35	2	6%
43	1	3%
Grand Totals	33	100%

ETHNICITY/RACE	COUNT	PERCENTAGES
Asian / Pacific Islander	6	18%
Black/African American	5	15%
Hispanic/ Latino	16	48%
White/ Caucasian	4	12%
American Indian	1	3%
Other	1	3%
Grand Totals	25	100%

Cohort Five

BOROUGH	COUNT	PERCENTAGES
Bronx	18	66%
Manhattan	6	23%
Other	3	11%
Grand Total	27	100%

GENDER	COUNT	PERCENTAGES
Female	6	23%
Male	21	77%
Grand Total	27	100%

AGE	COUNT	PERCENTAGES
21	4	15%
22	3	14%
23	7	26%
24	3	11%
25	1	4%
26	1	4%
27	2	7%
30	2	7%
34	1	4%
36	1	4%
40	1	4%
52	1	4%
Grand Totals	27	104%

ETHNICITY/RACE	COUNT	PERCENTAGES
Asian / Pacific Islander	2	7%
Black/African American	7	26%
Hispanic/ Latino	15	56%
White/ Caucasian	1	4%
Other	2	7%
Grand Totals	27	100%

From our exit data, we concluded that:

- 60% of our cohorts had at least one tech internship before graduation
- 33% of our cohorts had at least two tech internships before graduation
- 61% of our cohorts that graduated landed full-time employment in the tech industry within the first nine months
- 20% of our cohorts that graduated converted their internships into full-time employment in the tech industry
- $92K was the average starting salary for those who landed jobs

Section 7.4: CUNY Inclusive Economy Initiative (CIE)

The CUNY Inclusive Economy Initiative is a $16 million public-private partnership to improve career success of CUNY students and fuel the growth of businesses in New York City. The initiative builds on the proven models (such as TTP) for connecting students to careers and expanding talent pipelines with employers. The program is being funded through the Mayor's Office.

The initiative focuses on many sectors such as the tech industry, healthcare, and green jobs. At Lehman College in the tech field, CIE will be housed in the Department of Computer Science under the supervision of the Department Chair. The Tech CIE program at Lehman College will involve computer science and computer information systems students from the Department of Computer Science, data science minor students from the Sociology Department, and master of Information systems students from the School of Business.

The program focuses on the following three main areas:

- Building talent pipelines directly with the City's employers through dedicated industry engagement staff (advisors and industry specialists) on campus.
- Preparing students for the opportunities generated by the industry-campus backbone.
- Connecting students to internships

The program was announced by Mayor Eric Adams and the CUNY Chancellor Matos Rodrìguez in September of 2022.

Section 7.5: Freshman Year Initiative

The Freshman Year Initiative (FYI) is a two-semester comprehensive learning community program that offers first-year students a supportive structure that promotes an interdisciplinary curriculum, faculty collaboration, and peer support to foster a great sense of campus community and to establish a strong foundation that will ensure academic success. The FYI is designed to help first-year students transition to college and succeed in their academic careers. The program provides students with a range of support services, including academic advising, tutoring, and workshops on topics such as time management and study skills. It also offers opportunities for students to get involved in extracurricular activities and connect with other students on campus. Since 2018, The Computer Science Department has participated in the FYI program through CMP 167 and CMP 168.

Chapter 8: Student Clubs and Organizations

Section 8.1: The National Society of Black Engineers (NSBE)

The Society of Hispanic Professional Engineers (SHPE)

At Lehman College NSBE/SHPE operated as one. The NSBE/SHPE student organization at Lehman College focused on the advancement of students in Computer Science. The club was affiliated with both the National Society of Black Engineers (NSBE) and the Society of Hispanic Professional Engineers (SHPE), professional organizations that provide resources, networking opportunities, and support for students pursuing careers in engineering disciplines. NSBE at Lehman College was founded in Fall of 2014 at Lehman College. The faculty advisory board consisted of various CS department members and significant mentorship from the CS Lab coordinator.

The NSBE/SHPE club at Lehman College offered a variety of activities and programs to support its members, including academic workshops, networking events, hackathons, panel discussions, and mentorship opportunities. The club also had guest speakers and sponsored field trips to visit local companies and institutions in the STEM field.

One of the club's main goals was to increase the representation of minority students in STEM fields, and it provided support and resources for students interested in pursuing a career in computer science and technology. The club also sought to foster a sense of community among its members, providing a supportive and inclusive environment for students to connect with others who share their interests and goals.

NSBE/SHPE Student Leaders at the NSBE/SHPE Annual Hackathon interacting with Department of Computer Science Chair (Professor Brian Murphy, back right in black NSBE/SHPE t-shirt) and the Coordinator of the Computer Science Lab (Mr. Loric Madramootoo, far left)

Overall, the NSBE/SHPE student club at Lehman College provided valuable resources and support for students pursuing careers in STEM fields and works to increase the representation and success of minorities in these fields.

Section 8.2: Women in Computer Science (WiCS)

The Women in Computer Science student club at Lehman College was a student group dedicated to providing support, networking opportunities, and professional development to all students interested in pursuing careers in the field of computer science and technology. The club also educated and raised awareness about the importance of women in the field of computer science and technology. WICS welcomed all students, regardless of gender, who are interested in supporting and advancing women in the field.

Section 8.2: Women in Cybersecurity (WiCyS) student club

The Women in Cybersecurity **(WiCyS)** student club at Lehman College aimed to provide a supportive and inclusive community for women interested in pursuing careers in the field of cybersecurity. Through workshops, guest speakers, and networking events, the club aimed to empower women to gain the knowledge and skills necessary to succeed in the field.

The club also served as a resource for women looking to connect with other professionals and organizations in the cybersecurity industry. The club's members had access to mentorship opportunities, job and internship resources, and networking events with industry professionals.

In addition to providing professional development opportunities, the club also focused on promoting diversity and inclusion in the field of cybersecurity. Through outreach and community events, the club worked to raise awareness about the importance of diversity in the industry and encourage more women to pursue careers in the field.

For all the aforementioned clubs, membership was open to all students from the Lehman College community. Unfortunately, these three clubs suffered membership loss over the pandemic.

Section 8.4: Lehman College Computer Science Club

The CS Club is a student club dedicated to bringing all students who are interested in technology together. The club is open to all students who have an interest in computer science and computer information systems, regardless of their major or level of experience. The club's goal is to provide an inclusive and supportive environment through workshops, projects, and social gatherings.

The club provides opportunities for students interested in computer science to learn, network, and engage in activities related to the field. The club may host events such as guest speaker presentations, workshops, and hackathons, and may also provide resources such as mentorship and networking opportunities, and networking events. The specific activities and events offered by the club may vary depending on the leadership and interests of the members.

The club also has a mentorship program where students can connect with upperclassmen who can provide guidance and support in their studies and career goals.

The club is active throughout the academic year and holds regular meetings for members to discuss upcoming events and activities. Membership is free and open to all students at Lehman College.

Computer Science Clubs Members outside the Student Life Building- The Lehman College Computer Science Club members ready to tackle the latest tech trends and innovations #FutureTech #ComputerScience #LehmanCollege

Chapter 9: Moving Forward

Looking to the future the faculty of the Department of Computer Science would like to be optimists. We do see a glass half-filled. Unfortunately that half-filled glass looks like a metaphor for a half-staffed faculty. The current faculty is stretched to and perhaps beyond its limits. There is a great deal of talk of burnout from a faculty who absolutely love their jobs and the students that they serve. There are many things to be done beyond giving current faculty some time to catch their breaths that the department could be doing if properly staffed. Faculty could be pursuing more grants. (IR and OSP need to be fixed though.) Developing data science and cybersecurity programs are important new directions in which the department would like to expand. There have been ongoing discussions on these subjects going back to years before the department split from mathematics. More professorial faculty are critically needed to strengthen the graduate program. We must bring it back to what it was prior to the explosion of undergraduate enrollments that were coupled with the attrition of faculty which forced our graduate program to first slowly and then all at once transform into offering courses cross-listed with their corresponding undergraduate courses. This is not an exhaustive list.

During the lead up to and after the department split, the vision expressed by successive presidents was of a Computer Science department with a faculty quickly expanding to 15 and beyond once separated from Mathematics. Harsh realities have gotten in the way of expansion beyond 15, but there really is no excuse for the Computer Science department to have merely eight (8) teaching faculty two (2) of whom were hired only because a grant was landed to pay their salaries for the first two (2) years, particularly given all the students we serve. This is doing great harm to the students who would like to get a solid education in computer science, data science, computer information systems, or cybersecurity.

We are thrilled to have a posting for a cybersecurity specialist after conducting no searches for five (5) years. This will help us get a cybersecurity program started if we can hire the right person. We are however, very concerned that a 25% limit on lecturers has been broken for some time and is about to get far worse. In fall 2023 if all goes as currently planned, we will have 6 professors and 5 lecturers (one on Travia leave) in the department. A Computer Science faculty that is 45% lecturers will limit interest from talented candidates for professorial lines. Further, although we have had success hiring great lecturers in the past, each such search has revealed a severe lack of depth in the pool, whereas professorial searches have not.

In short, Computer Science needs an infusion of several talented professorial faculty. This is not a new problem. The appendix is full of documents to make this point clear. Computer Science is a growing field. Enrollments have been going up for more than a decade and continue to go up while almost every other area now goes down. Opportunities to offer a seat to students interested in data science and cybersecurity have already been missed for some time. We should not allow this to continue. For students well trained in these and the other areas that the Department of Computer Science can offer such programs are and can be a further driver of Lehman's upward mobility success. The Provost reported that in 2019 Lehman College Computer Science graduates averaged $87,500 starting salaries. This could be misremembered, but the number the Provost quoted in the same address for Lehman College graduates as a whole was less than half that. Further, word of mouth concerning the atmosphere, quality of the education, and prospects for employment provided by our department has enticed students to join our program, who would have otherwise attended or continued to attend other colleges. With faculty stretched so thin the message about our program has changed a bit.

This is a long running problem for the Department of Computer Science. Appendices D and E contain the department's previous Internal Review Report and the two (2) External Review Reports from 2011 respectively. At that point with a much larger but largely aged faculty (see graphic in report) and a quarter of the student majors that we have now, a desperate plea was made to prevent a "death spiral" for Computer Science. The plea went largely ignored. Appendix F contains the department's 2014-2015 Prioritization reports for Computer Science and Computer Information Systems. The looming disaster identified in the 2011 reports had already begun to manifest by the time the Prioritization reports were written. There were some hirings done, though not nearly enough to compensate for faculty attrition and certainly not enough considering the growth in student interest and the projections for further growth which have come to fruition. Appendix G contains a recent submission offered in an attempt to fill two professorial replacement lines. One is the line previously filled by Professor Gwang Jung who retired as of August 2022 after going out on Travia leave for Spring 2022. Professor Jung has not taught for us for more than a year and his retirement target of 2022 was telegraphed for more than a year so that this was no surprise. The other line was vacated in 2017 by a computer science professor and has remained unfilled, although promised to us repeatedly since 2017. The submission has been edited slightly in anticipation of the next opportunity to solicit those lines for searches. These documents collected in Appendices D-G are not afterthoughts but at the crux of the argument that Computer Science is in a desperate place. It has been for a long time and yet it has managed to function and even thrive in some regards for the students' sake. How much longer will the Department of Computer Science slowly swirl before all at once succumbing to the "death spiral"?

Appendix A: Syllabi

Spring 2023 Syllabi.zip

Appendix B: Program Assessment Reports

ASSESSMENT REPORT SUBMITTED FOR 01/25/2019 DEADLINE

When reviewing previous semesters' assessments of the Computer Science Department's Programming I (CMP167) and Programming II (CMP326/267) courses we realized that each section of these courses had its own unique syllabus. The syllabi and learning objectives were similar across sections, but not identical across all sections for the same course. Therefore, as part of the department's ongoing curriculum restructuring and as a result of the observations that were made while reviewing assessments we decided to create two syllabi, one for each of the two courses. The new syllabus for each course will be shared by all sections of that course beginning with the Spring 2019 semester. Additionally, we selected a common book to be used by all sections of Programming I and II, such that the first half of the book covers the content required in Programming I and the remaining content addresses the topics required in Programming II. Furthermore, we will be creating a shared departmental midterm exam and final exam for each of the two courses so that the assessment of student understanding and concepts is uniform among the sections of a course.

Lehman College

Academic Program Assessment

Assessment Plan – Due by October 11, 2019

Department/Program: ___________Computer Science____________________

Identify learning outcome(s), goal(s), objective(s) to be assessed:

Determine the criteria for measuring success:

HYPERLINK "https://sites.google.com/view/lehman-csandcis-assessment/home"https://sites.google.com/view/lehman-csandcis-assessment/home

Determine the criteria for measuring success:

HYPERLINK "https://sites.google.com/view/lehman-csandcis-assessment/home"https://sites.google.com/view/lehman-csandcis-assessment/home
HYPERLINK "https://sites.google.com/view/lehman-csandcis-assessment/details"https://sites.google.com/view/lehman-csandcis-assessment/details

Identify the method and measures:

HYPERLINK "https://sites.google.com/view/lehman-csandcis-assessment/home"https://sites.google.com/view/lehman-csandcis-assessment/home
HYPERLINK "https://sites.google.com/view/lehman-csandcis-assessment/details"https://sites.google.com/view/lehman-csandcis-assessment/details

The timetable for the collection and analysis of data:

HYPERLINK "https://sites.google.com/view/lehman-csandcis-assessment/details"https://sites.google.com/view/lehman-csandcis-assessment/details

Lehman College
Academic Program Assessment 2019-2020
Assessment Final Report – Steps 4, 5, 6

(sent March 30, 2020 & resent July 2, 2020)
Department/Program: Computer Science

Report on Data Analysis/Assessment findings:

When reviewing previous semesters' assessments of the Computer Science Department's Programming I (CMP167) and Programming II (CMP168 formerly 326/267) courses we realized that each section of these courses had its own unique syllabus. The syllabi and learning objectives were similar across sections, but not identical across all sections for the same course.

Planned next steps based on the assessment findings:

We put a plan into action to unify all sections of each course and proceeded to create two syllabi, one for each of the two courses. The new syllabus for each course will be shared by all sections of that course. We selected a customizable online book to be used by all sections of Programming I and II, such that the first half of the book covers the content required in Programming I and the remaining content addresses the topics required in Programming II. Additionally, we have created shared departmental content for each of the two courses so that the material as well as the assessment of student understanding and concepts is uniform among the sections of a course. The shared contents of online interactive activities to reinforce concepts, homework assignments to apply concepts, a midterm project, a midterm exam, a final project, and a final exam for each course. Furthermore, we have begun collecting data surrounding student participation, assignment completion, and course success.

Planned assessment of the next steps to determine their impact:

We plan to use the data collected to compare students' participation levels with their overall success in the introductory programming course sequence, and expect to see a positive correlation between the two. By unifying the sections in each of the courses, material coverage will match across all sections and grade integrity will be maintained. As a result, it is expected that students who earn a credit grade of "B-" or better will be prepared to move on to the next course and perform well in the second course given their effort persists. Over the course of multiple semesters, we plan to track student participation and success in the programming sequence as well as use the data to observe whether there are particular topics in the course sequence that may benefit from content adjustment if the data analysis shows students are struggling despite their effort.

Lehman College

2020-2021 Assessment Plan

Department/Program: Computer Science

Step 1: Identify Learning Outcome(s), Goal(s), Objective(s) to be assessed:

Goal 1: Write programs and be able to learn other languages easily.

Goal 2: Use basic algorithms for sorting, searching, and organizing data.

Goal 3: Use basic data structures such as lists, stacks, queues, tables, trees and graphs.

Step 2: Determine the criteria for measuring success:

Upon Completion of Programming 1 (CMP 167):
- Understand Logic and be able to Implement Popular Sorting Algorithms:
  - Bubble, Selection, Insertion
- Understand Logic and be able to Implement Popular Searching Algorithms:
  - Sequential aka Linear
- Independently design, create, debug Single-Class Java Applications

Upon Completion of Programming 2 (CMP 168):
- Understand Logic and be able to Implement Searching and Sorting Algorithms from CMP 167 in addition to Binary Search and MergeSort
- Independently design, develop, debug Multi-Class Java Applications
- Demonstrate knowledge of OOP through proper use of encapsulation, polymorphism and inheritance

Upon Completion of Data Structures & Algorithms I (CMP 338):
- Know the tradeoffs of each studied data structure so as to employ the appropriate one for a given situation
- Be able to design algorithms that incorporate data structures for efficient handling of data
- Be able to implement algorithms by coding data structures using an object oriented programming language

Step 3: Identify Methods and Measures for the Assessment:

Use the data collected to compare students' participation levels with their overall success in the introductory programming course sequence, and expect to see a positive correlation between the two.

By unifying the sections in each of the 2 introductory programming courses, material coverage will match across all sections and grade integrity will be maintained. As a result, it is expected that students who earn a credit grade of "B-" or better will be prepared to move on to the next course and perform well in the second course given their effort persists.

Over the course of multiple semesters, we plan to track student participation and success in the programming sequence as well as use the data to observe whether there are particular topics in the course sequence that may benefit from content adjustment if the data analysis shows students are struggling despite their effort.

Step 4: Identify the timeline for the collection and analysis of data:

Over the course of multiple semesters, we plan to track student participation and success in the programming sequence as well as use the data to observe whether there are particular topics in the course sequence that may benefit from content adjustment if the data analysis shows students are struggling despite their effort.

Lehman College

Academic Program Assessment 2020-2021
Assessment Report – Steps 5, 6, 7

Department/Program: Computer Science

Step 5: Report Assessment findings:

We have unified the curriculum across all sections of CMP167 and CMP168 by creating 1 syllabus for each course, as well as adopting and customizing an online book for all sections.

The material covered matches across all sections and the same rubrics, exams, assignments and problem sets are used in assessing student learning.

We have established grade integrity by hosting meetings where all instructors are present and review the scores prior to official grade submission.

In the Fall 2020 semester we ran our first combined lecture section with coordinated labs for CMP 167. We began offering CMP168 using the same structure in the Spring 2021 semester.

Data has begun being collected for all sections of CMP167 and CMP 168.

Let it be noted that along with all the changes described above, this year had the added challenge of the COVID-19 pandemic and distance learning. We are not sure how this may affect the course outcomes and student success.

Step 6: Planned next steps based on the assessment findings:

We have unified the courses and changed the delivery structure from independent lecture-lab sections to one unified lecture with multiple coordinated lab sections. We have also begun collecting data surrounding student participation, assignment completion, exam scores, and overall course success.

We will continue to use the shared contents of online interactive activities to reinforce concepts, homework assignments to apply concepts, a midterm project, a midterm exam, a final project, and a final exam for each course.

Furthermore, we plan on incentivizing students to complete their homework assignments by increasing the weighted value for homework assignments from 15% to 25%. We also plan on reducing the weighted value for exams from 60% to 50% to compensate for the change in weights and offer students more opportunities for success.

Step 7: Planned assessment of the next steps to determine their impact:

We plan to continue collecting data and use it to compare students' participation levels with their overall success in the introductory programming course sequence. We expect to see a positive correlation between the two.

By offering a single lecture section with coordinated labs, students will all attend the same exact lecture and engage in reinforcement through programming practice in their coordinated lab sections. This will further ensure standardization across sections of the course, which will enable grade integrity to be maintained and then used in data analysis. It is expected that students who earn a credit grade of "B-" or better will be prepared to move on to the next course and perform well in the second course given their effort persists.

Over the course of multiple semesters, we plan to track student participation and success in the programming sequence as well as use the data to observe whether there are particular topics in the course sequence that may benefit from content adjustment if the data analysis shows students are struggling despite their effort.

Future assessments of the collected data over the course of multiple semesters will enable us to draw a conclusion on how the unification supports student success and how the pandemic may have also affected learning outcomes.

Lehman College
2021-2022 Assessment Plan

Department/Program: Computer Science

Step 1: Identify Learning Outcome(s), Goal(s), Objective(s) to be assessed:

Goal 1: Write programs and be able to learn other languages easily.

Goal 2: Use basic algorithms for sorting, searching, and organizing data.

Goal 3: Use basic data structures such as lists, stacks, queues, tables, trees and graphs.

Step 2: Determine the criteria for measuring success:

Upon Completion of Programming 1 (CMP 167):
- Understand Logic and be able to Implement Popular Sorting Algorithms:
  - Bubble, Selection, Insertion
- Understand Logic and be able to Implement Popular Searching Algorithms:
  - Sequential aka Linear
- Independently design, create, debug Single-Class Java Applications

Upon Completion of Programming 2 (CMP 168):
- Understand Logic and be able to Implement Searching and Sorting Algorithms from CMP 167 in addition to Binary Search and MergeSort
- Independently design, develop, debug Multi-Class Java Applications
- Demonstrate knowledge of OOP through proper use of encapsulation, polymorphism and inheritance

Upon Completion of Data Structures & Algorithms I (CMP 338):
- Know the tradeoffs of each studied data structure so as to employ the appropriate one for a given situation
- Be able to design algorithms that incorporate data structures for efficient handling of data
- Be able to implement algorithms by coding data structures using an object oriented programming language

Step 3: Identify Methods and Measures for the Assessment:

Use the data collected to compare students' participation levels with their overall success in the introductory programming course sequence, and expect to see a positive correlation between the two.

By unifying the sections in each of the 2 introductory programming courses, material coverage will match across all sections and grade integrity will be maintained. As a result, it is expected that students who earn a credit grade of "B-" or better will be prepared to move on to the next course and perform well in the second course given their effort persists.

Over the course of multiple semesters, we plan to track student participation and success in the introductory programming sequence as well as Data Structures & Algorithms and use the data to observe whether there are particular topics in the course sequence that may benefit from content adjustment if the data analysis shows students are struggling despite their effort.

Now that we have changed the weights for homeworks and exams in the 2 introductory programming courses, we plan on comparing the data from the preceding semesters with the current and future semesters in order to determine if the changes indeed incentivized students to complete their homework and enabled them to better succeed. Please note, the changes made to the weights for homework assignments were from 15% to 25% and exams from 60% to 50% of the overall weighted grade.

Furthermore, we plan on extending the tracking of student participation and success beyond the 2 introductory programming courses and into the Data Structures & Algorithms course and hope that a grade of B- or above in CMP 167 will lead to continued success in CMP 168 and CMP 338 given their efforts persist.

Step 4: Identify the timeline for the collection and analysis of data:

Over the course of multiple semesters, we plan to track student participation and success in the programming sequence as well as use the data to observe whether there are particular topics in the course sequence that may benefit from content adjustment if the data analysis shows students are struggling despite their effort.

Academic Program Assessment 2021-2022
Assessment Report – Steps 5, 6, 7

Department/Program: Computer Science

Step 5: Report Assessment findings:

We continue to use the unified curriculum, customized online book and single syllabus for each course across all sections of CMP167 and CMP168. We continue to maintain grade integrity by hosting meetings where all instructors are present and review the scores prior to official grade submission.

The material covered in CMP 167 and CMP 168 matches across all sections. The same rubrics, exams, assignments, projects and problem sets are used in assessing student learning.

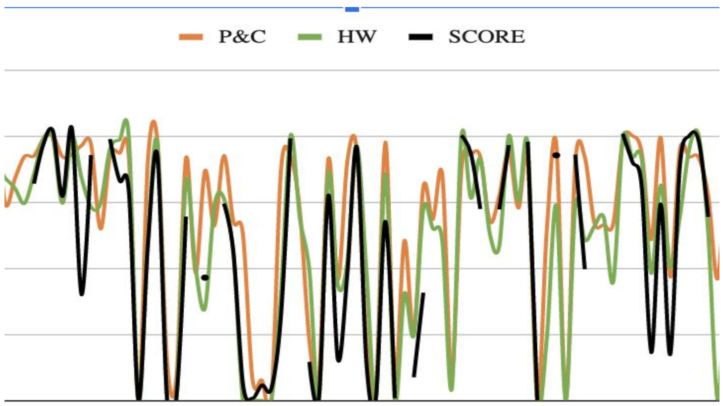
Through the interactive online book we have been able to track student reading and participation (P&C) as well as homework submission to determine how it affects their overall success in a single course. The calculated course grades for CMP 167 and CMP 168 use the same formula to calculate the weighted average: P&C 10%, HW 25%, Projects 15%, Midterm Exam 25%, Final Exam 25%. Upon reviewing the data there is a very strong positive correlation between P&C completion (interactive reading), successful homework correctness and completion, as well as overall course grade. Students who put in the effort by reading the assigned material are able to complete the homework assignments and proceed to correctly completing the projects as well as performing successfully on the exams, thus leading to overall success in the course.

Tracking of individual student success along with continued effort and commitment from one course to the next has not been inspected at this point in time. It is expected that students who earn a credit grade of "B-" or better will be prepared to move on to the next course and perform well in the second course given their effort persists. Data is however being collected so that we can track individual student success in the programming course sequence given their effort persists.

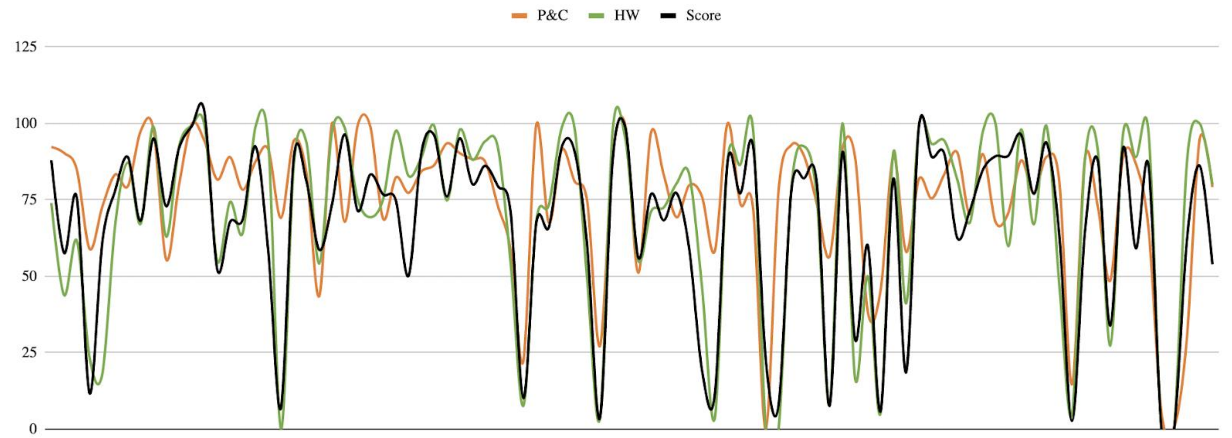
With regard to CMP 338, an online interactive book has been adopted and customized so that student reading and participation as well as homework submission can be tracked. The same assignments, and problem sets in the book are being used to assess student learning across all sections of the course and track student success. The same strong positive correlation between P&C completion (interactive reading), successful homework completion, and overall course grade has been observed.

We are still in the process of collecting data in order to be able to track students along the programming course sequence. That being said, it is evident that properly focused effort on assigned materials leads to student success in the course they are completing the work for. Below are 3 charts depicting the correlation between P&C, HW, and Course Grade for the Fall 2021 semester.

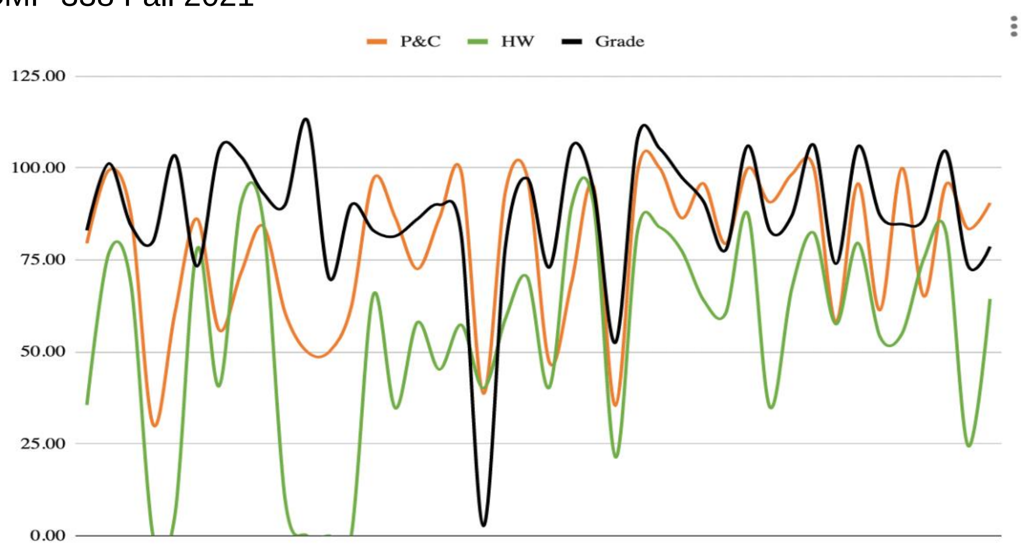
CMP 167 Fall 2021



CMP 168 Fall 2021



CMP 338 Fall 2021



Step 6: Planned next steps based on the assessment findings:

We have unified the 3 programming courses and have changed the delivery structure for the first 2 from independent lecture-lab sections to one unified lecture with multiple coordinated lab sections. We continue to collect data surrounding student participation, assignment completion, exam scores, and overall course success for all 3 courses.

We will continue to use the shared unified content across course sections for each of the courses.

Step 7: Planned assessment of the next steps to determine their impact:

Over the course of multiple semesters, we plan to track student participation and success in each course as well as their progression through the programming sequence. We also plan to use the data to observe whether there are particular topics in the course sequence that may benefit from content adjustment if the data analysis shows students are struggling despite their effort.

Lehman College
2022-2023 Assessment Plan

Department/Program: Computer Science

Step 1: Identify Learning Outcome(s), Goal(s), Objective(s) to be assessed:

Goal 1: Write programs and be able to learn other languages easily.

Goal 2: Use basic algorithms for sorting, searching, and organizing data.

Goal 3: Use basic data structures such as lists, stacks, queues, tables, trees and graphs.

Step 2: Determine the criteria for measuring success:

Upon Completion of Programming 1 (CMP 167):

- Understand Logic and be able to Implement Popular Sorting Algorithms:
  - Bubble, Selection, Insertion
- Understand Logic and be able to Implement Popular Searching Algorithms:
  - Sequential aka Linear
- Independently design, create, debug Single-Class Java Applications

Upon Completion of Programming 2 (CMP 168):

- Understand Logic and be able to Implement Searching and Sorting Algorithms from CMP 167 in addition to Binary Search and MergeSort
- Independently design, develop, debug Multi-Class Java Applications
- Demonstrate knowledge of OOP through proper use of encapsulation, polymorphism and inheritance

Upon Completion of Data Structures & Algorithms I (CMP 338):

- Know the tradeoffs of each studied data structure so as to employ the appropriate one for a given situation
- Be able to design algorithms that incorporate data structures for efficient handling of data
- Be able to implement algorithms by coding data structures using an object oriented programming language

Step 3: Identify Methods and Measures for the Assessment:

Use the data collected to compare students' participation levels with their overall success in the introductory programming course sequence, and expect to see a positive correlation between the two.

By unifying the sections in each of the 2 introductory programming courses, material coverage will match across all sections and grade integrity will be maintained. As a result, it is expected that students who earn a credit grade of "B-" or better will be prepared to move on to the next course and perform well in the second course given their effort persists.

Over the course of multiple semesters, we plan to track student participation and success in the introductory programming sequence as well as Data Structures & Algorithms and use the data to observe whether there are particular topics in the course sequence that may benefit from content adjustment if the data analysis shows students are struggling despite their effort.

Now that we have changed the weights for homeworks and exams in the 2 introductory programming courses, we plan on comparing the data from the preceding semesters with the current and future semesters in order to determine if the changes indeed incentivized students to complete their homework and enabled them to better succeed. Please note, the changes made to the weights for homework assignments were from 15% to 25% and exams from 60% to 50% of the overall weighted grade.

Furthermore, we plan on extending the tracking of student participation and success beyond the 2 introductory programming courses and into the Data Structures & Algorithms course and hope that a grade of B- or above in CMP 167 will lead to continued success in CMP 168 and CMP 338 given their efforts persist.

Due to a problem with registration, many students who completed the CMP 167 prerequisite during the Spring 2022 semester were unable to register in time to take CMP 168 the next immediate semester (Fall 2022). It is expected that these students will take CMP 168 during the Spring 2023 semester. The collection of data will be continued for multiple semesters which will give us the opportunity to study not only the progression of students through the introductory programming sequence, but the difference taking CMP 167 and CMP 168 in non-consecutive semesters versus consecutive semesters has on student retention of previously acquired knowledge and their success in the second course.

Step 4: Identify the timeline for the collection and analysis of data:

Over the course of multiple semesters, we plan to track student participation and success in the programming sequence as well as use the data to observe whether there are particular topics in the course sequence that may benefit from content adjustment if the data analysis shows students are struggling despite their effort.
The progression of students through the introductory programming sequence, as well as the difference in student success when taking CMP 167 and CMP 168 in consecutive semesters versus non-consecutive semesters will be studied.

Lehman College
2022-2023 Assessment Plan

Department/Program: Computer Science

Step 1: Identify Learning Outcome(s), Goal(s), Objective(s) to be assessed:

Goal 1: Write programs and be able to learn other languages easily.

Goal 2: Use basic algorithms for sorting, searching, and organizing data.

Goal 3: Use basic data structures such as lists, stacks, queues, tables, trees and graphs.

Step 2: Determine the criteria for measuring success:

Upon Completion of Programming 1 (CMP 167):
- Understand Logic and be able to Implement Popular Sorting Algorithms:
  - Bubble, Selection, Insertion
- Understand Logic and be able to Implement Popular Searching Algorithms:
  - Sequential aka Linear
- Independently design, create, debug Single-Class Java Applications

Upon Completion of Programming 2 (CMP 168):
- Understand Logic and be able to Implement Searching and Sorting Algorithms from CMP 167 in addition to Binary Search and MergeSort
- Independently design, develop, debug Multi-Class Java Applications
- Demonstrate knowledge of OOP through proper use of encapsulation, polymorphism and inheritance

Upon Completion of Data Structures & Algorithms I (CMP 338):
- Know the tradeoffs of each studied data structure so as to employ the appropriate one for a given situation
- Be able to design algorithms that incorporate data structures for efficient handling of data
- Be able to implement algorithms by coding data structures using an object oriented programming language

Step 3: Identify Methods and Measures for the Assessment:

Use the data collected to compare students' participation levels with their overall success in the introductory programming course sequence, and expect to see a positive correlation between the two.

By unifying the sections in each of the 2 introductory programming courses, material coverage will match across all sections and grade integrity will be maintained. As a result, it is expected that students who earn a credit grade of "B-" or better will be prepared to move on to the next course and perform well in the second course given their effort persists.

Over the course of multiple semesters, we plan to track student participation and success in the introductory programming sequence as well as Data Structures & Algorithms and use the data to observe whether there are particular topics in the course sequence that may benefit from content adjustment if the data analysis shows students are struggling despite their effort.

Now that we have changed the weights for homeworks and exams in the 2 introductory programming courses, we plan on comparing the data from the preceding semesters with the current and future semesters in order to determine if the changes indeed incentivized students to complete their homework and enabled them to better succeed. Please note, the changes made to the weights for homework assignments were from 15% to 25% and exams from 60% to 50% of the overall weighted grade.

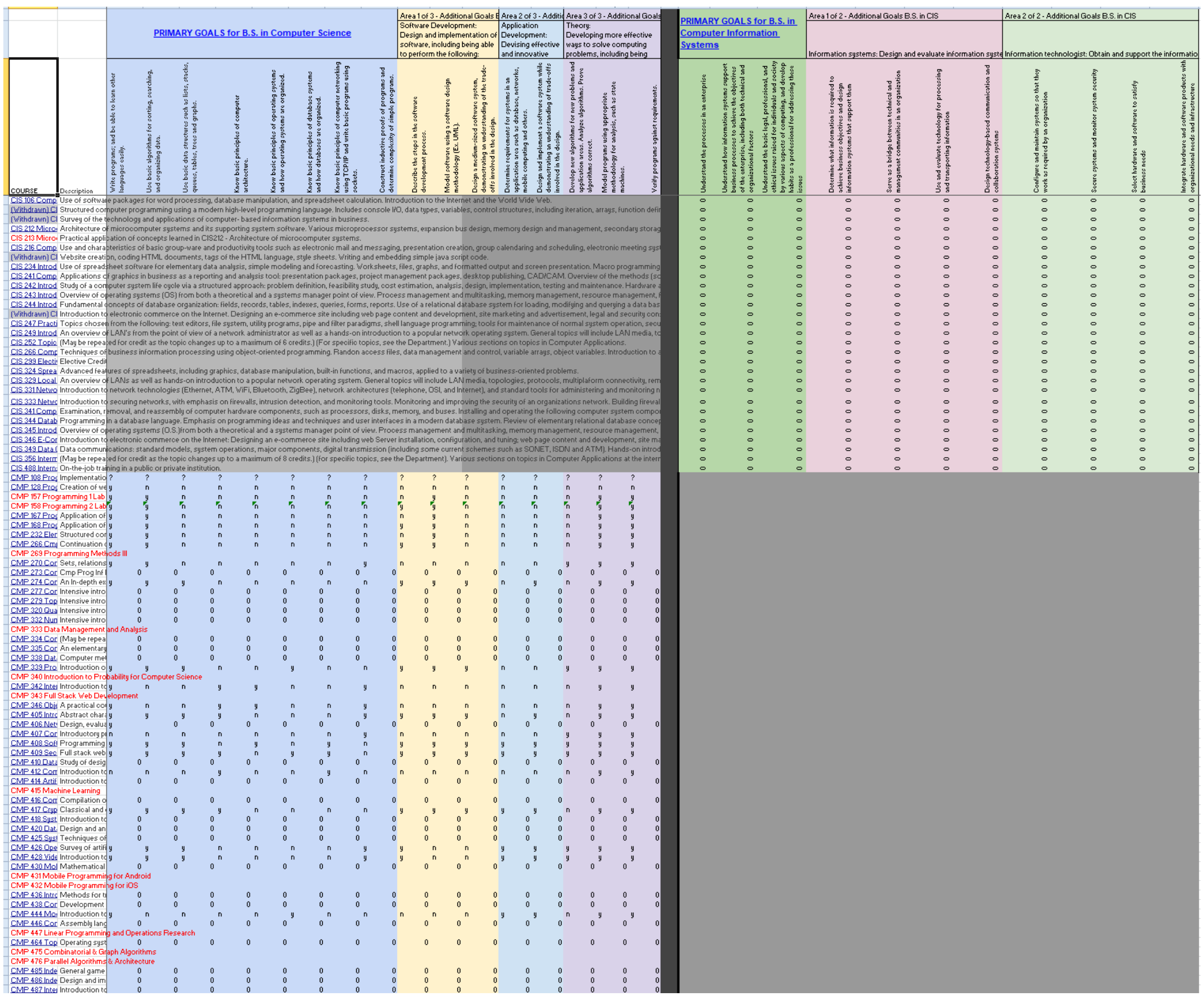
Furthermore, we plan on extending the tracking of student participation and success beyond the 2 introductory programming courses and into the Data Structures & Algorithms course and hope that a grade of B- or above in CMP 167 will lead to continued success in CMP 168 and CMP 338 given their efforts persist.

Due to a problem with registration, many students who completed the CMP 167 prerequisite during the Spring 2022 semester were unable to register in time to take CMP 168 the next immediate semester (Fall 2022). It is expected that these students will take CMP 168 during the Spring 2023 semester. The collection of data will be continued for multiple semesters which will give us the opportunity to study not only the progression of students through the introductory programming sequence, but the difference taking CMP 167 and CMP 168 in non-consecutive semesters versus consecutive semesters has on student retention of previously acquired knowledge and their success in the second course.

Step 4: Identify the timeline for the collection and analysis of data:

Over the course of multiple semesters, we plan to track student participation and success in the programming sequence as well as use the data to observe whether there are particular topics in the course sequence that may benefit from content adjustment if the data analysis shows students are struggling despite their effort.
The progression of students through the introductory programming sequence, as well as the difference in student success when taking CMP 167 and CMP 168 in consecutive semesters versus non-consecutive semesters will be studied.

Course_Objective_Al
ignment_CMP_CIS.xls



Above you will find an embedded Assessment Excel file.

Appendix C: Faculty CVs

CURRICULUM VITAE

Victor Y. Pan
Distinguished Professor

Department of Mathematics and Computer Science
Lehman College and the Graduate Center
City University of New York
Bronx, New York 10468
Tel. (718) 960-8568 (o ce)
Fax (718) 960-8969
E-Mail: victor.pan @ lehman.cuny.edu
Website: http://comet.lehman.cuny.edu/vpan/

1 PERSONAL DATA AND EDUCATION

1.1 PERSONAL DATA

Born in Moscow, USSR, on September 8, 1939 in the family of Yakov Solomonovich Pan, 1906-1941, the author of a bestseller of 1940 about scienti c discoveries, republished in the 21st century in England and France see Wikipedia article: Œ º Ł ˇ (Yakov Solomonovich Pan", in Russian), and Rievka Kalmanovna Kogan, 1904-1965

Immigrated to the U.S. in 1977

U.S. Citizen since 1982

Married to Lidia Perelman (Pan), literary critic see Wikipedia article: ¸Łº ˇ (Lilya Pann", in Russian) by link from: ˝ ˚ — ææŒ Ø ¸Ł ß (Novaya Karta Russkoy Literatury", in Russian)

Hobbies: Reading and writing poetry, Mountaineering, Swimming, Skiing and CrossCountry Skiing

Languages: English, Russian, French

1.2 HIGH SCHOOL EDUCATION

1946 1956: 59th High School in Moscow, USSR

Moscow High School Olympiads in Mathematics: prizes in 1954, 1955 and 1956

1.3 HIGHER EDUCATION:

1956 1964: Department of Mechanics and Mathematics Moscow State University (MGU)

1961: M.S. in Mathematics

1964: Ph.D. in Mathematics (Thesis Advisor: A. G. Vitushkin)

2 EMPLOYMENT AND CONSULTING:

2.1 EMPLOYMENT

1988 Visiting Professor, Professor, and (since 2000) Distinguished Professor at the Department of Mathematics and Computer Science of Lehman College of the City University of New York (CUNY) and in the Ph.D. Programs in Computer Science and in Mathematics of the Graduate Center of CUNY

1979 80 and 1981 1991 Professor at the Computer Science Department of the State University of New York at Albany (SUNYA)

August 2002 Visiting Scientist

Ontario Research Center in Computer Algebra (ORCCA)

Waterloo and London, Western Ontario, Canada

June 2002 Visiting Scientist

Mathematics and Informatics Departments, University of Pisa, Italy

August-September 1998 Senior Key Scientist

Mathematical Science Research Institute, Berkeley, California

July 1998 Visiting Scientist

Fields Research Institute, Toronto, Canada

March August 1996 and March June 1997 Invited Scientist Project SAFIR, INRIA

Sophia Antipolis, France

January 1991 and July August 1992 Visiting Scientist

International Computer Science Institute, Berkeley, California

1989 90 Visiting Professor

Computer Science Department, Columbia University, New York

July 1984 Visiting Professor

Department of Mathematics, University of Pisa and CNR, Italy

April June 1981 Visiting Professor

Computer Science Department, Stanford University, California

1980 81 Visiting Member

the Institute for Advanced Study, Princeton, New Jersey

1977 79 and August 1980 Visiting Scientist

IBM Research Center, Yorktown Heights, New York

1969 76 Senior Researcher

Department of Models for National Economy

Institute of Economics, Academy of Science, Moscow, Russia

1965 69 Senior Researcher

Department of Computations for Economics

Institute of Electronic Control Machines, Moscow, Russia

1964 65 Junior Researcher

Department of Computations for Economics

Institute of Electronic Control Machines, Moscow, Russia

2.2 CONSULTING:

ATT Bell Laboratories, Murray Hill, New Jersey, 1991 1993

General Electric Research and Development Center, Schenectady, New York, 1980

3 PROFESSIONAL SOCIETIES; RESEARCH AREAS

3.1 MEMBERSHIP IN PROFESSIONAL SOCIETIES:

American Mathematical Society, since 1977

Designation of AMS Fellowship For Contributions to the Mathematical Theory of Computation", 2014

Association for Computing Machinery, since 1977

Society for Industrial and Applied Mathematics, since 1977

European Association for Theoretical Computer Science

International Linear Algebra Society

3.2 AREAS OF RESEARCH SPECIALIZATION Also see section 12: MY RESEARCH JOURNEY

Design and Analysis of Algorithms

Computational Complexity

Polynomial Computations

Computations with General, Data Sparse, and Random Matrices
Numerical Algorithms

Symbolic Algorithms

Symbolic Numerical Algorithms

Parallel Algorithms

Graph Algorithms

4 GRANTS AND AWARDS:

Special Creativity Extension Award from the Numeric, Symbolic, and Geometric Computation Program of the CCR Division in the Directorate CISE of NSF (1993) Best Paper Award 2000, Journal of Complexity: $3,000 (shared) NSF Grants (individual): $1,931,143 (1980 2016), including NSF Grants (joint): $1,056,241 (2016 2021) 26 PSC-CUNY Awards (individual): $135,077, 1989 2021

CUNY Institute for Software Design and Development Grants: $8,000, 2001 2002

Shuster Foundation Award: $4,000, 1994 2000

Lehman College CUNY, Faculty Award for Research and Scholarship: $1,000, 1994

Institute for Advanced Study, Grant: $13,000, 1980 81

SUNY University Award: $2,000, 1980

4.1 MOST RECENT:

NSF Grant (individual) CCF 1116736,

Novel Methods for Fundamental Matrix and Polynomial Computations", $350,000

(from 8/1/2011 to 12/31/2016)

NSF Grant (joint: A. Ovchinnikov PI, V. Pan co-PI, and C. Yap co-PI) CCF - 1563942 AF: Medium: Collaborative Research: Numerical Algebraic Di erential Equations, $608,205 (from 7/1/2016 to 6/30/2021)

NSF Grant (joint: Bo Yuan PI, V. Pan co-PI, and Xue Lin co-PI) AitF, CCF1733834, Medium: Collaborative Research: A Framework of Simultaneous Acceleration and Storage Reduction on Deep Neural Networks Using Structured Matrices, $448,086.00 (from 9/15/2017 to 12/31/2021)

PSC CUNY AWARD 69813 00 48: New Progress in Matrix and Polynomial Computations, $11,998.92 (from July 1, 2017 to December 31, 2018)
PSC CUNY AWARD 62797-00 50: $11,999.74 (from June 30, 2019 to December 31, 2020)

PSC CUNY AWARD 63677-00 51:

$5,999.87 (from July 1, 2020 to June 30, 2021)

4.2 PSC CUNY AWARDS (LISTING SINCE 2003):

AWARD 65393 0034: "Algebraic and Numerical Algorithms",

$3,297, 6/30/2003 7/1/2004

AWARD 6643 7 0035: "Algebraic and Numerical Computing",

$3,495, 6/30/2004 7/1/2005

AWARD 67297 0036: "Matrix and Polynomial Computations",

$2,805, 6/30/2005 7/1/2006

AWARD 68291 0037: "Matrix and Polynomial Computations",

$3,176, 6/30/2006 7/1/2007

AWARD 69330 00 38: Algebraic and Numerical Algorithms for Matrix and Polynomial Computations", $3,990, 6/30/2007 7/1/2008

AWARD 61406 00 39: Algebraic and Numerical Algorithms for Matrix and Polynomial Computations", $3,800, 6/30/2008 7/1/2009

AWARD 62230 00 40: Algebraic and Numerical Algorithms for Matrix and Polynomial Computations", $4,300, 6/30/2009 7/1/2010

AWARD 63153 00 41 Algebraic and Numerical Algorithms for Matrix and Polynomial Computations", $2,860, 6/30/2010 7/1/2011

AWARD 64512 0042: Matrix and Polynomial Computations",

$6,000, 6/30/2011 7/1/2012

AWARD 65792 0043: Matrix and Polynomial Algorithms",

$11,998.92, 6/30/2012 7/1/2013

AWARD 67699 00 45: Advancing Matrix and Polynomial Computations",

$6,000, 6/30/2014 7/1/2015

AWARD 68862 00 46: Advancing Matrix and Polynomial Computations",

$11,998, 7/1/2015 12/31/2016

AWARD 69813 00 48: New Progress in Matrix and Polynomial Computations", $11,998.92, 6/30/2017 7/1/2018

AWARD 62797-00 50: $11,999.74

7/1/2019 12/31/2020

AWARD 63677-00 51,

$5,999.87, July 1, 2020 June 30, 2021

5 SERVICE TO PROFESSION

5.1 JOURNAL EDITING

Area Editor:

Theoretical Computer Science (since 1985 to present)

Computers and Mathematics (with Applications), (1980 2011)

Calcolo (1999 2020)

Special Issues (Corresponding and Managing Editor):

I. Z. Emiris, B. Mourrain, V. Y. Pan, Guest Editors. Special Issue on Algebraic and Numerical Algorithms, Theoretical Computer Science, 315, 2 3, 307 672, 2004

Special Issue on Symbolic Numerical Algorithms (D. A. Bini, V. Y. Pan, and J. Verschelde editors), Theoretical Computer Science, 409, 2, 155 331, 2008

Special Issue on Algebraic and Numerical Algorithms (I. S. Kotsireas, B. Mourrain, and V. Y. Pan, editors), Theoretical Computer Science, 412, 16, 1443 1543, 2011

Special Issue on Algebraic and Numerical Algorithms (I. S. Kotsireas, B. Mourrain, V. Y. Pan, and Lihong Zhi, editors), Theoretical Computer Science, 479, 1 186, 2013

5.2 PROGRAM AND SCIENTIFIC COMMITTEES MEMBER

ACM Annual International Symposium on Symbolic and Algebraic Computation (ISSAC 1999), Vancouver, British Columbia, Canada, July-August 1999

ACM Annual International Symposium on Symbolic and Algebraic Computation (ISSAC 2007), Waterloo, Ontario, Canada, July-August 2007

The 2nd International Workshop on Symbolic-Numeric Computation (SNC 2007), London, Ontario, Canada, July 2007

The Annual International Conference on Polynomial Computer Algebra, St. Petersburg, Russia, Aprils of 2008 2021 (Fourteen Committees)

The 4th International Workshop on Symbolic-Numeric Computation (SNC 2011), San Jose, California, June 2011

International Symposium on Linear Algebra (ILAS 2013), Providence, RI, June 2013

The 5th International Workshop on Symbolic-Numeric Computation (SNC 2014), Shanghai, China, July 2014

5.3 LOCAL ARRANGEMENTS CHAIR

ACM Annual International Symposium on Symbolic and Algebraic Computation (ISSAC 2018), New York, NY, the Graduate Center of CUNY, July 2018

5.4 OTHER PROFESSIONAL ACTIVITIES

Organization of Conferences and Conference Minisymosia (latest two Minisymposia at SIAMLA 2021)

Refereeing and Reviewing for Professional Journals, Conferences and Surveys

Lectures and Invited Lectures at Conferences in Computer Science, Mathematics, and Applied Mathematics in North and South Americas, Europe, Asia, and Australia (see the Lists of Publications and Talks at the Conferences)

Colloquium Lectures at the Universities and Research Centers

6 SERVICE TO LEHMAN COLLEGE AND CUNY

6.1 SERVICE TO LEHMAN COLLEGE:

Personnel and Budget Committee Member, 2017 2021

Supervising Syllabus in Computer Science

Advising students in Mathematics and Computer Science Observing Junior Instructors in

Mathematics and Computer Science

6.2 SERVICE TO CUNY:

Teaching at the Graduate School and University Center (1989 2021, except for the sabbatical year of 1996 97)

Advising Ph.D. Students: 27 Ph.D. Defenses (see the List of Ph.D. Defenses)

Serving as the Chair of 27 PhD Defense Committees in Mathematics and Computer Science (since 1991)

Member of Distinguished Professor Selection Committee (2005-2007, 2013 and 2016)

Member of the Leadership Committee of PhD Program in Computer Science (member, 2012 2013)

Member of the PhD Defense Committees in Mathematics (18) and Computer Science (23) since 1991

7 Ph.D. STUDENTS OF CUNY SUPERVISED AND MENTORED BY VICTOR PAN (27 students by 2023)

STUDENT NAME, PhD THESIS DEFENSE, GRADUATION DATE, Ph.D. PROGRAM

Atinkpahoun, A., April 11, 1995; June 1995, Computer Science

Cebecioglu, H., May 23, 2001; October 2001, Mathematics

Chakraborty, H., April 16, 2021; June 2021, Computer Science
Chen, Z.Q., November 9, 1999; February 2000, Mathematics

Dias, O., November 26, 1996; January 1997, Mathematics

Huang, X., July 1997; October 1997, Mathematics

Landowne, E., November 1995; February 1996, Computer Science

Lin, Y., March 1991; June 1991, Computer Science

Luan, Q., March 27, 2020; August 2020, Mathematics

Murphy, B., March 27, 2007; May 2007, Computer Science

Providence, S., December 14, 1999; February 2000, Computer Science

Rami, Y., February 22, 2000; June 2000, Mathematics

Retamoso Urbano, I.O., December 17, 2014; February 2015, Mathematics Rosholt, R.E., April

4, 2003; May 2003, Computer Science ou, A., January 12, 1996; October 1996, Computer

Science Serme, A., February 2008; May 2008, Mathematics

Sobze, I., April 12, 1994; June 1994, Computer Science

Stuart, C., April 1998; June 1998, Computer Science

Svadlenka, J., April 3, 2020; June 2020, Computer Science

Abu Tabanjeh, M.A., November 9, 1999; February 2000, Mathematics

Taj Eddin, I., March 27, 2007; September 2007, Computer Science

Wang, X., April 4, 2003; May 2003, Mathematics

Wolf, J., January 7, 2015; May 2015, Mathematics

Yan, X., January 29, 2015; February 2015, Computer Science

Yu, Y., April 1998; June 1998, Computer Science

Zhao, L., April 6, 2017, June 2017, Mathematics

Zheng, A., October 16, 1997; January 1998, Mathematics

A. Atinkpahoun, O. Dias, S. Providence, A. Sadikou, A. Serme, and I. Sobze are African Americans. H. Celecioglu, O. Dias, Y. Lin, and H. Chakraborty are females. At all the listed defenses, Victor Pan has served as the Mentor, the Adviser and the Chair of the Examination Committees, except that for H. Chakraborty he was a co-Adviser and a co-Chair.

8 PUBLICATIONS BY SUBJECTS

Also see PUBLICATIONS and RESEARCH in sections 9, 10, and 12

8.1 POLYNOMIALS: EVALUATION, INTERPOLATION, MULTIPLICATION, DIVISION, GCDs

BOOKS

1. Polynomial and Matrix Computations", Volume 1: Fundamental Algorithms" (XVI + 415 pages) (by D. Bini and V. Y. Pan), in the series Progress in Theoretical Computer Science (R.V. Book editor), Birkh user, Boston (1994).

2. Structured Matrices and Polynomials: Uni ed Superfast Algorithms" (XXV + 278 pages), Birkh user/Springer

REVIEW ARTICLES AND BOOK CHAPTERS

1."On Methods of Computing the Values of Polynomials", Uspekhi MatematicheskikhNauk (in Russian), 21, 1 (127), 103 134 (1966). (Transl. Russian Mathematical Surveys, 21, 1 (127), 105 137 (1966).)

2."Complexity of Computations with Matrices and Polynomials," SIAM Review, 34, 2,225 262 (1992).

3."Algebraic Algorithms" (by A. Diaz, E. Kaltofen and V. Y. Pan), Chapter 10 in theComputer Science and Engineering Handbook (Allen B. Tucker, Jr., editor), 226-249, CRC Press Inc., Boca Raton, Florida (1997) and Chapter 8 in the Computer Science and Engineering Handbook (Allen B. Tucker, editor), pp. 8 1 to 8 24, Chapman and Hall/CRC Press, 2004.

4."Some Recent Algebraic/Numerical Algorithms", Electronic Procs. IMACS/ACA'98 (1998). Available at http:www-troja.fj .cvut.cz/aca98/sessions/approximate

5."Algebraic Algorithms" (by A. Diaz, I. Z. Emiris, E. Kaltofen and V. Y. Pan), Chapter16 in Handbook "Algorithms and Theory of Computations", pp. 16 1 to 16 27 (M. Atallah, editor), CRC Press Inc., Boca Raton, Florida (1999).

6.. "Fast Fourier Transform and Its Applications" (by I. Z. Emiris and V. Y. Pan),Chapter 17 in Handbook "Algorithms and Theory of Computations", pp. 17 1 to 17 30 (M. Atallah, editor), CRC Press Inc., Boca Raton, Florida (1999).

7.Algebraic and Numerical Algorithms" (by I. Z. Emiris, V. Y. Pan, and E. Tsigaridas), in Algorithms and Theory of Computations Handbook", Second Edition, Volume 1 (1016 pages): General Concepts and Techniques, pages 1 34 in Chapter 17 (Mikhail J. Atallah and Marina Blanton, editors), CRC Press Inc., Boca Raton, Florida (2009).

8."Fast Fourier Transform and Its Applications" (by I. Z. Emiris and V. Y. Pan), in Algorithms and Theory of Computations Handbook", Second Edition, Volume 1 (1016 pages): General Concepts and Techniques, pages 1 31 in Chapter 18 (Mikhail J. Atallah and Marina Blanton, editors), CRC Press Inc., Boca Raton, Florida (2009).

9.Algebraic Algorithms" (by I. Z. Emiris, V. Y. Pan, and E. Tsigaridas), Chapter 10 (pages from 10 1 to 10-40) of Computing Handbook (Third edition), Volume I: Computer Science and Software Engineering (Allen B. Tucker, Teo Gonzales, and Jorge L. Diaz-Herrera, editors), Taylor and Francis Group, 2014. Available at arXiv 1311.3731 [cs.DS]

RESEARCH PAPERS (in journals and refereed proceedings of conferences).

1."Some Schemes for the Evaluation of Polynomials with Real Coe cients", Doklady Akademii Nauk SSSR (in Russian), 127, 2, 266 269 (1959).

2."Some Schemes for the Evaluation of Polynomials with Real Coe cients" Problemy Kibernetiki (in Russian), (edited by A.A. Lyapunov), 5, 17 29 (1961). (Transl. Problems of Cybernetics, USSR, 5, 14 32, Pergamon Press (1961).)

3."On Some Methods of Computing Polynomial Values", Problemy Kibernetiki (in Russian), (edited by A.A. Lyapunov), 7, 21 30 (1962). (Transl. Problems of Cybernetics, USSR, 7, 20 30, U.S. Dept. of Commerce (1962).)

4."Schemes with Preconditioning for the Evaluation of Polynomials and a Program forAutomatic Preconditioning", Zhurnal Vychislitel’noy Matematiki i Matematicheskoy Fiziki (in Russian), 2, 1, 133 140 (1962). (Transl. from USSR Computational Mathematics and Mathematical Physics, 1, 137 146 (1963).)

5."Methods for Computing Polynomials" (in Russian), Ph.D. thesis, Dept. of Mechanicsand Mathematics, Moscow State University (1964).

6."The Evaluation of Polynomials of the Fifth and Seventh Degrees with Real Coe cients", Zhurnal Vychislitel’noy Matematiki i Matematicheskoy Fiziki (in Russian), 5, 1, 116 118 (1965). (Transl. USSR Computational Mathematics and Mathematical Physics, 5, 1, 159 161 (1965).)

7."On Simultaneous Evaluation of Several Polynomials of Low Degree (Two to Five)",Zhurnal Vychislitel’noy Matematiki i Matematicheskoy Fiziki (in Russian), 6, 2, 352 357 (1966). (Transl. USSR Computational Mathematics and Mathematical Physics, 6, 2, 222 227 (1966).)

8."Computational Complexity of Computing Polynomials over the Fields of Real andComplex Numbers", Proceedings of the Tenth Annual ACM Symposium on Theory of Computing (STOC’78), 162 172, ACM Press, New York (1978).

9."Convolution of Vectors over the Real Field of Constants", J. of Algorithms, 1, 297 300 (1980).

10. "Fast Parallel Polynomial Division via Reduction to Polynomial Inversion Modulo aPower" (by D. Bini and V. Y. Pan), Information Processing Letters, 21, 79 81 (1985).

11. "Algorithms for Polynomial Division" (by D. Bini and V. Y. Pan), Proc. EuropeanConference on Computer Algebra, Linz, Austria, Lecture Notes in Computer Science, 204, 1 3, Springer (1985).

12. "A Logarithmic Boolean Time Algorithm for Parallel Polynomial Division" (by D.Bini and V. Y. Pan), VLSI Algorithms and Architectures, Lecture Notes in Computer Science, 227, 246 251, Springer, Berlin (1986).

13. "Polynomial Division and Its Computational Complexity" (by D. Bini and V. Y. Pan),Journal of Complexity, 2, 179 203 (1986).

14. "Fast Parallel Algorithms for Polynomial Division over Arbitrary Field of Constants"(by D. Bini and V. Y. Pan), Computers and Mathematics (with Applications), 12A, 11, 1105 1118 (1986).

15. "Fast Evaluation and Interpolation at the Chebyshev Sets of Points", Applied Math.Letters, 2, 3, 255 258 (1989).

16. "Univariate Polynomial Division with a Remainder by Means of Evaluation and Interpolation" (by V. Y. Pan, E. Landowne, and A. Sadikou), Proc. of 3rd IEEE Symp. on Parallel and Distributed Processing, 212 217, IEEE Computer Society Press, Los Alamitos, California (1991).

17. "Polynomial Division with a Remainder by Means of Evaluation and Interpolation"(by V. Y. Pan, E. Landowne, and A. Sadikou), Information Processing Letters, 44, 149 153 (1992).

18. "The Power of Combining the Techniques of Algebraic and Numerical Computing: Improved Approximate Multipoint Polynomial Evaluation and Improved Multipole Algorithms" (by V. Y. Pan, J. Reif, and S. Tate), Proc. of 33rd Ann. IEEE Symp. on Foundations of Computer Science (FOCS ’92), 703 713, IEEE Computer Society Press, Los Alamitos, California (1992).

19. "Improved Parallel Polynomial Division and Its Extension" (by D. Bini and V. Y. Pan),Proc. of 33rd Ann. IEEE Symp. on Foundations of Computer Science (FOCS’92), 131 136, IEEE Computer Society Press, Los Alamitos, California (1992).

20. "A New Approach to Fast Polynomial Interpolation and Multipoint Evaluation" (byV. Y. Pan, A. Sadikou, E. Landowne, and O. Tiga), Computers and Math. (with Applications), 25, 9, 25 30 (1993)

21. Improved Parallel Polynomial Division" (by D. Bini and V. Y. Pan), SIAM J. on Computing, 22, 3, 617 627 (1993).

22. "Simple Multivariate Polynomial Multiplication", J. Symbolic Computation, 18, 183 186 (1994).

23. "Algebraic Improvement of Numerical Algorithms: Interpolation and Economizationof Taylor Series", Mathematical and Computer Modeling, 20, 1, 23 26 (1994).

24. "An Algebraic Approach to Approximate Evaluation of a Polynomial on a Set of RealPoints", Advances in Computational Mathematics, 3, 41 58 (1995).

25. "Parallel Computation of Polynomial GCD and Some Related Parallel Computationsover Abstract Fields", Theoretical Computer Science, 162, 2, 173 223 (1996).

26. "Computing $x^n \bmod p(x)$ and an Application to Splitting a Polynomial into Factors over a Fixed Disc", Journal of Symbolic Computations, 22, 377 380 (1996).

27. "Fast Multipoint Polynomial Evaluation and Interpolation via Computation withStructured Matrices" (by V. Y. Pan, A. Zheng, X. Huang, and Y. Yu), Annals of Numerical Math., 4, 483 510 (1997).

28. "New Fast Algorithms for Polynomial Interpolation and Evaluation on the ChebyshevNode Set", Computers and Math. (with Applications), 35, 3, 125 129 (1998).

29. "Approximate Polynomial Gcds, PadØ Approximation, Polynomial Zeros, and Bipartite Graphs", Proc. 9th Ann. ACM-SIAM Symp. on Discrete Algorithms (SODA’98), 68 77, ACM Press, New York, and SIAM Publications, Philadelphia (1998).

30. Approximate Real Polynomial Division via Approximate Inversion of Real Triangular Toeplitz Matrices" (by V. Y. Pan and Z. Q. Chen), Applied Math. Letters, 12, 1 2 (1999).

31. Polynomial and Rational Interpolation and Multipoint Evaluation (with Structured Matrices)" (by V. Olshevsky and V. Y. Pan), Proc. 26th Intern. Colloquium on Automata, Languages

and Programming (ICALP'99), 1644, 585 594, Springer's Lecture Notes in Computer Science, Springer, Berlin (July 1999).

32. Numerical Computation of a Polynomial GCD and Extensions", Information and Computation, 167, 2, 71 85 (2001).

33. Polynomial Evaluation and Interpolation and Transformations of Matrix Structures", Proceedings of the 15th International Workshop on Computer Algebra in Scienti c Computing (CASC'2013), (V. P. Gerdt, V. Koepf, E. W. Mayr, and E. V. Vorozhtsov, editors), Lecture Notes in Computer Science, 8136, 273 287, Springer, Heidelberg (2013).

34. Fast Approximate Computations with Cauchy Matrices, Polynomials and Rational Functions", Proc. of the Ninth International Computer Science Symposium in Russia (CSR'2014), (E. A. Hirsch et al., editors), Moscow, Russia, June 2014, Lecture Notes in Computer Science (LNCS), 8476, pp. 287 300, Springer International Publishing, Switzerland (2014).

35. Nearly Optimal Computations with Structured Matrices" by Victor Y. Pan and Elias Tsigaridas, Proc. of the International Conference on Symbolic Numeric Computation (SNC 2014), (edited by S. Watt and J. Verschelde), 21 30, ACM Press, New York,
2014.

Also April 18, 2014, arXiv:1404.4768 [math.NA] and http://hal.inria.fr/hal-00980591

36. Transformations of Matrix Structures Work Again", Linear Algebra and Its Applications, 465, 1 32 (2015).

37. How Bad Are Vandermonde Matrices?", SIAM Journal of Matrix Analysis and Applications, 37, 2, 676 694 (2016).

38. Fast approximate computations with Cauchy matrices and polynomials", Math. of Computation, 86, 2799 2826, 2017. DOI: https://doi.org/10.1090/mcom/3204

8.2 UNIVARIATE POLYNOMIAL ROOT-FINDING AND FACTORIZATION

A BOOK

Numerical Methods for Roots of Polynomials" (by J. M. McNamee and V. Y. Pan), Part 2 (XXII + 718 pages), Elsevier (2013).

REVIEW ARTICLES AND BOOK CHAPTERS

1."Solving a Polynomial Equation: Some History and Recent Progress", SIAM Review,39, 2, 187 220 (1997).

2."Solving Polynomials with Computers", American Scientist, 86, 62 69 (JanuaryFebruary 1998).

3."Some Recent Algebraic/Numerical Algorithms", Electronic Procs. IMACS/ACA'98 (1998). Available at http:www-troja.fj .cvut.cz/aca98/sessions/approximate

4."Algebraic Algorithms" (by A. Diaz, I. Z. Emiris, E. Kaltofen and V. Y. Pan), Chapter16 in Handbook "Algorithms and Theory of Computations", pp. 16 1 to 16 27 (M. Atallah, editor), CRC Press Inc., Boca Raton, Florida (1999).

5.Root- nding with Eigen-solving" (by V. Y. Pan, D. Ivolgin, B. Murphy, R. E. Rosholt, Y. Tang, X. Wang, and X. Yan), pages 185 210 in Symbolic-Numeric Computation (Dongming Wang and Lihong Zhi, editors), Birkha ser, Basel/Boston (2007).

6.Algebraic and Numerical Algorithms" (by I. Z. Emiris, V. Y. Pan, and E. Tsigaridas), in Algorithms and Theory of Computations Handbook", Second Edition, Volume 1 (1016 pages): General Concepts and Techniques, pages 1 34 in Chapter 17 (Mikhail J. Atallah and Marina Blanton, editors), CRC Press Inc., Boca Raton, Florida (2009).

7.Algebraic Algorithms" (by I. Z. Emiris, V. Y. Pan, and E. Tsigaridas), Chapter 10 (pages from 10 1 to 10-40) of Computing Handbook (Third edition), Volume I: Computer Science and Software Engineering (Allen B. Tucker, Teo Gonzales, and Jorge L. Diaz-Herrera, editors), Taylor and Francis Group, 2014. Available at arXiv 1311.3731 [cs.DS]

RESEARCH PAPERS (in journals and refereed proceedings of conferences).

1. "On Application of Some Recent Techniques of the Design of Algebraic Algorithmsto the Sequential and Parallel Evaluation of the Roots of a Polynomial and to Some Other Numerical Problems", Computers and Math. (with Applications), 11, 9, 911 917 (1985).

2. "Fast and E cient Algorithms for Sequential and Parallel Evaluation of Polynomial Zeros and of Matrix Polynomials," Proc. 26th Ann. IEEE Symp. on Foundations of Computer Science (FOCS'85), 522 531, IEEE Computer Society Press, Los Angeles, California (1985).

3. "Algebraic Complexity of Computing Polynomial Zeros", Computers and Math. (withApplications), 14, 4, 285 304 (1987).

4. "Sequential and Parallel Complexity of Approximate Evaluation of Polynomial Zeros",Computers and Mathematics (with Applications), 14, 8, 591 622 (1987).

5. "Fast and E cient Parallel Evaluation of the Zeros of a Polynomial Having Only Real Zeros", Computers and Mathematics (with Applications), 17, 11, 1475 1480 (1989).

6. "New Resultant Inequalities and Complex Polynomial Factorization", Proc. 1st IsraelSymp. on Theory of Computing and Systems (ISTCS '92), Lecture Notes in Computer Science, 601, 122 136, Springer, Berlin (1992).

7. "New Resultant Inequalities and Complex Polynomial Factorization", SIAM Journalon Computing 23, 5, 934 950 (1994).

8. "New Techniques for Approximating Complex Polynomial Zeros", Proc. 5th Ann. ACM-SIAM Symp. on Discrete Algorithms (SODA'94), 260 270, ACM Press, New York, and SIAM Publications, Philadelphia (1994).

9. "Deterministic Improvement of Complex Polynomial Factorization Based on the Properties of the Associated Resultant", Computers and Math. (with Applications), 30, 2, 71 94 (1995).

10. "Optimal (up to Polylog Factors) Sequential and Parallel Algorithms for Approximating Complex Polynomial Zeros", Proc. 27th Ann. ACM Symposium on Theory of Computing (STOC'95), 741 750, ACM Press, New York (1995).

11. "Optimal and Nearly Optimal Algorithms for Approximating Polynomial Zeros", Computers and Math. (with Applications), 31, 12, 97 138 (1996).

12. "Computing $x^n \bmod p(x)$ and an Application to Splitting a Polynomial into Factors over a Fixed Disc", Journal of Symbolic Computations, 22, 377 380 (1996).

13. "On Isolation of Real and Nearly Real Zeros of a Univariate Polynomial and Its Splitting into Factors" (by V. Y. Pan, M.-h. Kim, A. Sadikou, X. Huang, and A. Zheng), J. of Complexity, 12, 572 594 (1996).

14. "Grae e's, Chebyshev-like, and Cardinal's Processes for Splitting a Polynomial intoFactors" (by D. Bini and V. Y. Pan), J. of Complexity, 12, 492 511 (1996).

15. Computing Matrix Eigenvalues and Polynomial Zeros Where the Output Is Real" (by D. Bini and V. Y. Pan), SIAM J. on Computing, 27, 4, 1099 1115 (1998).

16. Approximating Complex Polynomial Zeros: Modi ed Quadtree (Weyl's) Construction and Improved Newton's Iteration", J. of Complexity, 16, 1, 213 264 (2000).

17. Lifting/Descending Processes for Polynomial Zeros and Applications" (by B. Mourrain and V. Y. Pan), J. of Complexity, 16, 1, 265 273 (2000).

18. A New Proximity Test for Polynomial Zeros", Computers and Math. (with Applications), 41, 12, 1559 1560 (2001).

19. Univariate Polynomials: Nearly Optimal Algorithms for Factorization and Root nding", Proc. Intern. Symp. on Symbolic Algebraic Comp. (ISSAC'01), 253 267, ACM Press, NY (2001).

20. Nearly Optimal Algorithms for Univariate Polynomial Factorization and Root nding II: Computing a Basic Annulus for Splitting", Trudy of the Steklov Math. Institute of the Russian Academy of Science (in English), volume 235, pp. 211 223, Moscow (2001).

21. Nearly Optimal Algorithms for Numerical Univariate Polynomial Factorization and Root nding I: Splitting a Polynomial into Factors over an Annulus", Proceedings of the Smalefest 2000 (F. Cucker and M. Rojas, editors), Foundations of Computational Math. Series, 325 353, World Scienti c, New Jersey (2002).

22. Univariate Polynomials: Nearly Optimal Algorithms for Numerical Factorization and Root-Finding", J. of Symbolic Computation, 33, 5, 701 733 (2002).

23. Inverse Power and Durand-Kerner Iteration for Univariate Polynomial Root Finding" (by D. A. Bini, L. Gemignani and V. Y. Pan), Computers and Mathematics (with Applications), 47, 2/3, 447 459 (2004).

24. Improved Initialization of the Accelerated and Robust QR like Polynomial Root nding" (by D. A. Bini, L. Gemignani, and V. Y. Pan), Electronic Transactions on Numerical Analysis, 17, 195 205 (2004). (Proc. version in Proceedings of the 7th International Workshop on Computer Algebra in Scienti c Computing (CASC'04), St. Petersburg, Russia (2004), (edited by E. W. Mayr, V. G. Ganzha, and E. V. Vorozhtzov), 39-50, Technische Univ. M nchen, Germany (2004).)

25. Fast and Stable QR Eigenvalue Algorithms for Generalized Semiseparable Matrices and Secular Equation" (by D. A. Bini, L. Gemignani and V. Y. Pan), Numerische Mathematik, 3, 373 408 (2005)

26. Coe cient-free Adaptation of Polynomial Root- nders", Computers and Mathematics with Applications, 50, 263 269 (2005).

27. Amended DSeSC Power Method for Polynomial Root- nding", Computers and Mathematics with Applications, 49, 9 10, 1515 1524 (2005).

28. Real Root- nding" (by V. Y. Pan, G. Qian, B. Murphy, R. E. Rosholt, and Y. Tang),Proceedings of the Third International Workshop on Symbolic Numeric Computation (SNC 2007), July 2007, London, Ontario, Canada (Jan Verschelde and Stephen Watt, editors), 161 169, ACM Press, New York (2007).

29. Root Squaring with DPR1 Matrices", in Zapiski Nauchnykh Seminarov POMI (in English), volume 373 (edited by N. N. Vasiliev and A. M. Vershik), pp. 189 193 (2009).

30. Real and Complex Polynomial Root-Finding with Eigen-Solving and Preprocessing" (by V. Y. Pan and A.-L. Zheng), (edited by Stephen Watt), in Proc. International Symp. on Symbolic and Algebraic Computation (ISSAC'2010), pages 219 226, ACM Press, New York (July 2010).

31. Matrix Computations and Polynomial Root- nding with Preprocessing" (by V. Y. Pan, G. Qian, A.-L. Zheng, and Z. Chen), Linear Algebra and Its Applications, 434, 854 879 (2011).

32. New Progress in Real and Complex Polynomial Root-Finding" (by V. Y. Pan and A.-L. Zheng), Computers and Math. (with Applications), 61, 1305 1334 (2011).

33. Univariate Polynomial Root-Finding by Arming with Constraints", in Proc. International Symposium on Symbolic-Numerical Computations (SNC'2011), San Jose, California, 2011 (edited by Marc Moreno Masa), 112 121, ACM Press, New York (2011).

34. Root- nding by Expansion with Independent Constraints" (by V. Y. Pan and A.-L. Zheng), Computers and Mathematics with Applications, 62, 3164 3182 (2011).

35. E cient Polynomial Root-re ners: A Survey and New Record E ciency Estimate" (by J. M. McNamee and V. Y. Pan), Computers and Mathematics with Applications, 63, 239 254 (2012).

36. Root-re ning for a Polynomial Equation", Proceedings of the 14th International Workshop on Computer Algebra in Scienti c Computing (CASC'2012), (V. P. Gerdt, V. Koepf, E. W. Mayr, and E. V. Vorozhtsov, editors), Lecture Notes in Computer Science, 7442, 271 282, Springer, Heidelberg (2012).

37. Real and Complex Polynomial Root- nding via Eigen-solving and Randomization"(by V. Y. Pan, G. Qian, and A.-L. Zheng), Proceedings of the 14th International Workshop on Computer Algebra in Scienti c Computing (CASC'2012), (V. P. Gerdt, V. Koepf, E. W. Mayr, and E. V. Vorozhtsov, editors), Lecture Notes in Computer Science, 7442, 283 293, Springer, Heidelberg (2012).

38. On the Boolean Complexity of the Real Root Re nement" (by V. Y. Pan and E. P. Tsigaridas), in Proc. International Symp. on Symbolic and Algebraic Computations (ISSAC'2013), Boston, Massachusetts, June 2013 (M. Kauers, editor), 299 306, ACM Press, New York (2013).

39. Accelerated Approximation of the Complex Roots of a Univariate Polynomial" by Victor Y. Pan and Elias Tsigaridas, Proc. of the International Conference on Symbolic Numeric Computation (SNC 2014), (edited by S. Watt and J. Verschelde), pages 132 134, ACM Press, New York, 2014.

40. Real Polynomial Root- nding by Means of Matrix and Polynomial Iterations", by V. Y. Pan, Proceedings of the 16th International Workshop on Computer Algebra in Scienti c Computing (CASC'2014), (V.P. Gerdt, V. Koepf, W.M. Zeiler, and E.V. Vorozhtsov, editors), Lecture Notes in Computer Science, 8660, 335 349, Springer, Cham (2014). Also Tech. Report TR 20140011, PhD Program in Comp. Sci., Graduate Center, CUNY, 2014.

Available at http://www.cs.gc.cuny.edu/tr/techreport.php?id=476

41. Nearly Optimal Re nement of Real Roots of a Univariate Polynomial" by V. Y. Pan and E. P. Tsigaridas, J. of Symbolic Computation, 74, 181 204 (2016)

42. Real Root Isolation by Means of Root Radii Approximation" (by V. Y. Pan and L. Zhao), Proceedings of the 17th International Workshop on Computer Algebra in Scienti c Computing (CASC'2015), (V. P. Gerdt, V. Koepf, and E. V. Vorozhtsov, editors), Lecture Notes in Computer Science, 9301, 347 358, Springer, Heidelberg (2015), (also in arXiv:1501.05386 [math.NA] (36 pages, 12 tables), 22 January 2015, revised October 21, 2015.).

43. Real Polynomial Root nding by Means of Matrix and Polynomial Iterations" (by Victor Y. Pan and Liang Zhao), Theoretical Computer Science, Special Issue on Symbolic Numerical Algorithms (Stephen Watt, Jan Verschelde, and Lihong Zhi, editors), 681, 101-116 (2017), http://dx.doi.org/10.1016/j.tcs.2017.03.032.

44. Accelerated Approximation of the Complex Roots and Factors of a Univariate Polynomial" (by V. Y. Pan and E. P. Tsigaridas), Theoretical Computer Science, Special Issue on Symbolic Numerical Algorithms (Stephen Watt, Jan Verschelde, and Lihong Zhi, editors), 681, 138 145 (2017). http://dx.doi.org/10.1016/j.tcs.2017.03.030.

45. Simple and Nearly Optimal Polynomial Root Finding by Means of Root Radii Approximation", Springer International Publishing (I.S. Kotsireas and E. MartinezMoro, editors), Applications of Computer Algebra, Springer, Chapter 23, Proceedings in Mathematics and Statistics, 198 AG (2017). Chapter DOI.10.1007/978-3-31956932-1_23

46. Implementation of a Near-Optimal Complex Root Clustering Algorithm" (by R. Imbach, V.Y. Pan, C. Yap), Proc. of International Congress on Math Software (ICMS 2018), 235 244, 2018.

47. Old and New Nearly Optimal Polynomial Root-Finders", In: Proceedings of the 21st International Workshop on Computer Algebra in Scienti c Computing (CASC’2019), (M. England, W. Koepf, T.M. Sadikov, W.M. Seiler, and E. V. Vorozhtsov, editors), Lecture Notes in Computer Science, 11661, 393 411, Springer, Nature Switzerland (2019), doi: 10.1007/978-3-030-26831-2 and arXiv:1805.12042 Submitted on 30 May 2018

48. Root-Finding with Implicit De ation" (with R. Imbach, C., Yap, I.S. Kotsireas, V. Zaderman), In: Proceedings of the 21st International Workshop on Computer Algebra in Scienti c Computing (CASC’2019), (M. England, W. Koepf, T.M. Sadikov, W.M. Seiler, and E. V. Vorozhtsov, editors), Lecture Notes in Computer Science, 11661, 236 245, Springer, Nature Switzerland (2019), doi: 10.1007/978-3-030-26831-2 and arxiv:1606.01396, submitted on 21 May (2019)

49. New Progress in Univariate Polynomial Root- nding" (by R. Imbach, V. Y. Pan), Proceedings of ACM-SIGSAM ISSAC 2020, ACM ISBN 978-1-4503-7100-1/20/07 https://doi.org/10.1145/3373207.3403979

50. Acceleration of Subdivision Root-Finding for Sparse Polynomials", In: Computer Algebra in Scienti c Computing (CASC’20), Springer Nature Switzerland AG 2020, F. Boulier et al. (Eds.): CASC 2020, LNCS 12291, Ch. 27, pp. 1 17, 2020 https://doi.org/10.1007/978 − 3 − 030 − 60026 − 6_27

51. Faster Numerical Univariate Polynomial Root-Finding by Means of Subdivision Iterations" (by Qi Luan, Victor Y. Pan, Wongeun Kim, and Vitaly Zaderman), In: Computer Algebra in Scienti c Computing (CASC’20), Springer Nature Switzerland AG 2020, F. Boulier et al. (Eds.): CASC 2020, LNCS 12291, Ch. 25, pp. 1 16, 2020 https://doi.org/10.1007/978 − 3 − 030 − 60026 − 6_25

52. Root Radii and Subdivision for Polynomial Root-Finding" (by R. Imbach, V. Y. Pan), In: Computer Algebra in Scienti c Computing (CASC’21), Springer Nature Switzerland AG 2021, F. Boulier et al. (Eds.): CASC 2021, LNCS 12865, pp. 1 21, 2021. *https* : *//doi.org/*10.1007/978 − 3 − 030 − 85165 − 1_9

8.3 MULTIRIATE POLYNOMIAL ROOT-FINDING

RESEARCH PAPERS (in journals and refereed proceedings of conferences).

1. "Multidimensional Structured Matrices and Polynomial Systems" (by B. Mourrainand V. Y. Pan), Calcolo (Special Issue on Toeplitz Matrices: Structure, Algorithms and Applications), 33, 389 401 (1996).

2. "Techniques for Exploiting Structure in Matrix Formulae of the Sparse Resultant" (byI. Z. Emiris and V. Y. Pan), Calcolo (Special Issue on Toeplitz Matrices: Structure, Algorithms and Applications), 33, 353 369 (1996).

3. "Solving Special Polynomial Systems by Using Structured Matrices and AlgebraicResidues" (by B. Mourrain and V. Y. Pan), Proc. of the AMS-SIAM Conference on Foundation of Computational Math. (FoCM’97), (F. Cucker and M. Shub, editors), Rio-de-Janeiro, January 1997, 287 304, Springer (1997).

4. "The Structure of Sparse Resultant Matrices" (by I. Z. Emiris and V. Y. Pan), Proc.Annual ACM International Symposium on Symbolic and Algebraic Computations (ISSAC'97), 189 196, ACM Press, New York (1997).

5. Asymptotic Acceleration of Solving Polynomial Systems" (by B. Mourrain and V. Y. Pan), Proc. 30th Annual ACM Symp. on Theory of Computing (STOC'98), 488 496, ACM Press, New York (1998).

6. Controlled Iterative Methods for Solving Polynomial Systems" (by D. Bondyfalat, B. Mourrain, and V. Y. Pan), Proc. ACM Annual Intern. Symp. on Symbolic and Algebraic Comp. (ISSAC'98), 252 259, ACM Press, New York (1998).

7. Multivariate Polynomials, Duality and Structured Matrices" (by B. Mourrain and V. Y. Pan), J. of Complexity, 16, 1, 110 180 (2000).

8. Computation of a Speci ed Root of a Polynomial System of Equations Using Eigenvectors" (by D. Bondyfalat, B. Mourrain, and V. Y. Pan), Linear Algebra and Its Applications, 319, 193 209 (2000).

9. Asymptotic Acceleration of the Solution of Multivariate Polynomial Systems of Equations" (by B. Mourrain, V. Y. Pan and O. Ruatta), Proceedings of the Smalefest 2000 (F. Cucker and M. Rojas, editors), Foundations of Computational Math. Series, 267 294, World Scienti c, New Jersey (2002).

10. Symbolic and Numerical Methods for Exploiting Structure in Constructing Resultant Matrices" (by I. Z. Emiris and V. Y. Pan), Journal of Symbolic Computation, 33, 393 413 (2002).

11. Accelerated Solution of Multivariate Polynomial Systems of Equations" (by B. Mourrain, V.Y. Pan and O. Ruatta), SIAM J. on Computing, 32, 2, 435 454 (2003).

12. Improved Algorithms for Computing Determinants and Resultants" (by I. Z. Emiris and V. Y. Pan), J. of Complexity, 21, 1, 43 71 (2005).

8.4 MATRIX EIGEN-SOLVING

A BOOK CHAPTER

Root- nding with Eigen-solving" (by V. Y. Pan, D. Ivolgin, B. Murphy, R. E. Rosholt, Y. Tang, X. Wang, and X. Yan), pages 185 210 in Symbolic-Numeric Computation (Dongming Wang and Lihong Zhi, editors), Birkha ser, Basel/Boston (2007).

RESEARCH PAPERS (in journals and refereed proceedings of conferences).

1. "E cient Algorithms for the Evaluation of the Eigenvalues of (Block) Banded Toeplitz Matrices" (by D. Bini and V. Y. Pan), Mathematics of Computation, 50, 182, 431 448 (1988).

2. "Estimating the Extremal Eigenvalues of a Symmetric Matrix", Computers and Math.(with Applications), 20, 2, 17 22 (1990).

3. "Parallel Complexity of Tridiagonal Symmetric Eigenvalue Problem" (by D. Biniand V. Y. Pan), Proc. 2nd Ann. ACM-SIAM Symposium on Discrete Algorithms (SODA'91), 384 393, ACM Press, New York, and SIAM Publications, Philadelphia (1991).

4. "On the Evaluation of the Eigenvalues of a Banded Toeplitz Block Matrix" (by D. Biniand V. Y. Pan), J. of Complexity, 7, 408 424 (1991). "Practical Improvement of the Divide-and-Conquer Eigenvalue Algorithms" (by D. Bini and V. Y. Pan), Computing, 48, 109 123 (1992).

5. "Improving the Solution of the Symmetric Eigenvalue Problem and an Extension",Applied Mathematics Letters, 5, 6, 49 50 (1992).

6. "A New Algorithm for the Symmetric Tridiagonal Eigenvalue Problem" (by V. Y. Panand J. Demmel), Journal of Complexity, 9, 387 405 (1993).

7. "Weyl's Quadtree Algorithm for Unsymmetric Eigenvalue Problem", Applied Math.Letters, 8, 5, 87 88 (1995).

8. Computing Matrix Eigenvalues and Polynomial Zeros Where the Output Is Real" (by D. Bini and V. Y. Pan), SIAM J. on Computing, 27, 4, 1099 1115 (1998).

9. The Complexity of the Matrix Eigenproblem" (by V. Y. Pan and Z. Chen), Proc. 31st Annual ACM Symposium on Theory of Computing (STOC'99), 507 516, ACM Press, New York (1999).

10. A New Approach to Bisection Acceleration for the Symmetric Tridiagonal Eigenvalue Problem" (by V. Y. Pan and E. Linzer), Numerical Algorithms, 22, 13 39 (1999).

11. Computation of a Speci ed Root of a Polynomial System of Equations Using Eigenvectors" (by D. Bondyfalat, B. Mourrain, and V. Y. Pan), Linear Algebra and Its Applications, 319, 193 209 (2000).

12. Inverse Power and Durand-Kerner Iteration for Univariate Polynomial Root-Finding" (by D. A. Bini, L. Gemignani and V. Y. Pan), Computers and Mathematics (with Applications), 47, 2/3, 447 459 (2004).

13. Improved Initialization of the Accelerated and Robust QR-like Polynomial Rootnding" (by D. A. Bini, L. Gemignani, and V. Y. Pan), Electronic Transactions on Numerical Analysis, 17, 195 205 (2004). (Proc. version in Proceedings of the 7th International Workshop on Computer Algebra in Scienti c Computing (CASC'04), St. Petersburg, Russia (2004), (edited by E. W. Mayr, V. G. Ganzha, and E. V. Vorozhtzov), 39-50, Technische Univ. M nchen, Germany (2004).)

14. Can the TPR1 Structure Help Us to Solve the Algebraic Eigenproblem?", in Proc. of the 16th Ann. ACM-SIAM Symposium on Discrete Algorithms (SODA'05), 1069 1078, ACM Press, New York, and SIAM Publications, Philadelphia (2005).

15. Fast and Stable QR Eigenvalue Algorithms for Generalized Semiseparable Matrices and Secular Equation" (by D. A. Bini, L. Gemignani and V. Y. Pan), Numerische Mathematik, 3, 373 408 (2005).

16. Null Space and Eigenspace Computation with Additive Preconditioning" (by V. Y. Pan and X. Yan), Proceedings of the Third International Workshop on Symbolic Numeric Computation (SNC'2007), July 2007, London, Ontario, Canada (Jan Verschelde and Stephen Watt, editors), 170 179, ACM Press, New York (2007).

17. Eigen-solving via Reduction to DPR1 Matrices" (by V. Y. Pan, B. Murphy, R. Rosholt, Y. Tang, X. Wang, and A. Zheng), Computers and Mathematics with Applications, 56, 166 171 (2008).

18. Additive Preconditioning, Eigenspaces, and the Inverse Iteration" (by V. Y. Pan and X. Yan), Linear Algebra and Its Applications, 430, 186 203 (2009).

19. Real Polynomial Root nding by Means of Matrix and Polynomial Iterations" (by Victor Y. Pan and Liang Zhao), Theoretical Computer Science, Special Issue on Symbolic Numerical Algorithms (Stephen Watt, Jan Verschelde, and Lihong Zhi, editors), 681, 101-116 (2017), http://dx.doi.org/10.1016/j.tcs.2017.03.032.

8.5 MATRIX MULTIPLICATION

A BOOK

How to Multiply Matrices Faster", Lecture Notes in Computer Science, vol. 179 (XI + 212 pages), Springer, Berlin (1984).

REVIEW ARTICLES

"How Can We Speed Up Matrix Multiplication?", SIAM Review, 26, 3, 393-415 (1984).

RESEARCH PAPERS (in journals and refereed proceedings of conferences).

1."On Schemes for the Evaluation of Products and Inverses of Matrices", UspekhiMatematicheskikh Nauk (in Russian), 27, 5 (167), 249 250 (1972). Also see translation into English: "Better Late Than Never: Filling a Void in the History of Fast Matrix Multiplication and Tensor Decompositions", arXiv:1411.1972 (6 Nov 2014)

2."Strassen's Algorithm Is Not Optimal. Trilinear Technique of Aggregating, Uniting and Canceling for Constructing Fast Algorithms for Matrix Multiplication", Proceedings of the 19th Annual IEEE Symposium on Foundations of Computer Science (FOCS'78), 166 176, IEEE Computer Society Press, Long Beach, California (1978).

3."Fields Extension and Trilinear Aggregating, Uniting and Canceling for the Acceleration of Matrix Multiplication", Proceedings of the 20th Annual IEEE Symposium on Foundations of Computer Science (FOCS'79), 28 38, IEEE Computer Society Press, Long Beach, California (1979).

4."New Fast Algorithms for Matrix Operations", SlAM J. on Computing, 9, 2, 321 342 (1980).

5."New Combinations of Methods for the Acceleration of Matrix Multiplications", Computers and Mathematics (with Applications), 7, 73 125 (1981).

6."A Uni ed Approach to the Analysis of Bilinear Algorithms", Journal of Algorithms,2, 301 310 (1981).

7."The Bit Operation Complexity of Matrix Multiplication and All Pair Shortest PathProblem", Computers and Math. (with Applications), 7, 5, 431 438 (1981).

8."Trilinear Aggregating with Implicit Canceling for a New Acceleration of Matrix Multiplication", Computers and Math. (with Applications), 8, 1, 23 34 (1982).

9."Fast Matrix Multiplication without APA-Algorithms", Computers and Math. (withApplications), 8, 5, 343 366 (1982).

10. "Trilinear Aggregating is the Basis for the Asymptotically Fastest Known Algorithmsfor Matrix Multiplication", Conference Record, Second Conference on Foundations of Software Technology and Theoretical Computer Science (FST and TCS'82), 321 337, Indian Institute of Technology, Bangalore, India (December 1982).

11. "Trilinear Aggregating is the Basis for the Asymptotically Fastest Known Algorithmsfor Matrix Multiplication" (extended abstract), Methods of Operations Research, 45, 493 494 (1983)

12. "Trilinear Aggregating and the Recent Progress in the Asymptotic Acceleration ofMatrix Operations", Theoretical Computer Science, 33, 117 138 (1984).

13. "The Additive and Logical Complexities of Linear and Bilinear Algorithms", J. ofAlgorithms, 4, 1 34 (1983).

14. "The Trade-o Between the Additive Complexity and Asynchronicity of Linear and Bilinear Algorithms", Information Processing Letters, 22, 11 14 (1986)

15. "The Bit Complexity of Matrix Multiplication and of the Related Computations inLinear Algebra. The Segmented λ-algorithms", Computers and Math. (with Applications), 11, 9, 919 928 (1985).

16. "On Practical Algorithms for Accelerated Matrix Multiplication" (by J. Laderman, V.Y. Pan and H. X. Sha), Linear Algebra and Its Applications, 162 164, 557 588 (1992).

17. "Fast Rectangular Matrix Multiplication and Improving Parallel Matrix Computations" (by X. Huang and V. Y. Pan), Proc. Annual ACM International Symposium on Parallel Algebraic and Symbolic Computation (PASCO'97), 11 23, ACM Press, New York (1997).

18. Fast Rectangular Matrix Multiplication and Applications" (by X. Huang and V. Y. Pan), J. of Complexity, 14, 257 299 (1998).

19. Parallel Matrix Multiplication on a Linear Array with a Recon gurable Pipelined Bus System" (by K. Li and V. Y. Pan), Proc. 13th Intern. Parallel Processing Symp. and 10th Symp. on Parallel and Distributed Computing (IPPS/SPDP 1999), 31 35, IEEE Computer Soc. Press, Los Alamitos, California (April 1999).

20. Fast Rectangular Matrix Multiplication and Some Applications" (by S. Ke, B. Zeng, W. Han, V. Y. Pan), Science in China, Series A: Mathematics, 51, 3, 389 406, 2008.

8.6 NEWTON'S ITERATION WITH GENERAL AND STRUCTURED MATRICES

RESEARCH PAPERS (in journals and refereed proceedings of conferences).

1. "E cient Parallel Solution of Linear Systems" (by V. Y. Pan and J. Reif), Proc. 17th Ann. ACM Symp. on Theory of Computing (STOC'85), 143 152, ACM Press, New York (1985).

2. "Fast and E cient Parallel Algorithms for the Exact Inversion of Integer Matrices", Proc. Fifth Conf. on Foundations of Software Technology and Theoretical Computer Science (FST and TCS'85), (edited by K. V. Nori), Lecture Notes in Computer Science, 206, 504 521, Springer, Berlin (1985).

3. "Complexity of Parallel Matrix Computations", Theoretical Computer Science, 54,65 85 (1987).

4. "An Improved Newton Iteration for the Generalized Inverse of a Matrix, with Applications" (by V. Y. Pan and R. Schreiber), SIAM J. on Scienti c and Statistical Computing, 12, 5, 1109 1131 (1991).

5. "Parallel Solution of Toeplitz-like Linear Systems", J. of Complexity, 8, 1 21 (1992).

6. "Parametrization of Newton's Iteration for Computations with Structured Matricesand Applications", Computers and Mathematics (with Applications), 24, 3, 61 75 (1992).

7. "Concurrent Iterative Algorithm for Toeplitz-like Linear Systems", IEEE Trans. onParallel and Distributed Systems, 4, 5, 592 600 (1993).

8. "Newton's Iteration for Inversion of Cauchy-like and Other Structured Matrices" (by V. Y. Pan, A. Zheng, X. Huang, and O. Dias), J. of Complexity, 13, 108 124 (1997).

9. Newton's Iteration for Structured Matrices and Linear Systems of Equations" (by V. Y. Pan, S. Branham, R. Rosholt, and A. Zheng), SIAM volume on Fast Reliable Algorithms for Matrices with Structure (T. Kailath and A. H. Sayed, editors), ch. 7, pp. 189 210, SIAM Publications, Philadelphia (1999).

10. Approximating Complex Polynomial Zeros: Modi ed Quadtree (Weyl's) Construction and Improved Newton's Iteration", J. of Complexity, 16, 1, 213 264 (2000). Parallel Complexity of Computations with General and Toeplitz-like Matrices Filled with Integers", SIAM J. on Computing, 30, 1080 1125 (2000).

11. A Homotopic Residual Correction Process", Proc. of the Second Conference on Numerical Analysis and Applications (L. Vulkov, J. Wasniewsky and P. Yalamov, editors), Lecture Notes in Computer Science, 1988, 644 649, Springer, Berlin (2001).

12. Newton's Iteration for the Inversion of Structured Matrices" (by V. Y. Pan and Y. Rami), in "Structured Matrices: Recent Developments in Theory and Computation", 79 90 (edited by D. Bini, E. Tyrtyshnikov and P. Yalamov), Nova Science Publishers, USA (2001).

13. Iterative Inversion of Structured Matrices" (by V. Y. Pan, M. Van Barel, X. Wang and G. Codevico), Theoretical Computer Science, 315, 2 3, 581 592 (2004).

14. The Least-Squares Compression Policy for Newton-like Iteration for Structured Matrices" (by G. Codevico, V. Y. Pan, M. Van Barel, X. Wang, and A.-L. Zheng) in Proceedings of the 6th

International Mathematica Symposium (IMS 2004), (edited by C. Jacob and P. Mitic), Ban , Alberta, Canada (August 2004).

15. Newton-like Iteration Based on Cubic Polynomials for Structured Matrices" (by G. Codevico, V. Y. Pan, and M. Van Barel), Numerical Algorithms, 36, 365 380 (2004).

16. Homotopic Residual Correction Algorithms for General and Structures Matrices" (by V. Y. Pan, M. Kunin, R. Rosholt, and H. Kodal), Math. of Computation, 75, 345 368 (2006).

17. New Homotopic/Factorization and Symmetrization Techniques for Newton's and Newton's Structured Iteration", Computers Math. with Applications, 54, 721 729 (2007).

18. Newton's Iteration for Matrix Inversion, Advances and Extensions", pp. 364 381, in MATRIX METHODS: THEORY, ALGORITHMS AND APPLICATIONS (dedicated to the Memory of Gene Golub, edited by Vadim Olshevsky and Eugene Tyrtyshnikov), World Scienti c Publishing, New Jersey, ISBN-13 978-981-283-601-4, ISBN-10-981283-601-2 (2010).

8.7 MULTIGRID ALGORITHMS

RESEARCH PAPERS (in journals and refereed proceedings of conferences).

1."Methods of Aggregations" (by W. L. Miranker and V. Y. Pan), Linear Algebra andIts Applications, 29, 231 257 (1980).

2."Compact Multigrid" (by V. Y. Pan and J. Reif), SIAM J. on Scienti c and StatisticalComputing, 13, 1, 119 127 (1992).

3."On the Bit-Complexity of Discrete Solution of PDEs: Compact Multigrid" (by V.Y. Pan and J. Reif), Proc. 17th International Colloquium on Automata, Languages and Programming (ICALP'90), Springer's Lecture Notes in Computer Science, 443, 612 625 (1990).

4."The Bit-Complexity of the Discrete Solution of PDEs: Compact Multigrid" (by V.Y. Pan and J. Reif), Computers and Math. (with Applications), 20, 2, 9 16 (1990).
5."Compact Multigrid" (by V. Y. Pan and J. Reif), SIAM J. on Scienti c and StatisticalComputing, 13, 1, 119 127 (1992).

6."Generalized Compact Multigrid" (by V. Y. Pan and J. Reif), Computers and Math. (with Applications), 25, 9, 3 5 (1993).

8.8 COMPUTATION OF A MATRIX DETERMINANT

A BOOK

Polynomial and Matrix Computations", Volume 1: Fundamental Algorithms" (XVI + 415 pages) (by D. Bini and V. Y. Pan), in the series Progress in Theoretical Computer Science (R.V. Book editor), Birkh user, Boston (1994).

RESEARCH PAPERS (in journals and refereed proceedings of conferences).

1. "Computing the Determinant and the Characteristic Polynomial of a Matrix via Solving Linear Systems of Equations", Information Processing Letters, 28, 2, 71 75 (1988).

2. "Computing Exact Geometric Predicates Using Modular Arithmetic with Single Precision" (by H. Br nnimann, I. Z. Emiris, V. Y. Pan and S. Pion), Proc. 13th Ann. ACM Symp. on Computational Geometry, 174 182, ACM Press, New York (1997).

3. "Algebraic and Numerical Techniques for the Computation of Matrix Determinants"(by V. Y. Pan, Y. Yu, and C. Stewart), Computers and Mathematics (with Applications), 34, 1, 43 70 (1997).

4. Sign Determination in Residue Number Systems" (by H. Br nnimann, I. Z. Emiris, V. Y. Pan and S. Pion), Theoretical Computer Science, 210, 1, 173 197 (1999).

5. On Theoretical and Practical Acceleration of Randomized Computation of the Determinant of an Integer Matrix", Zapiski Nauchnykh Seminarov POMI (in English), Vol. 316, 163 187, St. Petersburg, Russia (2004).

6. Improved Algorithms for Computing Determinants and Resultants" (by I. Z. Emiris and V. Y. Pan), J. of Complexity, 21, 1, 43 71 (2005).

7. Schur Aggregation for Linear Systems and Determinants" (by V. Y. Pan, D. Grady, B. Murphy, G. Qian, R. E. Rosholt, and A. Ruslanov), Theoretical Computer Science, Special Issue on Symbolic-Numerical Algorithms (D.A. Bini, V. Y. Pan, and J. Verschelde, editors), 409, pp. 255 268 (2008).

8.9 OTHER GENERAL MATRIX COMPUTATIONS

A BOOK

Polynomial and Matrix Computations", Volume 1: Fundamental Algorithms" (XVI + 415 pages) (by D. Bini and V. Y. Pan), in the series Progress in Theoretical Computer Science (R.V. Book editor), Birkh user, Boston (1994).

REVIEW ARTICLES AND BOOK CHAPTERS

1."Linear Systems of Algebraic Equations", in Encyclopedia of Physical Sciences andTechnology, 7, 304 329 (1987), (rst edition, Marvin Yelles, editor); 8, 779 804 (1992) (second edition), and 8, 617 642 (2001) (third edition, Robert A. Meyers, editor), Academic Press, San Diego, California.

2."Complexity of Algorithms for Linear Systems of Equations", in Computer Algorithmsfor Solving Linear Algebraic Equations (The State of the Art), (E. Spedicato, editor), NATO ASI Series, Series F: Computer and Systems Sciences, 77, 27 56, Springer, Berlin (1991) and Academic Press, Dordrecht, the Netherlands (1992).

3."Complexity of Computations with Matrices and Polynomials," SIAM Review, 34, 2,225 262 (1992).

4."Algebraic Algorithms" (by A. Diaz, E. Kaltofen and V. Y. Pan), Chapter 10 in theComputer Science and Engineering Handbook (Allen B. Tucker, Jr., editor), 226-249, CRC Press Inc., Boca Raton, Florida (1997) and Chapter 8 in the Computer Science and Engineering Handbook (Allen B. Tucker, editor), pp. 8 1 to 8 24, Chapman and Hall/CRC Press, 2004.

5."Algebraic Algorithms" (by A. Diaz, I. Z. Emiris, E. Kaltofen and V. Y. Pan), Chapter16 in Handbook "Algorithms and Theory of Computations", pp. 16 1 to 16 27 (M. Atallah, editor), CRC Press Inc., Boca Raton, Florida (1999).

6.Algebraic and Numerical Algorithms" (by I. Z. Emiris, V. Y. Pan, and E. Tsigaridas), in Algorithms and Theory of Computations Handbook", Second Edition, Volume 1 (1016 pages): General Concepts and Techniques, pages 1 34 in Chapter 17 (Mikhail J. Atallah and Marina Blanton, editors), CRC Press Inc., Boca Raton, Florida (2009).

7.Algebraic Algorithms" (by I. Z. Emiris, V. Y. Pan, and E. Tsigaridas), Chapter 10 (pages from 10 1 to 10-40) of Computing Handbook (Third edition), Volume I: Computer Science and Software Engineering (Allen B. Tucker, Teo Gonzales, and Jorge L. Diaz-Herrera, editors), Taylor and Francis Group, 2014. Available at arXiv 1311.3731 [cs.DS]

RESEARCH PAPERS (in journals and refereed proceedings of conferences).

1."How Fast Can We Solve a System of Linear Inequalities in the Worst and in theAverage Case?", Methods of Operations Research, 51, 107 118 (1984).

2."E cient Parallel Solution of Linear Systems" (by V. Y. Pan and J. Reif), Proc. 17th Ann. ACM Symp. on Theory of Computing (STOC'85), 143 152, ACM Press, New York (1985).

3."Fast Finite Methods for a System of Linear Inequalities", Computers and Math.(with Applications), 11, 4, 355 394 (1985).

4."On the Complexity of a Pivot Step of the Revised Simplex Algorithm", Computersand Math. (with Applications), 11, 11, 1127 1140 (1985).

5."Fast and E cient Linear Programming and Linear Least-Squares Computations" (by V. Y. Pan and J. Reif), Computers and Mathematics (with Applications), 12A, 12, 1217 1227 (1986).
6."E cient Parallel Linear Programming" (by V. Y. Pan and J. Reif), Operations Research Letters, 5, 3, 127 135 (1986).

7."Complexity of Parallel Matrix Computations", Theoretical Computer Science, 54,65 85 (1987).

8."Parallel Evaluation of the Determinant and of the Inverse of a Matrix" (by Z. Galiland V. Y. Pan), Information Processing Letters, 30, 41 45 (1989).

9."Parallel Least-Squares Solution of General and Toeplitz-like Linear Systems", Proc.2nd Ann. ACM Symp. on Parallel Algorithms and Architecture (SPAA'90), 244 253, ACM Press, New York (1990).

10. "The Modi ed Barrier Function Method for Linear Programming and Its Extensions",Computers and Math. (with Applications), 20, 3, 1 14 (1990).

11. "Processor E cient Parallel Solution of Linear Systems over an Abstract Field" (by E. Kaltofen and V. Y. Pan), Proc. 3rd Ann. ACM Symp. on Parallel Algorithms and Architectures (SPAA'91), 180 191, ACM Press, New York (1991).

12. "Processor-E cient Parallel Solution of Linear Systems II. The Positive Characteristic and Singular Cases" (by E. Kaltofen and V. Y. Pan), Proc. of 33rd Ann. IEEE Symp. on Foundations of Computer Science (FOCS'92), 714 723, IEEE Computer Society Press, Los Alamitos, California (1992).

13. "Improved Parallel Solution of a Triangular Linear System", Computers and Math.(with Applications), 27, 11, 41 43 (1994).

14. Null Space and Eigenspace Computation with Additive Preconditioning" (by V. Y. Pan and X. Yan), Proceedings of the Third International Workshop on Symbolic Numeric Computation (SNC'2007), July 2007, London, Ontario, Canada (Jan Verschelde and Stephen Watt, editors), 170 179, ACM Press, New York (2007).

15. The Schur Aggregation for Solving Linear Systems of Equations" (by V. Y. Pan, B. Murphy, R. E. Rosholt, and M. Tabanjeh), Proceedings of the Third International Workshop on Symbolic Numeric Computation (SNC'2007), July 2007, London, Ontario, Canada (Jan Verschelde and Stephen Watt, editors), 180 188, ACM Press, New York (2007).

16. Additive Preconditioning and Aggregation in Matrix Computations" (by V. Y. Pan, D. Ivolgin, B. Murphy, R. E. Rosholt, I. Taj-Eddin, Y. Tang, and X. Yan), Computers and Mathematics with Applications, 55, 8, 1870 1886 (2008).

17. Additive Preconditioning for Matrix Computations" (by V. Y. Pan, D. Ivolgin, B. Murphy, R. E. Rosholt, Y. Tang, and X. Yan), in Proc. of the Third International Computer Science Symposium in Russia (CSR'2008), Lecture Notes in Computer Science (LNCS), 5010, 372 383 (2008).

18. Degeneration of Integer Matrices Modulo an Integer" (by V. Y. Pan and X. Wang), Linear Algebra and Its Applications, 429, 2113 2130 (2008).

19. Schur Aggregation for Linear Systems and Determinants" (by V. Y. Pan, D. Grady, B. Murphy, G. Qian, R. E. Rosholt, and A. Ruslanov), Theoretical Computer Science, Special Issue on Symbolic-Numerical Algorithms (D.A. Bini, V. Y. Pan, and J. Verschelde, editors), 409, pp. 255 268 (2008).

20. Preconditioning, Randomization, Solving Linear Systems, Eigen-Solving, and RootFinding" (by V. Y. Pan, G. Qian, and A.-L. Zheng), Proc. International Symposium on Symbolic- Numerical Computations (SNC'2009), Kyoto, Japan, August 2009, (edited by Hiroshi Kai and Hiroshi Sekigawa), pp. 5 6, ACM Press, New York(2009).

21. Solving Homogeneous Linear Systems with Randomized Preprocessing" (by V. Y. Pan and G. Qian), Linear Algebra and Its Applications, 432, 3272 3318 (2010).

22. Additive Preconditioning for Matrix Computations" (by V. Y. Pan, D. Ivolgin, B. Murphy, R. E. Rosholt, Y. Tang, and X. Yan), Linear Algebra and Its Applications, 432, 1070 1089 (2010).

23. Advancing Matrix Computations with Randomized Preprocessing" (by V. Y. Pan, G. Qian, and A.-L. Zheng), in Proc. of the Fifth International Computer Science Symposium in Russia (CSR'2010), Kazan, Russia, June 2010 (Farid Ablaev and Ernst W. Mayr, editors), Lecture Notes in Computer Science (LNCS), pages 303 314, Springer, Berlin (2010).

24. Matrix Computations and Polynomial Root- nding with Preprocessing" (by V. Y. Pan, G. Qian, A.-L. Zheng, and Z. Chen), Linear Algebra and Its Applications, 434, 854 879 (2011).

25. Nearly Optimal Solution of Rational Linear Systems of Equations with Symbolic Lifting and Numerical Initialization", Computers and Mathematics with Applications, 62, 1685 1706 (2011).

26. Solving Linear Systems of Equations with Randomization, Augmentation and Aggregation" (by V. Y. Pan and G. Qian), Linear Algebra and Its Applications", 437, 2851 2876 (2012).

27. Randomized Preconditioning versus Pivoting" (by V. Y. Pan, G. Qian, and A.-L.Zheng), Linear Algebra and Its Applications, 438, 4, 1883 1889 (2013).

28. New Studies of Randomized Augmentation and Additive Preprocessing" (by Victor Y. Pan and Liang Zhao), Linear Algebra Appl., 512, 1, 256 305 (2017) http://dx.doi.org/10.1016/j.laa.2016.09.035 arxiv 1412.5864 math.NA (38 pages), December 18, 2014, revised on July 10, 2015.

29. "Numerically Safe Gaussian Elimination with No Pivoting" (by Victor Y. Pan andLiang Zhao), Linear Algebra and Its Applications, 527, 349 383 (2017).
http://dx.doi.org/10.1016/j.laa.2017.04.007

30. An E cient Computation of Generalized Inverse of a Matrix" (by V.Y. Pan, F. Soleymani and L. Zhao), Applied Mathematics and Computation, 316, 89 101, 2018. http://dx.doi.org/10.1016/j.amc.2017.08.010

8.10 COMPUTATIONS WITH DATA SPARSE (SPARSE AND/OR STRUCTURED) MATRICES

BOOKS

1. Polynomial and Matrix Computations", Volume 1: Fundamental Algorithms" (XVI + 415 pages) (by D. Bini and V. Y. Pan), in the series Progress in Theoretical Computer Science (R.V. Book editor), Birkh user, Boston (1994).
2. Structured Matrices and Polynomials: Uni ed Superfast Algorithms" (XXV + 278 pages), Birkh user/Springer, Boston/New York (June 2001).

CHAPTERS IN BOOKS AND SURVEY ARTICLES

1. "Complexity of Computations with Matrices and Polynomials," SIAM Review, 34, 2,225 262 (1992).

2. "Parallel Solution of Sparse Linear and Path Systems", in Synthesis of Parallel Algorithms (J. H. Reif, editor), Chapter 14, pp. 621 678. Morgan Kaufmann publishers, San Mateo, CA (1993).

3. "Algebraic Algorithms" (by A. Diaz, E. Kaltofen and V. Y. Pan), Chapter 10 in theComputer Science and Engineering Handbook (Allen B. Tucker, Jr., editor), 226-249, CRC Press Inc., Boca Raton, Florida (1997) and Chapter 8 in the Computer Science and Engineering Handbook (Allen B. Tucker, editor), pp. 8 1 to 8 24, Chapman and Hall/CRC Press, 2004.

4. "Computational Complexity of Solving Large Sparse and Large Special Linear Systemsof Equations", pp. 1 24, in "Algorithms for Large Scale Linear Algebraic Systems: Applications in

Science and Engineering" (G. Winter Althaus and E. Spedicato, editors), NATO Advanced Science Institute Series, Kluwer Academic Publishers, Dordrecht, The Netherlands (1998).

5. "Some Recent Algebraic/Numerical Algorithms", Electronic Procs. IMACS/ACA'98 (1998). Available at http:www-troja.fj .cvut.cz/aca98/sessions/approximate

6. "Algebraic Algorithms" (by A. Diaz, I. Z. Emiris, E. Kaltofen and V. Y. Pan), Chapter16 in Handbook "Algorithms and Theory of Computations", pp. 16 1 to 16 27 (M. Atallah, editor), CRC Press Inc., Boca Raton, Florida (1999).

7. "Fast Fourier Transform and Its Applications" (by I. Z. Emiris and V. Y. Pan), Chapter17 in Handbook "Algorithms and Theory of Computations", pp. 17 1 to 17 30 (M. Atallah, editor), CRC Press Inc., Boca Raton, Florida (1999).

8. Algebraic and Numerical Algorithms" (by I. Z. Emiris, V. Y. Pan, and E. Tsigaridas), in Algorithms and Theory of Computations Handbook", Second Edition, Volume 1 (1016 pages): General Concepts and Techniques, pages 1 34 in Chapter 17 (Mikhail J. Atallah and Marina Blanton, editors), CRC Press Inc., Boca Raton, Florida (2009).

9. "Fast Fourier Transform and Its Applications" (by I. Z. Emiris and V. Y. Pan), in Algorithms and Theory of Computations Handbook", Second Edition, Volume 1 (1016 pages): General Concepts and Techniques, pages 1 31 in Chapter 18 (Mikhail J. Atallah and Marina Blanton, editors), CRC Press Inc., Boca Raton, Florida (2009).

10. Algebraic Algorithms" (by I. Z. Emiris, V. Y. Pan, and E. Tsigaridas), Chapter 10 (pages from 10 1 to 10-40) of Computing Handbook (Third edition), Volume I: Computer Science and Software Engineering (Allen B. Tucker, Teo Gonzales, and Jorge L. Diaz-Herrera, editors), Taylor and Francis Group, 2014. Available at arXiv 1311.3731 [cs.DS]

RESEARCH PAPERS (in journals and refereed proceedings of conferences).

1. "Polynomial Division and Its Computational Complexity" (by D. Bini and V. Y. Pan),Journal of Complexity, 2, 179 203 (1986).

2. "Complexity of Parallel Matrix Computations", Theoretical Computer Science, 54,65 85 (1987).

3. "Some Polynomial and Toeplitz Matrix Computations" (by V. Y. Pan and J. Reif),Proc. 28th Ann. IEEE Symp. on Foundations of Computer Science (FOCS'87), 173 184, IEEE Computer Society Press, Los Angeles, California (1987).

4. "E cient Algorithms for the Evaluation of the Eigenvalues of (Block) Banded Toeplitz Matrices" (by D. Bini and V. Y. Pan), Mathematics of Computation, 50, 182, 431 448 (1988).

5. "On Some Computations with Dense Structured Matrices", Proc. Annual ACMSIGSAM International Symposium on Symbolic and Algebraic Computation (ISSAC'89), 34 42, ACM Press, New York (1989).

6. "Fast and E cient Parallel Inversion of Toeplitz and Block Toeplitz Matrices", Operator Theory: Advances and Applications, 40, 359 389, Birkh user, Basel, Switzerland (1989).

7. "Parallel Least-Squares Solution of General and Toeplitz-like Linear Systems", Proc.2nd Ann. ACM Symp. on Parallel Algorithms and Architecture (SPAA'90), 244 253, ACM Press, New York (1990).

8. "On Computations with Dense Structured Matrices", Math. Comp., 55, 191, 179 190 (1990).

9. "Parallel Complexity of Tridiagonal Symmetric Eigenvalue Problem" (by D. Bini andV. Y. Pan), Proc. 2nd Ann. ACM-SIAM Symp. on Discrete Algorithms (SODA'91), 384 393, ACM Press, New York, and SIAM Publications, Philadelphia (1991).

10. "On the Evaluation of the Eigenvalues of a Banded Toeplitz Block Matrix" (by D.Bini and V. Y. Pan), J. of Complexity, 7, 408 424 (1991).

11. "Parametrization of Newton's Iteration for Computations with Structured Matricesand Applications", Computers and Mathematics (with Applications), 24, 3, 61 75 (1992).

12. "A Fast, Preconditioned Conjugate Gradient Toeplitz Solver" (by V. Y. Pan andR. Schreiber), Computers and Math. (with Applications), 24, 7, 17 24 (1992). 96. "Practical Improvement of the Divide-and-Conquer Eigenvalue Algorithms" (by D. Bini and V. Y. Pan), Computing, 48, 109 123 (1992).

13. "Improved Parallel Computations with Toeplitz-like and Hankel-like Matrices" (by D.Bini and V. Y. Pan), Proc. of Annual ACM-SIGSAM International Symposium on Symbolic and Algebraic Computation (ISSAC'93), 193 200, ACM Press, New York (1993).

14. "The Power of Combining the Techniques of Algebraic and Numerical Computing: Improved Approximate Multipoint Polynomial Evaluation and Improved Multipole Algorithms" (by V. Y. Pan, J. Reif, and S. Tate), Proc. of 33rd Ann. IEEE Symp. on Foundations of Computer Science (FOCS '92), 703 713, IEEE Computer Society Press, Los Alamitos, California (1992).

15. "Concurrent Iterative Algorithm for Toeplitz-like Linear Systems", IEEE Trans. onParallel and Distributed Systems, 4, 5, 592 600 (1993).

16. "Fast and E cient Parallel Solution of Sparse Linear Systems" (by V. Y. Pan and J. Reif), SIAM J. on Computing, 22, 6, 1227 1250 (1993).

17. "Decreasing the Displacement Rank of a Matrix", SIAM J. on Matrix Analysis, 14, 1,118 121 (1993).

18. "A New Algorithm for the Symmetric Tridiagonal Eigenvalue Problem" (by V. Y. Panand J. Demmel), Journal of Complexity, 9, 387 405 (1993).

19. "Improved Parallel Computations with Toeplitz-like and Hankel-like Matrices" (by D.Bini and V. Y. Pan), Linear Algebra and Its Applications, 188, 189, 3 29 (1993).

20. "Optimum Parallel Computations with Band Matrices" (by V. Y. Pan, I. Sobzeand A. Atinkpahoun), Proc. 5th Ann. ACM-SIAM Symp. on Discrete Algorithms (SODA'94), 649 658, ACM Press, New York, and SIAM Publications, Philadelphia

(1994).

21. "Parallel Solution of Toeplitz and Toeplitz-like Linear Systems over Fields of SmallPositive Characteristic" (by E. Kaltofen and V. Y. Pan), Proceedings of First Intern. Symposium on Parallel Symbolic Computation (PASCO'94), Linz, Austria (Sept. 1994), Lecture Notes Series in Computing, 5, 225 233, World Scienti c Publishing Company, Singapore (1994).

22. "Parallel Computation of a Krylov Matrix for a Sparse and Structured Input", Mathematical and Computer Modeling, 21, 11, 97 99 (1995).

23. "On Parallel Computations with Band Matrices" (by V. Y. Pan, I. Sobze, and A.Atinkpahoun), Information and Computation, 120, 2, 237 250 (1995).

24. "A Fast, Preconditioned Conjugate Gradient Toeplitz and Toeplitz-like Solver" (by V.Y. Pan, A. Zheng, O. Dias, X. Huang), Computers and Math. (with Applications), 30, 8, 57 63 (1995).

25. "A New Approach to Parallel Computation of Polynomial GCDs and to Related Parallel Computations over Fields and Integer Rings", Proc. 7th Ann. ACM-SIAM Symposium on Discrete Algorithms (SODA'96), 518 527, ACM Press, New York, and SIAM Publications, Philadelphia (1996).

26. "E ective Parallel Computations with Toeplitz and Toeplitz-like Matrices Filled withIntegers", in Proc. of the AMS-SIAM Workshop "Mathematics of Numerical Analysis", Park City, Utah, July-August 1995 (J. Renegar, M. Shub, and S. Smale, editors), Lectures in Applied Math., 32, 593 641, Amer. Math. Society Press, Providence, Rhode Island (1996).

27. "Parallel Computation of Polynomial GCD and Some Related Parallel Computationsover Abstract Fields", Theoretical Computer Science, 162, 2, 173 223 (1996).

28. New Transformations of Cauchy Matrices and Trummer's Problem" (by V. Y. Pan, M. Abu Tabanjeh, Z. Q. Chen, E. Landowne, and A. Sadikou), Computers and Mathematics (with Applications), 35, 12, 1 5 (1998).

29. Transformations of Cauchy Matrices for Trummer's Problem and a Cauchy-like Linear Solver" (by V. Y. Pan, M. Abu Tabanjeh, Z. Chen, S. Providence and A. Sadikou), Proc. of 5th Annual International Symposium on Solving Irregularly Structured Problems in Parallel (IREGULAR'98), (A. Ferreira, J. Rolim, H. Simon, and S.-H. Teng, editors), Lecture Notes in Computer Science, 1457, 274 284, Springer, Berlin (1998).

30. A Uni ed Superfast Algorithm for Boundary Rational Tangential Interpolation Problem and for Inversion and Factorization of Dense Structured Matrices" (by V. Olshevsky and V. Y. Pan), Proc. 39th Annual IEEE Symposium on Foundation of Computer Science (FOCS'98), 192 201, IEEE Computer Society Press, Los Alamitos, California (1998).

31. Superfast Computations with Singular Structured Matrices over Abstract Fields" (by V. Y. Pan, A. Zheng, M. Abutabanjeh, Z. Chen, and S. Providence), Proc. Second Workshop on Computer Algebra in Scienti c Computing (CASC'99), (V. G. Ganzha, E. W. Mayr, E. V. Vorozhtsov, Editors), 323 338, Springer (May 1999).

32. Superfast Algorithms for Cauchy-like Matrix Computations and Extensions" (by V. Y. Pan and A. Zheng), Linear Algebra and Its Applications, 310, 83 108 (2000).

33. Matrix Structure, Polynomial Arithmetic, and Erasure-Resilient Encoding/Decoding", Proc. Intern. Symp. on Symbolic and Algebraic Computations (ISSAC'2000), 266 271, ACM Press, New York (2000).

34. Parallel Complexity of Computations with General and Toeplitz-like Matrices Filled with Integers", SIAM J. on Computing, 30, 1080 1125 (2000).

35. Can We Optimize Toeplitz/Hankel Computations?" in Proc. the 5th International Workshop on Computer Algebra in Scienti c Computing (CASC'02), (edited by E. W. Mayr, V. G. Ganzha, and E. V. Vorozhtzov), 253 264, Technische Univ. of M nchen, Germany (2002).

36. Inversion of Displacement Operators" (by V. Y. Pan and X. Wang), SIAM J. on Matrix Analysis and Applications, 24, 3, 660 677 (2003).

37. 'Matrix Structure and Loss-Resilient Encoding/Decoding", Computers and Mathematics (with Applications), 46, 493 499 (2003).

38. The Least-Squares Compression Policy for Newton-like Iteration for Structured Matrices" (by G. Codevico, V. Y. Pan, M. Van Barel, X. Wang, and A.-L. Zheng) in Proceedings of the 6th International Mathematica Symposium (IMS 2004), (edited by C. Jacob and P. Mitic), Ban , Alberta, Canada (August 2004).

39. An E cient Solution for Cauchy-like Systems of Linear Equations" (by Z. Chen and V. Y. Pan), Computers and Mathematics with Applications, 48, 529 537 (2004).

40. Fast and Stable QR Eigenvalue Algorithms for Generalized Semiseparable Matrices and Secular Equation" (by D. A. Bini, L. Gemignani and V. Y. Pan), Numerische Mathematik, 3, 373 408 (2005). Linking the TPR1, DPR1 and Arrow-head Matrix Structures" (by V. Y. Pan, B. Murphy, R. E. Rosholt, Y. Tang, X. Yan, and W. Cao), Computers and Mathematics with Applications, 52, 10-11, 1603 1608 (2006).

41. Eigen-solving via Reduction to DPR1 Matrices" (by V. Y. Pan, B. Murphy, R. Rosholt, Y. Tang, X. Wang, and A. Zheng), Computers and Mathematics with Applications, 56, 166 171 (2008).

42. Nearly Optimal Symbolic-Numerical Algorithms for Structured Integer Matrices and Polynomials" (by V. Y. Pan, B. Murphy, and R. E. Rosholt), Proc. Intern. Symp. Symbolic-Numerical Comp. (SNC'2009), Kyoto, Japan, August 2009, (edited by Hiroshi Kai and Hiroshi Sekigawa), 105 113, ACM Press, New York (2009).

43. Root Squaring with DPR1 Matrices", in Zapiski Nauchnykh Seminarov POMI (in English), volume 373 (edited by N. N. Vasiliev and A. M. Vershik), pp. 189 193 (2009).

44. Uni ed Nearly Optimal Algorithms for Structured Matrices" (by V. Y. Pan, B. Murphy, and R. E. Rosholt), Operator Theory: Advances and Applications, 199, 359 375, Birkha ser, Basel (2010).

45. Matrix Computations and Polynomial Root- nding with Preprocessing" (by V. Y. Pan, G. Qian, A.-L. Zheng, and Z. Chen), Linear Algebra and Its Applications, 434, 854 879 (2011).

46. Randomized Preconditioning of the MBA Algorithm" (by V. Y. Pan, G. Qian, and A.-L. Zheng), in Proc. International Symp. on Symbolic and Algebraic Computation (ISSAC'2011), San Jose, California, June 2011 (edited by Anton Leykin), 281 288, ACM Press, New York (2011).

47. Polynomial Evaluation and Interpolation and Transformations of Matrix Structures", Proceedings of the 15th International Workshop on Computer Algebra in Scienti c Computing (CASC'2013), (V. P. Gerdt, V. Koepf, E. W. Mayr, and E. V. Vorozhtsov, editors), Lecture Notes in Computer Science, 8136, 273 287, Springer, Heidelberg (2013).

48. Transformations of Matrix Structures Work Again", Linear Algebra and Its Applications, 465, 1 32 (2015).

49. Estimating the Norms of Circulant and Toeplitz Random Matrices and Their Inverses" (by V. Y. Pan, J. Svadlenka, and L. Zhao), Linear Algebra and Its Applications, 468, 197 210 (2015)

50. Randomized Circulant and Gaussian Preprocessing" (by V. Y. Pan and L. Zhao), Proceedings of the 17th International Workshop on Computer Algebra in Scienti c Computing (CASC'2015), (V. P. Gerdt, V. Koepf, and E. V. Vorozhtsov, editors), Lecture Notes in Computer Science, 9301, 359 373, Springer, Heidelberg (2015).

51. Nearly Optimal Computations with Structured Matrices" (by V. Y. Pan and E. P. Tsigaridas), Theoretical Computer Science, Special Issue on Symbolic Numerical Algorithms (Stephen Watt, Jan Verschelde, and Lihong Zhi, editors)), 681, 117 137, 2017. http://dx.doi.org/10.1016/j.tcs.2017.03.031.

52. How Bad Are Vandermonde Matrices?", SIAM Journal of Matrix Analysis, 37 (2016), no. 2, 676 694.

53. Fast approximate computations with Cauchy matrices and polynomials", Math. of Computation, 86, 2799 2826, 2017. DOI: https://doi.org/10.1090/mcom/3204.

54. Sublinear Cost Low Rank Approximation via Subspace Sampling" (by V. Y. Pan, Q. Luan, J. Svadlenka, and L. Zhao), In LNCS 11989, Book: Mathematical Aspects of Computer and Information Sciences (MACIS 2019), D. Salmanig et al (Eds.), Springer Nature Switzerland AG 2020, Chapter No: 9, pages 1 16, Springer Nature Switzerland AG 2020 Chapter DOI:10.1007/978-3-030-43120-4_9

55. CUR LRA at Sublinear Cost Based on Volume Maximization" (by Q. Luan, V. Y. Pan), LNCS 11989, In Book: Mathematical Aspects of Computer and Information Sciences (MACIS 2019), D. Salmanig et al (Eds.), Springer Nature Switzerland AG 2020, Chapter No: 10, pages 1 17, Chapter DOI:10.1007/978-3-030-43120-4_10

8.11 PARALLEL AND VLSI COMPUTATIONS

CHAPTERS IN BOOKS AND SURVEY ARTICLES, including prefaces

1. CHAPTER 4 of Polynomial and Matrix Computations", Volume 1: Fundamental Algorithms" (XVI + 415 pages) (by D. Bini and V. Y. Pan), in the series Progress in Theoretical Computer Science (R.V. Book editor), Birkh user, Boston (1994).

2. "Parallel Solution of Sparse Linear and Path Systems", in Synthesis of Parallel Algorithms (J. H. Reif, editor), Chapter 14, pp. 621 678. Morgan Kaufmann publishers, San Mateo, CA (1993).

RESEARCH PAPERS (in journals and refereed proceedings of conferences).

1. "E cient Parallel Solution of Linear Systems" (by V. Y. Pan and J. Reif), Proc. 17th Ann. ACM Symp. on Theory of Computing (STOC’85), 143 152, ACM Press, New York (1985).

2. "Improved Processor Bounds for Algebraic and Combinatorial Problems in RNC" (byZ. Galil and V. Y. Pan), Proc. 26th Ann. IEEE Symp. on Foundations of Computer Science (FOCS’85), 490 495, IEEE Computer Society Press, Los Angeles, California (1985).

3. . "Fast and E cient Algorithms for Sequential and Parallel Evaluation of Polynomial Zeros and of Matrix Polynomials," Proc. 26th Ann. IEEE Symp. on Foundations of Computer Science (FOCS’85), 522 531, IEEE Computer Society Press, Los Angeles, California (1985).

4. "Fast and E cient Parallel Algorithms for the Exact Inversion of Integer Matrices", Proc. Fifth Conf. on Foundations of Software Technology and Theoretical Computer Science (FST and TCS’85), (edited by K. V. Nori), Lecture Notes in Computer Science, 206, 504 521, Springer, Berlin (1985).

5. "Fast Parallel Polynomial Division via Reduction to Polynomial Inversion Modulo aPower" (by D. Bini and V. Y. Pan), Information Processing Letters, 21, 79 81 (1985).

6. "Fast and E cient Parallel Linear Programming and Least Squares Computations" (by V. Y. Pan and J. Reif), VLSI Algorithms and Architectures, Lecture Notes in Computer Science, 227, 283 295, Springer, Berlin (1986).

7. "A Logarithmic Boolean Time Algorithm for Parallel Polynomial Division" (by D.Bini and V. Y. Pan), VLSI Algorithms and Architectures, Lecture Notes in Computer Science, 227, 246 251, Springer, Berlin (1986).

8. "Fast Parallel Algorithms for Polynomial Division over Arbitrary Field of Constants"(by D. Bini and V. Y. Pan), Computers and Mathematics (with Applications), 12A, 11, 1105 1118 (1986).

9. "Extension of the Parallel Nested Dissection Algorithm to the Path Algebra Problems"(by V. Y. Pan and J. Reif), Proc. Sixth Conference on Foundations of Software Technology and Theoretical Computer Science (FST and TCS’86), (New Delhi, India), Lecture Notes in Computer Science, 241, 470 487, Springer, Berlin (1986).

10. "Parallel Nested Dissection for Path Algebra Computations" (by V. Y. Pan and J.Reif), Operations Research Letters, 5, 4, 177 184 (1986).

11. "E cient Parallel Linear Programming" (by V. Y. Pan and J. Reif), Operations Research Letters, 5, 3, 127 135 (1986).

12. "A Logarithmic Boolean Time Algorithm for Parallel Polynomial Division" (by D. Biniand V. Y. Pan), Information Processing Letters, 24, 233 237 (1987). 60. "Complexity of Parallel Matrix Computations", Theoretical Computer Science, 54, 65 85 (1987).

13. "Complexity of Parallel Matrix Computations", Theoretical Computer Science, 54,65 85 (1987).

14. "Sequential and Parallel Complexity of Approximate Evaluation of Polynomial Zeros",Computers and Mathematics (with Applications), 14, 8, 591 622 (1987).

15. "Some Polynomial and Toeplitz Matrix Computations" (by V. Y. Pan and J. Reif),Proc. 28th Ann. IEEE Symp. on Foundations of Computer Science (FOCS'87), 173 184, IEEE Computer Society Press, Los Angeles, California (1987).

16. "Improved Processor Bounds for Combinatorial Problems in RNC" (by Z. Galil and V. Y. Pan), Combinatorica, 8, 2, 189 200 (1988).

17. "Fast and E cient Solution of Path Algebra Problems" (by V. Y. Pan and J. Reif), Journal of Computer and Systems Sciences, 38, 3, 494 510 (1989).

18. "Parallel Evaluation of the Determinant and of the Inverse of a Matrix" (by Z. Galiland V. Y. Pan), Information Processing Letters, 30, 41 45 (1989).

19. "Fast and E cient Parallel Evaluation of the Zeros of a Polynomial Having Only Real Zeros", Computers and Mathematics (with Applications), 17, 11, 1475 1480 (1989).

20. "Fast and E cient Parallel Solution of Dense Linear Systems" (by V. Y. Pan and J. Reif), Computers and Math. (with Applications), 17, 11, 1481 1491 (1989).

21. "Fast and E cient Parallel Inversion of Toeplitz and Block Toeplitz Matrices", Operator Theory: Advances and Applications, 40, 359 389, Birkh user, Basel, Switzerland (1989).

22. "Parallel Complexity of Tridiagonal Symmetric Eigenvalue Problem" (by D. Bini andV. Y. Pan), Proc. 2nd Ann. ACM-SIAM Symp. on Discrete Algorithms (SODA'91), 384 393, ACM Press, New York, and SIAM Publications, Philadelphia (1991).

23. "Improved Parallel Computations with Matrices and Polynomials" (by D. Bini, L.Gemignani and V. Y. Pan), Proc. 18th Intern. Colloquium on Automata, Languages and Programming (ICALP'91), Lecture Notes in Computer Science, 510, 520 531, Springer, Berlin (1991).

24. "Processor E cient Parallel Solution of Linear Systems over an Abstract Field" (by E. Kaltofen and V. Y. Pan), Proc. 3rd Ann. ACM Symp. on Parallel Algorithms and Architectures (SPAA'91), 180 191, ACM Press, New York (1991).

25. "The Parallel Computation of the Minimum Cost Paths in Graphs by Stream Contraction" (by V. Y. Pan and J. Reif), Information Processing Letters, 40, 79 83 (1991).

26. "On Parallel Complexity of Integer Linear Programming, GCD and the Iterated ModFunction" (by Yu Lin-Kriz and V. Y. Pan), Proc. 3rd Ann. ACM-SIAM Symposium on Discrete Algorithms (SODA'92), 124 137, ACM Press, New York and SIAM Publications, Philadelphia (1992).

27. "E cient Parallel Algorithms for Computing All Pair Shortest Paths in Directed Graphs" (by Y. Han, V. Y. Pan and J. Reif), Proc. 4th Ann. ACM Symp. on Parallel Algorithms and Architectures (SPAA'92), 353 362, ACM Press, New York (1992).

28. "Supere ective Slow-down of Parallel Computations" (by V. Y. Pan and F. P.Preparata), Proc. 4th Ann. ACM Symp. on Parallel Algorithms and Architectures (SPAA'92), 402 409, ACM Press, New York (1992).

29. "Parallel Solution of Toeplitz-like Linear Systems", J. of Complexity, 8, 1 21 (1992).

30. "Concurrent Iterative Algorithm for Toeplitz-like Linear Systems", IEEE Trans. onParallel and Distributed Systems, 4, 5, 592 600 (1993).

31. "Fast and E cient Parallel Solution of Sparse Linear Systems" (by V. Y. Pan and J. Reif), SIAM J. on Computing, 22, 6, 1227 1250 (1993).

32. Improved Parallel Polynomial Division" (by D. Bini and V. Y. Pan), SIAM J. on Computing, 22, 3, 617 627 (1993).

33. "Improved Parallel Computations with Toeplitz-like and Hankel-like Matrices" (by D.Bini and V. Y. Pan), Proc. of Annual ACM-SIGSAM International Symposium on Symbolic and Algebraic Computation (ISSAC'93), 193 200, ACM Press, New York (1993).

34. "The NC-Equivalence of Planar Integer Linear Programming and Euclidean GCD"(by D. Shallcross, V. Y. Pan, and Yu Lin-Kriz), Proc. 34th Ann. IEEE Symp. on Foundations of Computer Science (FOCS'93), 557 564, IEEE Computer Society Press, Los Alamitos, California (1993).

35. "Improved Parallel Solution of a Triangular Linear System", Computers and Math.(with Applications), 27, 11, 41 43 (1994).

36. Optimum Parallel Computations with Band Matrices" (by V. Y. Pan, I. Sobze and A. Atinkpahoun), Proc. 5th Ann. ACM-SIAM Symp. on Discrete Algorithms (SODA'94), 649 658, ACM Press, New York, and SIAM Publications, Philadelphia (1994).

37. "Parallel Solution of Toeplitz and Toeplitz-like Linear Systems over Fields of SmallPositive Characteristic" (by E. Kaltofen and V. Y. Pan), Proceedings of First Intern. Symposium on Parallel Symbolic Computation (PASCO'94), Linz, Austria (Sept. 1994), Lecture Notes Series in Computing, 5, 225 233, World Scienti c Publishing Company, Singapore (1994).

38. "Work-Preserving Speed-up of Parallel Matrix Computations" (by V. Y. Pan and F. P. Preparata), SIAM J. on Computing, 24, 4, 811 821 (1995).

39. "Parallel Computation of a Krylov Matrix for a Sparse and Structured Input", Mathematical and Computer Modeling, 21, 11, 97 99 (1995).

40. "On Parallel Computations with Band Matrices" (by V. Y. Pan, I. Sobze, and A.Atinkpahoun), Information and Computation, 120, 2, 237 250 (1995).

41. "Optimal (up to Polylog Factors) Sequential and Parallel Algorithms for Approximating Complex Polynomial Zeros", Proc. 27th Ann. ACM Symposium on Theory of Computing (STOC'95), 741 750, ACM Press, New York (1995).

42. "Optimal and Nearly Optimal Algorithms for Approximating Polynomial Zeros", Computers and Math. (with Applications), 31, 12, 97 138 (1996).

43. "A New Approach to Parallel Computation of Polynomial GCDs and to Related Parallel Computations over Fields and Integer Rings", Proc. 7th Ann. ACM-SIAM Symposium on Discrete Algorithms (SODA'96), 518 527, ACM Press, New York, and SIAM Publications, Philadelphia (1996).

44. "E ective Parallel Computations with Toeplitz and Toeplitz-like Matrices Filled withIntegers", in Proc. of the AMS-SIAM Workshop "Mathematics of Numerical Analysis", Park City, Utah, July-August 1995 (J. Renegar, M. Shub, and S. Smale, editors), Lectures in Applied Math., 32, 593 641, Amer. Math. Society Press, Providence, Rhode Island (1996).

45. "Parallel Computation of Polynomial GCD and Some Related Parallel Computationsover Abstract Fields", Theoretical Computer Science, 162, 2, 173 223 (1996).

46. "E cient Parallel Algorithms for Computing All Pair Shortest Paths in Directed Graphs" (by Y. Han, V. Y. Pan and J. Reif), Algorithmica, 17, 399 415 (1997).

47. Planar Integer Linear Programming Is NC-equivalent to Euclidean GCD" (by D. F. Shallcross, V. Y. Pan and Y. Lin-Kriz), SIAM Journal on Computing, 27, 4, 960 971 (1998).

48. Computing Matrix Eigenvalues and Polynomial Zeros Where the Output Is Real" (by D. Bini and V. Y. Pan), SIAM J. on Computing, 27, 4, 1099 1115 (1998).

49. Parallel Matrix Multiplication on a Linear Array with a Recon gurable Pipelined Bus System" (by K. Li and V. Y. Pan), Proc. 13th Intern. Parallel Processing Symp. and 10th Symp. on Parallel and Distributed Computing (IPPS/SPDP 1999), 31 35, IEEE Computer Soc. Press, Los Alamitos, California (April 1999).

50. Parallel Complexity of Computations with General and Toeplitz-like Matrices Filled with Integers", SIAM J. on Computing, 30, 1080 1125 (2000).

51. Parallel Matrix Multiplication on a Linear Array with a Recon gurable Pipelined Bus System" (by K. Li and V. Y. Pan), IEEE Transactions on Computing, 50, 5, 519 525 (2001).

8.12 GRAPH ALGORITHMS

CHAPTERS IN BOOKS AND SURVEY ARTICLES

1. "Parallel Solution of Sparse Linear and Path Systems", in Synthesis of Parallel Algorithms (J. H. Reif, editor), Chapter 14, pp. 621 678. Morgan Kaufmann publishers, San Mateo, CA (1993).

RESEARCH PAPERS (in journals and refereed proceedings of conferences).

1."The Bit Operation Complexity of Matrix Multiplication and All Pair Shortest PathProblem", Computers and Math. (with Applications), 7, 5, 431 438 (1981).

2."Extension of the Parallel Nested Dissection Algorithm to the Path Algebra Problems"(by V. Y. Pan and J. Reif), Proc. Sixth Conference on Foundations of Software Technology and Theoretical Computer Science (FST and TCS'86), (New Delhi, India), Lecture Notes in Computer Science, 241, 470 487, Springer, Berlin (1986).

3."Parallel Nested Dissection for Path Algebra Computations" (by V. Y. Pan and J.Reif), Operations Research Letters, 5, 4, 177 184 (1986).

4."Improved Processor Bounds for Combinatorial Problems in RNC" (by Z. Galil and V. Y. Pan), Combinatorica, 8, 2, 189 200 (1988).

5."Fast and E cient Solution of Path Algebra Problems" (by V. Y. Pan and J. Reif), Journal of Computer and Systems Sciences, 38, 3, 494 510 (1989). item"The Parallel Computation of the Minimum Cost Paths in Graphs by Stream Contraction" (by V. Y. Pan and J. Reif), Information Processing Letters, 40, 79 83 (1991).

6."E cient Parallel Algorithms for Computing All Pair Shortest Paths in Directed Graphs" (by Y. Han, V. Y. Pan and J. Reif), Proc. 4th Ann. ACM Symp. on Parallel Algorithms and Architectures (SPAA'92), 353 362, ACM Press, New York (1992).

7."E cient Parallel Algorithms for Computing All Pair Shortest Paths in Directed Graphs" (by Y. Han, V. Y. Pan and J. Reif), Algorithmica, 17, 399 415 (1997).

8.13 COMPUTATIONS WITH INTEGERS, OVER INTEGERS AND RATIONALS, AND IN FINITE FIELDS

RESEARCH PAPERS (in journals and refereed proceedings of conferences).

1. "The Bit Complexity of Arithmetic Algorithms", Journal of Algorithms, 2, 144 163 (1981).

2. "The Bit Operation Complexity of Approximate Evaluation of Matrix and PolynomialProducts Using Modular Arithmetic", Computers and Math. (with Applications), 8, 2, 137 140 (1982).

3. "Fast and E cient Parallel Algorithms for the Exact Inversion of Integer Matrices", Proc. Fifth Conf. on Foundations of Software Technology and Theoretical Computer Science (FST and TCS'85), (edited by K. V. Nori), Lecture Notes in Computer Science, 206, 504 521, Springer, Berlin (1985).

4. "A Logarithmic Boolean Time Algorithm for Parallel Polynomial Division" (by D.Bini and V. Y. Pan), VLSI Algorithms and Architectures, Lecture Notes in Computer Science, 227, 246 251, Springer, Berlin (1986).

5. "Fast Parallel Algorithms for Polynomial Division over Arbitrary Field of Constants"(by D. Bini and V. Y. Pan), Computers and Mathematics (with Applications), 12A, 11, 1105 1118 (1986).

6. "A Logarithmic Boolean Time Algorithm for Parallel Polynomial Division" (by D.Bini and V. Y. Pan), Information Processing Letters, 24, 233 237 (1987).

7. "Extended Concept of Signi cant Digits and Lower Precision Computations", AppliedMathematics Letters, 5, 2, 3 6 (1992).

8. "Binary Segmentation for Matrix and Vector Operations", Computers and Math.(with Applications), 25, 3, 69 71 (1993).

9. "Faster Solution of the Key Equation for Decoding the BCH Error-Correcting Codes",Proc. 29th ACM Symposium on Theory of Computing (STOC'97), 168 175, ACM Press, New York (1997).

10. "Computing Exact Geometric Predicates Using Modular Arithmetic with Single Precision" (by H. Br nnimann, I. Z. Emiris, V. Y. Pan and S. Pion), Proc. 13th Ann. ACM Symp. on Computational Geometry, 174 182, ACM Press, New York (1997).

11. Modular Arithmetic for Linear Algebra Computations in the Real Field" (by I. Z. Emiris, V. Y. Pan and Y. Yu), J. of Symbolic Computation, 21, 1 17 (1998).

12. "Single Precision Computation of the Sign of Algebraic Predicates", Applied Math.Letters, 11, 6, 49 50 (1998).

13. Sign Determination in Residue Number Systems" (by H. Br nnimann, I. Z. Emiris, V. Y. Pan and S. Pion), Theoretical Computer Science, 210, 1, 173 197 (1999).

14. Certi ed Computation of the Sign of a Matrix Determinant" (by V. Y. Pan and Y. Yu), Proc. 10th Annual ACM-SIAM Symposium on Discrete Algorithms (SODA'99), 715 724, ACM Press, New York, and SIAM Publications, Philadelphia (1999).

15. New Techniques for the Computation of Linear Recurrence Coe cients", Finite Fields and Their Applications, 6, 93 118 (2000).

16. On Theoretical and Practical Acceleration of Randomized Computation of the Determinant of an Integer Matrix", Zapiski Nauchnykh Seminarov POMI (in English), Vol. 316, 163 187, St. Petersburg, Russia (2004).

17. Degeneration of Integer Matrices Modulo an Integer" (by V. Y. Pan and X. Wang), Linear Algebra and Its Applications, 429, 2113 2130 (2008).

18. A New Error free Floating-Point Summation Algorithm" (by V. Y. Pan, B. Murphy, G. Qian, and R. E. Rosholt), Computers and Mathematics with Applications, 57, 560 564 (2009).

19. Nearly Optimal Symbolic-Numerical Algorithms for Structured Integer Matrices and Polynomials" (by V. Y. Pan, B. Murphy, and R. E. Rosholt), Proc. Intern. Symp. Symbolic-Numerical Comp. (SNC'2009), Kyoto, Japan, August 2009, (edited by Hiroshi Kai and Hiroshi Sekigawa), 105 113, ACM Press, New York (2009).

20. Uni ed Nearly Optimal Algorithms for Structured Matrices" (by V. Y. Pan, B. Murphy, and R. E. Rosholt), Operator Theory: Advances and Applications, 199, 359 375, Birkha ser, Basel (2010).

21. "Nearly Optimal Solution of Rational Linear Systems of Equations with SymbolicLifting and Numerical Initialization", Computers and Mathematics with Applications, 62, 1685 1706 (2011).

8.14 LOWER BOUNDS

REVIEW ARTICLE

"On Methods of Computing the Values of Polynomials", Uspekhi Matematicheskikh Nauk (in Russian), 21, 1 (127), 103 134 (1966). (Transl. Russian Mathematical Surveys, 21, 1 (127), 105 137 (1966).)

RESEARCH PAPERS (in journals and refereed proceedings of conferences).

1."On Some Methods of Computing Polynomial Values", Problemy Kibernetiki (in Russian), (edited by A.A. Lyapunov), 7, 21 30 (1962). (Transl. Problems of Cybernetics, USSR, 7, 20 30, U.S. Dept. of Commerce (1962).)

2."The Lower Bounds on the Additive Complexity of Bilinear Problems in Terms ofSome Algebraic Quantities", Information Processing Letters, 13, 2, 71 72 (1981).
3."The Additive and Logical Complexities of Linear and Bilinear Algorithms", J. ofAlgorithms, 4, 1 34 (1983).

4."The Trade-o Between the Additive Complexity and Asynchronicity of Linear and Bilinear Algorithms", Information Processing Letters, 22, 11 14 (1986).

8.15 THEORY OF REAL AND COMPLEX FUNCTIONS

RESEARCH PAPERS (in journals).

1. "On One Question by N.N. Luzin", Nauchnye Doklady Vysshey Schkoly,Phiziko-Matematicheskie Nauki (in Russian), 4, 59 62 (1958).
2. "On Approximation of Analytic Functions by Rational Ones", Uspekhi Matematicheskikh Nauk (in Russian), 16, 5 (101), 195 197 (1961).

8.16 ECONOMICS AND MATHEMATICAL ECONOMICS

RESEARCH PAPERS (in journals and refereed proceedings of conferences).

1."Calculus of Rational Costs Based on Modern Economic Information" (by V. Belkin,A. Kronrod, Y. Nazarov and V. Y. Pan), Ekonomika i Matematicheskie Metody (in Russian), Akademiya Nauk SSSR, 1, 5, 699 717 (1965).

2."A Linear Model and Algorithm for Optimizing Foreign Trade", Tezisy Dokladovi Vystupleniy na Simpoziume po Modelirovaniyu Narodnogo Khozyaistva, Institut Ekonomiki AN SSSR (Proceedings of the Symposium on Models of Public Economy, Institute of Economics, Academy of Sciences of USSR), (in Russian), Moscow, 29 37 (1970).

3."On Solving a Distribution Problem with Upper Bounds on the Variables and a Simplied Criterion for Optimizing Foreign Trade", Trudy 4-oy Zimney Schkoly po Matematicheskomu Programmirovaniyu i Smezhnym Voprosam (in Russian), (Transactions of the 4-th Winter School on Mathematical Programming and Adjacent Problems), (edited by S. I. Zukhovitskiy), Iss. 5, 26 49, Drogobych (1972).

4."Models for Planning Costs with Optimization of Foreign Trade of Several Countriesunder the Economic Integration", Primenenie Ekonomiko-Matematicheskikh Modeley i EVM pri Planirovanii i Prognozirovanii Tsen (in Russian), 5 (1973).

5."A Model for the Optimization of Foreign Economic Relations Under the EconomicIntegration of the Socialist Countries", Ekonomika i Matematicheskie Metody (in Russian), Akademiya Nauk SSSR, 10, 2, 255 266 (1974). (Transl. USSR Trade and Services, 755, 1 16 (1974).)

6."A Model for the Dynamics of Costs and Expenditures", Trudy MezhdunarodnoyKonferentsii "Modelirovanie Ekonomicheskikh Protsessov" (Transactions of the International Conference, "Models of Economic Processes"), (in Russian), Erevan, 166 174 (1974).

7."A Modi cation of a Balanced Model of Income-Commodities into an EquilibriumModel" (by V. Belkin, V. Ivanter, N. Konstantinov and V. Pan), Ekonomika i Matematicheskie Methody, Akademiya Nauk SSSR (in Russian), 11, 6, 1037 1049 (1975).

9 SELECTED PUBLICATIONS

Also see PUBLICATIONS and RESEARCH in sections 8, 10 and 12

9.1 BOOKS

1. How to Multiply Matrices Faster", Lecture Notes in Computer Science, vol. 179 (XI + 212 pages), Springer, Berlin (1984).
2. Polynomial and Matrix Computations", Volume 1: Fundamental Algorithms" (XVI + 415 pages) (by D. Bini and V. Y. Pan), in the series Progress in Theoretical Computer Science (R.V. Book editor), Birkh user, Boston (1994).
3. Structured Matrices and Polynomials: Uni ed Superfast Algorithms" (XXV + 278 pages), Birkh user/Springer, Boston/New York (June 2001).
4. Numerical Methods for Roots of Polynomials" (by J. M. McNamee and V. Y. Pan), Part 2 (XXII + 718 pages), Elsevier (2013).

9.2 CHAPTERS IN BOOKS AND SURVEY ARTICLES (12 ITEMS SELECTED OUT OF 24). Items 1, 2, 3-5, 8, and 9 included new research results.

1. "On Methods of Computing the Values of Polynomials", Uspekhi Matematicheskikh Nauk, 21, 1 (127), 103 134 (1966). (Transl. Russian Mathematical Surveys, 21, 1 (127), 105 137 (1966).)
2. "How Can We Speed Up Matrix Multiplication?", SIAM Review, 26, 3, 393 415 (1984).
3. "Complexity of Computations with Matrices and Polynomials," SIAM Review, 34, 2,225 262 (1992).
4. "Parallel Solution of Sparse Linear and Path Systems", in Synthesis of Parallel Algorithms (J. H. Reif, editor), Chapter 14, pp. 621 678. Morgan Kaufmann publishers, San Mateo, CA (1993).
5. "Solving a Polynomial Equation: Some History and Recent Progress", SIAM Review,39, 2, 187 220 (1997).
6. "Solving Polynomials with Computers", American Scientist, 86, 62 69 (JanuaryFebruary 1998).
7. "Some Recent Algebraic/Numerical Algorithms", Electronic Proceedings of IMACS/ACA'98 (1998): http:www-troja.fj .cvut.cz/aca98/sessions/approximate 8. Newton's Iteration for Structured Matrices and Linear Systems of Equations" (by V. Y. Pan, S. Branham, R. Rosholt, and A. Zheng), SIAM volume on Fast Reliable Algorithms for Matrices with Structure (T. Kailath and A. H. Sayed, editors), chapter 7, pages 189 210, SIAM Publications, Philadelphia (1999).
9. Root nding with Eigen-solving" (by V. Y. Pan, D. Ivolgin, B. Murphy, R. E. Rosholt, Y. Tang, X. Wang, and X. Yan), pages 185 210 in Symbolic Numeric Computation (Dongming Wang and Lihong Zhi, editors), Birkh user, Basel/Boston (2007).

10. "Fast Fourier Transform and Its Applications" (by I. Z. Emiris and V. Y. Pan, in Algorithms and Theory of Computations Handbook", Second Edition, Volume 1 (1016 pages): General Concepts and Techniques, pages 1 31 in Chapter 18 (Mikhail J. Atallah and Marina Blanton, editors), CRC Press Inc., Boca Raton, Florida (2009).

11. Algebraic Algorithms" (by I. Z. Emiris, V. Y. Pan, and E. Tsigaridas), Chapter 10 (pages from 10 1 to 10-40) of Computing Handbook (Third edition), Volume I: Computer Science and Software Engineering (Allen B. Tucker, Teo Gonzales, and Jorge L. Diaz-Herrera, editors), Taylor and Francis Group (2014).

12. "Fast Matrix Multiplication and Its Algebraic Neighborhood", DOI: 10.4213/sm8833 Matem. Sbornik, 208, 11, 90 138 (2017) and DOI:10.1070/SM8833: Sb. Math., 208, 11, 1661 1704 (2017).

9.3 RESEARCH PAPERS (in journals and refereed proceedings of conferences). 37 SELECTED FROM 277

1. "On Schemes for the Evaluation of Products and Inverses of Matrices", Uspekhi Matematicheskikh Nauk (in Russian), 27, 5 (167), 249 250 (1972). Also see translation into English: "Better Late Than Never: Filling a Void in the History of Fast Matrix Multiplication and Tensor Decompositions", arXiv:1411.1972 (6 Nov 2014)

2. "Strassen's Algorithm Is Not Optimal. Trilinear Technique of Aggregating, Unitingand Canceling for Constructing Fast Algorithms for Matrix Multiplication", Proceedings of the 19th Annual IEEE Symposium on Foundations of Computer Science (FOCS'78), 166 176, IEEE Computer Society Press, Long Beach, California (1978). Journal version: "New Fast Algorithms for Matrix Operations", SlAM J. on Computing, 9, 2, 321 342 (1980).

3. "Methods of Aggregations" (by W. L. Miranker and V. Y. Pan), Linear Algebra andIts Applications, 29, 231 257 (1980).

4. "New Combinations of Methods for the Acceleration of Matrix Multiplications", Computers and Mathematics (with Applications), 7, 73 125 (1981).

5. "Trilinear Aggregating with Implicit Canceling for a New Acceleration of Matrix Multiplication", Computers and Mathematics (with Applications), 8, 1, 23 34 (1982).

6. "Complexity of Parallel Matrix Computations", Theoretical Computer Science, 54,65 85 (1987).

7. "On Computations with Dense Structured Matrices", Mathematics of Computation,55, 191, 179 190 (1990). Proceedings version in ISSAC 1989.

8. "An Improved Newton Iteration for the Generalized Inverse of a Matrix, with Applications" (by V. Y. Pan and R. Schreiber), SIAM J. on Scienti c and Statistical Computing, 12, 5, 1109 1131 (1991).

9. "On Practical Algorithms for Accelerated Matrix Multiplication" (by J. Laderman, V. Y. Pan and H-X. Sha), Linear Algebra and Its Applications, 162 164, 557 588 (1992).

10. "Parametrization of Newton's Iteration for Computations with Structured Matricesand Applications", Computers and Mathematics (with Applications), 24, 3, 61 75 (1992).

11. "Fast and E cient Parallel Solution of Sparse Linear Systems" (by V. Y. Pan and J. Reif), SIAM J. on Computing, 22, 6, 1227 1250 (1993). Proceedings version in STOC 1985.

12. "Optimal and Nearly Optimal Algorithms for Approximating Polynomial Zeros",Computers and Mathematics (with Applications), 31, 12, 97 138 (1996). Proceedings version: "Optimal (up to Polylog Factors) Sequential and Parallel Algorithms for Approximating Complex Polynomial Zeros", Proc. 27th Annual ACM Symposium on Theory of Computing (STOC'95), 741 750, ACM Press, New York (1995).

13. "Planar Integer Linear Programming Is NC-equivalent to Euclidean GCD" (by D.F. Shallcross, V. Y. Pan and Y. Lin-Kriz), SIAM Journal on Computing, 27, 4, 960 971 (1998). Proceedings version in IEEE FOCS 1993.

14. Fast Rectangular Matrix Multiplication and Applications" (by X. Huang and V. Y. Pan), J. of Complexity, 14, 257-299 (1998). Proceedings version in PASCO 1997.

15. Approximating Complex Polynomial Zeros: Modi ed Quadtree (Weyl's) Construction and Improved Newton's Iteration", J. of Complexity, 16, 1, 213 264 (2000).

16. Multivariate Polynomials, Duality and Structured Matrices" (by B. Mourrain and V. Y. Pan), J. of Complexity, 16, 1, 110 180 (2000).

17. Univariate Polynomials: Nearly Optimal Algorithms for Numerical Factorization and Root-Finding", J. of Symbolic Computation, 33, 5, 701 733 (2002). Proceedings version in ISSAC 2001.

18. Inversion of Displacement Operators" (by V. Y. Pan and X. Wang), SIAM J. on Matrix Analysis and Applications, 24, 3, 660 677 (2003).

19. Inverse Power and Durand-Kerner Iteration for Univariate Polynomial RootFinding" (by D. A. Bini, L. Gemignani and V. Y. Pan), Computers and Mathematics (with Applications), 47, 2/3, 447 459 (2004).

20. Fast and Stable QR Eigenvalue Algorithms for Generalized Semiseparable Matrices and Secular Equation" (by D. A. Bini, L. Gemignani and V. Y. Pan), Numerische Mathematik, 3, 373 408 (2005).

21. Improved Algorithms for Computing Determinants and Resultants" (by I. Z. Emirisand V. Y. Pan), J. of Complexity, 21, 1, 43 71 (2005).

22. Homotopic Residual Correction Algorithms for General and Structures Matrices" (by V. Y. Pan, M. Kunin, R. Rosholt, and H. Kodal), Mathematics of Computation, 75, 345 368 (2006).

23. Degeneration of Integer Matrices Modulo an Integer" (by V. Y. Pan and X. Wang), Linear Algebra and Its Applications, 429, 2113 2130 (2008).

24. Solving Homogeneous Linear Systems with Randomized Preprocessing" (by V. Y. Pan and G. Qian), Linear Algebra and Its Applications, 432, 3272 3318 (2010).

25. New Progress in Real and Complex Polynomial Root Finding" (by V. Y. Pan and A. L. Zheng), Computers and Mathematics (with Applications), 61, 1305 1334 (2011).

26. Solving Linear Systems of Equations with Randomization, Augmentation and Aggregation" (by V. Y. Pan and G. Qian), Linear Algebra and Its Applications, 437, 2851 2876 (2012).

27. Randomized Preconditioning versus Pivoting" (by V. Y. Pan, G. Qian, and A.-L. Zheng), Linear Algebra and Its Applications, 438, 4, 1883 1889 (2013).

28. Transformations of Matrix Structures Work Again", Linear Algebra and Its Applications, 465, 1-32 (2015).

29. Random Multipliers Numerically Stabilize Gaussian and Block Gaussian Elimination: Proofs and an Extension to Low-rank Approximation", Linear Algebra and Its Applications, 481, 202 234 (2015).

30. Nearly Optimal Re nement of Real Roots of a Univariate Polynomial" by Victor Y. Pan and Elias Tsigaridas, J. of Symbolic Computations, 74, 181 204 (2016). Proceedings version in ISSAC 2013.

31. How Bad Are Vandermonde Matrices?", SIAM Journal of Matrix Analysis, 37, 2, 676 694 (2016).

32. Numerically Safe Gaussian Elimination with No Pivoting" (by Victor Y. Pan and Liang Zhao), Linear Algebra and Its Applications, 527, 349 383 (2017).

http://dx.doi.org/10.1016/j.laa.017.04.007 Real Polynomial Root nding by Means of Matrix and Polynomial Iterations" (by Victor Y. Pan and Liang Zhao), Theoretical Computer Science, Special Issue on Symbolic Numerical Algorithms (Stephen Watt, Jan Verschelde, and Lihong Zhi, editors), 681, 101-116 (2017), http://dx.doi.org/10.1016/j.tcs.2017.03.032.

33. Nearly Optimal Computations with Structured Matrices" (by V. Y. Pan and E. P. Tsigaridas), Theoretical Computer Science, Special Issue on Symbolic Numerical Algorithms (Stephen Watt, Jan Verschelde, and Lihong Zhi, editors)), 681, 117 137, 2017. http://dx.doi.org/10.1016/j.t cs.2017.03.031.
34. Old and New Nearly Optimal Polynomial Root-Finders", In: Proceedings of the 21stInternational Workshop on Computer Algebra in Scienti c Computing (CASC'2019), (M. England, W. Koepf, T.M. Sadikov, W.M. Seiler, and E. V. Vorozhtsov, editors), Lecture Notes in Computer Science, 11661, 393 411, Springer, Nature Switzerland (2019), doi: 10.1007/978-3-030-26831-2 and arXiv:1805.12042 Submitted on 30 May 2018
35. CUR LRA at Sublinear Cost Based on Volume Maximization" (by Q. Luan, V. Y. Pan), LNCS 11989, In Book: Mathematical Aspects of Computer and Information Sciences (MACIS 2019), D. Salmanig et al (Eds.), Springer Nature Switzerland AG 2020, Chapter No: 10, pages 1 17, Chapter DOI:10.1007/978-3-030-43120-4_10
36. Acceleration of Subdivision Root-Finding for Sparse Polynomials", In: Computer Algebra in Scienti c Computing, Springer Nature Switzerland AG 2020, F. Boulier et al.
(Eds.): CASC 2020, LNCS 12291, Ch. 27, pp. 1 17, 2020 https://doi.org/10.1007/978 − 3 − 030 − 60026 − 6_27
37. Faster Numerical Univariate Polynomial Root-Finding by Means of Subdivision Iterations" (by Qi Luan, Victor Y. Pan, Wongeun Kim, and Vitaly Zaderman), In: Computer Algebra in Scienti c Computing, Springer Nature Switzerland AG 2020, F. Boulier et al.
(Eds.): CASC 2020, LNCS 12291, Ch. 25, pp. 1 16, 2020 https://doi.org/10.1007/978 − 3 − 030 − 60026 − 6_25

9.4 RESEARCH PAPERS (in journals and refereed proceedings of conferences). 80 SELECTED FROM 277

1. "On Schemes for the Evaluation of Products and Inverses of Matrices", UspekhiMatematicheskikh Nauk (in Russian), 27, 5 (167), 249 250 (1972). Also see translation into English: "Better Late Than Never: Filling a Void in the History of Fast Matrix Multiplication and Tensor Decompositions", arXiv:1411.1972 (6 Nov 2014)

2. "Strassen's Algorithm Is Not Optimal. Trilinear Technique of Aggregating, Uniting and Canceling for Constructing Fast Algorithms for Matrix Multiplication", Proceedings of the 19th Annual IEEE Symposium on Foundations of Computer Science (FOCS'78), 166 176, IEEE Computer Society Press, Long Beach, California (1978).

3. "Fields Extension and Trilinear Aggregating, Uniting and Canceling for the Acceleration of Matrix Multiplication", Proceedings of the 20th Annual IEEE Symposium on Foundations of Computer Science (FOCS'79), 28 38, IEEE Computer Society Press, Long Beach, California (1979).

4. "Methods of Aggregations" (by W. L. Miranker and V. Y. Pan), Linear Algebra andIts Applications, 29, 231 257 (1980).

5. "New Fast Algorithms for Matrix Operations", SlAM J. on Computing, 9, 2, 321 342 (1980).

6. "New Combinations of Methods for the Acceleration of Matrix Multiplications", Computers and Mathematics (with Applications), 7, 73 125 (1981).

7. "Trilinear Aggregating with Implicit Canceling for a New Acceleration of Matrix Multiplication", Computers and Mathematics (with Applications), 8, 1, 23 34 (1982).

8. "Polynomial Division and Its Computational Complexity" (by D. Bini and V. Y. Pan),Journal of Complexity, 2, 179 203 (1986).

9. "Complexity of Parallel Matrix Computations", Theoretical Computer Science, 54,65 85 (1987).

10. "Improved Processor Bounds for Combinatorial Problems in RNC" (by Z. Galil and V. Y. Pan), Combinatorica, 8, 2, 189 200 (1988).

11. "Fast and E cient Solution of Path Algebra Problems" (by V. Y. Pan and J. Reif), Journal of Computer and Systems Sciences, 38, 3, 494 510 (1989).

12. "On Computations with Dense Structured Matrices", Mathematics of Computation,55, 191, 179 190 (1990). Proceedings version in ISSAC 1989.

13. "Processor E cient Parallel Solution of Linear Systems over an Abstract Field" (by E. Kaltofen and V. Y. Pan), Proc. 3rd Annual ACM Symposium on Parallel Algorithms and Architectures (SPAA'91), 180 191, ACM Press, New York (1991).

14. "An Improved Newton Iteration for the Generalized Inverse of a Matrix, with Applications" (by V. Y. Pan and R. Schreiber), SIAM J. on Scienti c and Statistical Computing, 12, 5, 1109 1131 (1991).

15. "Parametrization of Newton's Iteration for Computations with Structured Matricesand Applications", Computers and Mathematics (with Applications), 24, 3, 61 75 (1992).

16. "Compact Multigrid" (by V. Y. Pan and J. Reif), SIAM J. on Scienti c and StatisticalComputing, 13, 1, 119 127 (1992).

17. "Parallel Solution of Toeplitz-like Linear Systems", J. of Complexity, 8, 1 21 (1992).

18. "A Fast, Preconditioned Conjugate Gradient Toeplitz Solver" (by V. Y. Pan and R.Schreiber), Computers and Mathematics (with Applications), 24, 7, 17 24 (1992).

19. "On Practical Algorithms for Accelerated Matrix Multiplication" (by J. Laderman, V.Y. Pan and H.-X. Sha), Linear Algebra and Its Applications, 162 164, 557 588 (1992).

20. "Practical Improvement of the Divide-and-Conquer Eigenvalue Algorithms" (by D.Bini and V. Y. Pan), Computing, 48, 109 123 (1992). Proceedings version in SODA 1991.

21. "Processor-E cient Parallel Solution of Linear Systems II. The Positive Characteristic and Singular Cases" (by E. Kaltofen and V. Y. Pan), Proc. of 33rd Annual IEEE Symposium on Foundations of Computer Science (FOCS'92), 714 723, IEEE Computer Society Press, Los Alamitos, California (1992).

22. "Concurrent Iterative Algorithm for Toeplitz-like Linear Systems", IEEE Trans. onParallel and Distributed Systems, 4, 5, 592 600 (1993).

23. "Fast and E cient Parallel Solution of Sparse Linear Systems" (by V. Y. Pan and J. Reif), SIAM J. on Computing, 22, 6, 1227 1250 (1993). Proceedings version in STOC 1985.

24. "Decreasing the Displacement Rank of a Matrix", SIAM J. on Matrix Analysis andApplications, 14, 1, 118 121 (1993).

25. "A New Algorithm for the Symmetric Tridiagonal Eigenvalue Problem" (by V. Y. Panand J. Demmel), Journal of Complexity, 9, 387 405 (1993).

26. Improved Parallel Polynomial Division" (by D. Bini and V. Y. Pan), SIAM J. on Computing, 22, 3, 617 627 (1993). Proceedings version in FOCS 1992.

27. "Fast and E cient Parallel Solution of Sparse Linear Systems" (by V. Y. Pan and J. Reif), SIAM J. on Computing, 22, 6, 1227 1250 (1993). Proceedings version in STOC 1985.

28. "New Resultant Inequalities and Complex Polynomial Factorization", SIAM Journalon Computing 23, 5, 934 950 (1994).

29. "An Algebraic Approach to Approximate Evaluation of a Polynomial on a Set of RealPoints", Advances in Computational Mathematics, 3, 41 58 (1995).

30. "Work-Preserving Speed-up of Parallel Matrix Computations" (by V. Y. Pan and F.P. Preparata), SIAM J. on Computing, 24, 4, 811 821 (1995). Proceedings version in SPAA 1992.

31. "Optimal (up to Polylog Factors) Sequential and Parallel Algorithms for Approximating Complex Polynomial Zeros", Proc. 27th Annual ACM Symposium on Theory of Computing (STOC'95), 741 750, ACM Press, New York (1995).

32. "Optimal and Nearly Optimal Algorithms for Approximating Polynomial Zeros", Computers and Mathematics (with Applications), 31, 12, 97 138 (1996).

33. "Parallel Computation of Polynomial GCD and Some Related Parallel Computationsover Abstract Fields", Theoretical Computer Science, 162, 2, 173-223 (1996).

34. "E cient Parallel Algorithms for Computing All Pair Shortest Paths in Directed Graphs" (by Y. Han, V. Y. Pan and J. Reif), Algorithmica, 17, 399 415 (1997).

35. Planar Integer Linear Programming Is NC-equivalent to Euclidean GCD" (by D. F. Shallcross, V. Y. Pan, and Y. Lin-Kriz), SIAM Journal on Computing, 27, 4, 960 971 (1998). Proceedings version in FOCS 1993.

36. Computing Matrix Eigenvalues and Polynomial Zeros Where the Output Is Real" (by D. Bini and V. Y. Pan), SIAM J. on Computing, 27, 4, 1099 1115 (1998). Proceedings version in SODA 1991.

37. Fast Rectangular Matrix Multiplication and Applications" (by X. Huang and V. Y. Pan), J. of Complexity, 14, 257 299 (1998). Proceedings version in PASCO 1997.

38. "E cient Parallel Algorithms for Computing All Pair Shortest Paths in Directed Graphs" (by Y. Han, V.Y. Pan, and J. Reif), Algorithmica, 17, 399 415 (1997).

39. A Uni ed Superfast Algorithm for Boundary Rational Tangential Interpolation Problem and for Inversion and Factorization of Dense Structured Matrices" (by V. Olshevsky and V. Y. Pan), Proc. 39th Annual IEEE Symposium on Foundation of Computer Science (FOCS’98), 192 201, IEEE Computer Society Press, Los Alamitos, California (1998).

40. ‘Sign Determination in Residue Number Systems" (by H. Br nnimann, I. Z. Emiris, V. Y. Pan and S. Pion), Theoretical Computer Science, 210, 1, 173 197 (1999).

41. Newton’s Iteration for Structured Matrices and Linear Systems of Equations" (by V. Y. Pan, S. Branham, R. Rosholt, and A. Zheng), SIAM volume on Fast Reliable Algorithms for Matrices with Structure (T. Kailath and A. H. Sayed, editors), ch. 7, pp. 189 210, SIAM Publications, Philadelphia (1999).

42. Approximating Complex Polynomial Zeros: Modi ed Quadtree (Weyl’s) Construction and Improved Newton’s Iteration", J. of Complexity, 16, 1, 213 264 (2000).

43. Multivariate Polynomials, Duality and Structured Matrices" (by B. Mourrain and V. Y. Pan), J. of Complexity, 16, 1, 110 180 (2000).

44. New Techniques for the Computation of Linear Recurrence Coe cients", Finite Fields and Their Applications, 6, 93 118 (2000).

45. Computation of a Speci ed Root of a Polynomial System of Equations Using Eigenvectors" (by D. Bondyfalat, B. Mourrain, and V. Y. Pan), Linear Algebra and Its Applications, 319, 193 209 (2000).

46. Certi cation of Numerical Computation of the Sign of the Determinant of a Matrix" (by V. Y. Pan and Y. Yu), Algorithmica, 30, 708 724 (2001).

47. Numerical Computation of a Polynomial GCD and Extensions", Information and Computation, 167, 2, 71 85 (2001).

48. Structured Matrices and Newton’s Iteration: Uni ed Approach" (by V. Y. Pan, Y. Rami, and X. Wang), Linear Algebra and Its Applications, 343/344, 233 265 (2002).

49. Symbolic and Numerical Methods for Exploiting Structure in Constructing Resultant Matrices" (by I. Z. Emiris and V. Y. Pan), Journal of Symbolic Computation, 33, 393 413 (2002).

50. Univariate Polynomials: Nearly Optimal Algorithms for Numerical Factorization and Root-Finding", J. of Symbolic Computation, 33, 5, 701 733 (2002).

51. Accelerated Solution of Multivariate Polynomial Systems of Equations" (by B. Mourrain, V. Y. Pan, and O. Ruatta), SIAM J. on Computing, 32, 2, 435 454 (2003).

52. Inversion of Displacement Operators" (by V. Y. Pan and X. Wang), SIAM J. on Matrix Analysis and Applications, 24, 3, 660 677 (2003).

53. Inverse Power and Durand-Kerner Iteration for Univariate Polynomial Root-Finding" (by D. A. Bini, L. Gemignani, and V. Y. Pan), Computers and Mathematics (with Applications), 47, 2/3, 447 459 (2004).

54. Fast and Stable QR Eigenvalue Algorithms for Generalized Semiseparable Matrices and Secular Equation" (by D. A. Bini, L. Gemignani, and V. Y. Pan), Numerische Mathematik, 3, 373 408 (2005).

55. Homotopic Residual Correction Algorithms for General and Structures Matrices" (by V. Y. Pan, M. Kunin, R. Rosholt, and H. Kodal), Mathematics of Computation, 75, 345 368 (2006).

56. Degeneration of Integer Matrices Modulo an Integer" (by V. Y. Pan and X. Wang), Linear Algebra and Its Applications, 429, 2113 2130 (2008).

57. Solving Homogeneous Linear Systems with Randomized Preprocessing" (by V. Y. Pan and G. Qian), Linear Algebra and Its Applications, 432, 3272 3318 (2010).

58. Additive Preconditioning for Matrix Computations" (by V. Y. Pan, D. Ivolgin, B. Murphy, R. E. Rosholt, Y. Tang, and X. Yan), Linear Algebra and Its Applications, 432, 1070 1089 (2010).

59. Newton's Iteration for Matrix Inversion, Advances and Extensions", pp. 364 381, in Matrix Methods: Theory, Algorithms and Applications (dedicated to the Memory of Gene Golub, edited by Vadim Olshevsky and Eugene Tyrtyshnikov), World Scienti c Publishing, New Jersey, ISBN-13 978-981-283-601-4, ISBN-10-981-283-601-2 (2010).

60. Randomized Preconditioning of the MBA Algorithm" (by V. Y. Pan, G. Qian, and A. L. Zheng), in Proc. International Symposium on Symbolic and Algebraic Computation (ISSAC'2011), San Jose, California, June 2011 (edited by Anton Leykin), 281 288, ACM Press, New York (2011).

61. New Progress in Real and Complex Polynomial Root Finding" (by V. Y. Pan and A. L. Zheng), Computers and Mathematics (with Applications), 61, 1305 1334 (2011).

62. Solving Linear Systems of Equations with Randomization, Augmentation and Aggregation" (by V. Y. Pan and G. Qian), Linear Algebra and Its Applications", 437, 2851 2876 (2012).

63. Randomized Preconditioning versus Pivoting" (by V. Y. Pan, G. Qian, and A.-L. Zheng), Linear Algebra and Its Applications, 438, 4, 1883 1889 (2013).

64. Estimating the Norms of Circulant and Toeplitz Random Matrices and Their Inverses" (by V. Y. Pan, J. Svadlenka, and L. Zhao), Linear Algebra and Its Applications, 468, 197 210 (2015).

65. Transformations of Matrix Structures Work Again", Linear Algebra and Its Applications, 465, 1 32 (2015).

66. Random Multipliers Numerically Stabilize Gaussian and Block Gaussian Elimination: Proofs and an Extension to Low-rank Approximation", Linear Algebra and Its Applications, 481, 202 234 (2015).

67. Low-rank Approximation of a Matrix: Novel Insights, New Progress, and Extensions" by Victor Y. Pan and Liang Zhao, Proc. of the Eleventh International Computer Science Symposium in Russia (CSR'2016), (Alexander Kulikov and Gerhard Woeginger, editors), St. Petersburg, Russia, June 2016, Lecture Notes in Computer Science (LNCS), Volume 9691, 352 366, Springer International Publishing, Switzerland (2016).

68. Nearly Optimal Re nement of Real Roots of a Univariate Polynomial" by Victor Y. Pan and Elias Tsigaridas, J. of Symbolic Computations, 74, 181 204 (2016).

69. How Bad Are Vandermonde Matrices?", SIAM Journal of Matrix Analysis and Applications, 37, 2, 676 694 (2016).

70. Real Polynomial Root- nding by Means of Matrix and Polynomial Iterations" by V. Y. Pan and L. Zhao, Theoretical Computer Science, Special Issue on Symbolic Numerical Algorithms (Stephen Watt, Jan Verschelde, and Lihong Zhi, editors), 681, 101 116 (2017), http://dx.doi.org/10.1016/j.tcs.2017.03.032

71. "Numerically Safe Gaussian Elimination with No Pivoting" (by Victor Y. Pan andLiang Zhao), Linear Algebra and Its Applications, 527, 349 383 (2017). http://dx.doi.org/10.1016/j.laa.2017.04.007

72. Fast Approximate Computations with Cauchy Matrices and Polynomials", Mathematics of Computation, 86, 2799 2826 (2017). https://doi.org/10.1090/mcom/3204

73. Real Polynomial Root nding by Means of Matrix and Polynomial Iterations" (by Victor Y. Pan and Liang Zhao), Theoretical Computer Science, Special Issue on Symbolic Numerical Algorithms (Stephen Watt, Jan Verschelde, and Lihong Zhi, editors), 681, 101-116 (2017), http://dx.doi.org/10.1016/j.tcs.2017.03.032.

74. Nearly Optimal Computations with Structured Matrices" (by V. Y. Pan and E. P. Tsigaridas), Theoretical Computer Science, Special Issue on Symbolic Numerical Algorithms (Stephen Watt, Jan Verschelde, and Lihong Zhi, editors)), 681, 117 137, 2017. http://dx.doi.org/10.1016/j.tcs.2017.03.031.

75. Old and New Nearly Optimal Polynomial Root-Finders", In: Proceedings of the 21st International Workshop on Computer Algebra in Scienti c Computing (CASC’2019), (M. England, W. Koepf, T.M. Sadikov, W.M. Seiler, and E. V. Vorozhtsov, editors), Lecture Notes in Computer Science, 11661, 393 411, Springer, Nature Switzerland (2019), doi: 10.1007/978-3-030-26831-2 and arXiv:1805.12042 Submitted on 30 May 2018

76. Sublinear Cost Low Rank Approximation via Subspace Sampling" (by V. Y. Pan, Q. Luan, J. Svadlenka, and L. Zhao), In LNCS 11989, Book: Mathematical Aspects of Computer and Information Sciences (MACIS 2019), D. Salmanig et al (Eds.), Springer Nature Switzerland AG 2020, Chapter No: 9, pages 1 16, Springer Nature Switzerland AG 2020 Chapter DOI:10.1007/978-3-030-43120-4_9

77. CUR LRA at Sublinear Cost Based on Volume Maximization" (by Q. Luan, V. Y. Pan), LNCS 11989, In Book: Mathematical Aspects of Computer and Information Sciences (MACIS 2019), D. Salmanig et al (Eds.), Springer Nature Switzerland AG 2020, Chapter No: 10, pages 1 17, Chapter DOI:10.1007/978-3-030-43120-4_10

78. Acceleration of Subdivision Root-Finding for Sparse Polynomials", In: Computer Algebra in Scienti c Computing, Springer Nature Switzerland AG 2020, F. Boulier et al. (Eds.): CASC 2020, LNCS 12291, Ch. 27, pp. 1 17, 2020 https://doi.org/10.1007/978-3-030-60026-6_27

79. Faster Numerical Univariate Polynomial Root-Finding by Means of Subdivision Iterations" (by Qi Luan, Victor Y. Pan, Wongeun Kim, and Vitaly Zaderman), In: Computer Algebra in Scienti c Computing, Springer Nature Switzerland AG 2020, F. Boulier et al. (Eds.): CASC 2020, LNCS 12291, Ch. 25, pp. 1 16, 2020 https://doi.org/10.1007/978 − 3 − 030 − 60026 − 6_25

10 PUBLICATIONS (COMPLETE LIST CHRONOLOGICALLY)

Also see PUBLICATIONS and RESEARCH in sections 8, 109 and 12
LISTS OF PAN'S PUBLICATIONS IN GOOGLE SCHOLAR AND DBLP INCLUDE

4 research monographs over 20 book chapters and survey articles over

170 refereed publications in journals over 80 refereed publications in

conference proceedings

OVER 12,500 CITATIONS

10.1 CLASSIFICATION BY RESEARCH SUBJECTS

ENUMERATION BELOW IS ACCORDING TO THE ORDER OF (i) THE BOOKS, (ii) SURVEYS AND BOOKS CHAPTERS AND (iii) RESEARCH PAPERS IN THE COMPLETE PUBLICATION LIST.

1. REAL AND COMPLEX FUNCTIONS: papers 1, 3.
2. ECONOMICS: papers 11-13, 15-18
3. LOWER BOUNDS IN ALGEBRAIC COMPUTATIONS: papers 5, 7, 10, 29, 49.
4. FUNDAMENTAL POLYNOMIAL OPERATIONS.

a) EVALUATION: survey paper 1 (containing also research results), research papers 2,4, 6-10, 19, 62, 70, 101, 105, 121, 140, 148, 166, 247-249, 253, 263, 267.

b) INTERPOLATION: papers 62, 70, 105, 120, 140, 148, 159, 166, 247-249, 253, 263,267 and survey paper 12.

c) MULTIPLICATION: papers 24, 118 (multivariate case), 249, 263.

d) DIVISION: papers 42, 48, 51-53, 58, 75, 86, 97, 103, 111, 249, 263.

5. MATH PROGRAMMING.

a) LINEAR PROGRAMMING: papers 13, 38, 39, 43, 50, 55, 57.

b) INTEGER LINEAR PROGRAMMING: papers 89, 93, 113, 152.

c) NONLINEAR PROGRAMMING: paper 79.

6. FAST MATRIX MULTIPLICATION: book 1, survey papers 2 and 24, and papers14, 20, 21, 23, 25, 30, 32, 33, 36, 37, 40, 95, 145, 157, 163, 222.

7. MULTIGRID ALGORITHMS.

a) ALGEBRAIC MULTIGRID: paper 22.

b) COMPACT MULTIGRID: papers 73, 77, 93, 109.

8. PARALLEL COMPUTATIONS (ALSO SEE RELEVANT ITEMS IN PARTS 9-14).

a) PROCESSOR EFFICIENT ALGORITHMS IN NC: book 2 (chapter 4) and papers44-47, 50-58, 60-64, 66-69, 72, 74, 75, 80-82, 85, 88, 90, 93, 100, 102, 103, 106, 107, 111, 112, 115, 117, 119, 123, 125, 126, 129, 131-133, 138, 147, 163, 175.

b) NC EQUIVALENCE OF LINEAR PROGRAMMING AND EUCLIDEAN GCD: pa-pers 89, 113, 152.

c) WORK-PRESERVING SPEED-UP: papers 91, 115, 122.

9. GRAPH ALGORITHMS.

a) MATCHING: papers 45, 63.

b) PATHS: item 6 in the list of book chapters; papers 54, 56, 66, 85, 90, 91, 122, 138, 147.

10. LINEAR SYSTEMS OF EQUATIONS AND MATRIX INVERSION (GENERAL INPUT MATRICES).

a) NEWTON'S ITERATION AND RESIDUAL CORRECTION PROCESSES: book 3(chapter 6), item 9 in the list of reviews and book chapters, and papers 44, 69, 83, 175, 178, 211, 216, 226, 231, 251, 268.

b) RANDOMIZED ALGORITHMS: see section 18.

c) PARALLEL ALGORITHMS: book 2 and papers 44, 47, 60, 67, 74, 81, 82, 91, 122, 175, 238.

11. LINEAR SYSTEMS OF EQUATIONS, MATRIX INVERSION (TRIANGULAR,BANDED OR SPARSE INPUT), AND LOW RANK APPROXIMATION OF MATRICES: item 6 in the list of reviews and book chapters and papers 44, 107, 115, 117, 125, 272, 273.

12. LINEAR SYSTEMS OF EQUATIONS AND MATRIX INVERSION (STRUCTURED INPUT).

a) DISPLACEMENT TRANSFORMATION OF MATRIX STRUCTURE; APPLICA-TIONS TO POLYNOMIAL EVALUATION AND INTERPOLATION: book 3 and papers 71, 76, 140, 150, 156, 203, 248, 253, 259, 267.

b) NEWTON'S ITERATION AND RESIDUAL CORRECTION PROCESSES: book 3(chapter 6), item 9 in the list of reviews and book chapters and papers 72, 83, 88, 93, 106, 132, 141, 178, 179, 187, 200, 201, 204, 211, 216, 229, 231, 251.

c) COMPRESSION OF THE DISPLACEMENTS: book 3 and papers 88, 93, 106, 108,141, 165, 168, 175, 187, 211.

d) HOMOTOPIC/CONTINUATION TECHNIQUES: book 3 (chapter 6) and papers 93, 106, 178, 187, 200, 201, 211, 216, 231.

e) INVERSION OF DISPLACEMENT OPERATORS: book 3 and paper 194.

f) SOLUTION WITH LIFTING TECHNIQUES: papers 192, 226, 238.

g) SOLUTION WITH PRECONDITIONED CONJUGATE GRADIENT METHOD:papers 94, 128.

h) UNIFICATION OF SUPERFAST ALGORITHMS: book 3 and papers 71, 76, 150, 156, 159, 168.

i) OTHER METHODS: books 2 and 3 and papers 60, 62, 72, 74, 81, 102, 131-133, 168,248, 249, 263.

j) APPLICATIONS TO POLYNOMIAL GCD AND RATIONAL INTERPOLATION:papers 133, 149, 159, 166, 195, 267.

k) NORM ESTIMATION: papers 254, 259.

13. DETERMINANT AND CHARACTERISTIC POLYNOMIAL: papers 60, 65, 67, 143, 146, 158, 160, 180, 197, 205, 208, 221.

14. ROOT-FINDING FOR POLYNOMIALS.

a) BOOK 4 and items 8, 10, 13, 18 and 22 in the list of survey articles and book chapters.

b) NEARLY OPTIMAL DIVIDE-AND-CONQUER ALGORITHMS: papers 126, 129,183, 184, 186, 191 and book 4 (chapter 15).

c) OTHER NEARLY OPTIMAL ALGORITHMS: papers: 68, 80, 153, 169, 246, 250,258, 262, 264, 269, 270, 274.

d) STRUCTURED MATRIX METHODS: item 17 in the list of surveys and book chapters; papers 198, 202, 207, 210, 212, 227, 233, 235, 239, 243, 246, 250, 252.

e) REAL POLYNOMIAL ROOT-FINDERS: papers 68, 80, 153, 213, 235, 246, 252, 258, 261.

g) OTHER ROOT-FINDING ALGORITHMS: papers 41, 46, 59, 61, 99, 114, 116, 124,130, 134, 135, 171, 177, 206, 210, 233-236, 239, 241, 242, 264, 271.

h) APPLICATION TO APPROXIMATE POLYNOMIAL GCD: papers 149, 182.

15. ROOT-FINDING FOR SYSTEMS OF POLYNOMIALS: papers 136, 137, 139, 144,151, 155, 170, 176, 185, 189, 193, 197, 208.

16. EIGEN-SOLVING: papers 64, 78, 80, 84, 96, 98, 110, 153, 167, 206, 207, 212, 218,223, 227, 243.

17. SYMBOLIC-NUMERICAL COMPUTATIONS (ALSO SEE PARTS 8, 12-15, and 19).

a) BOOKS AND SURVEYS: books 2 and 3 and 4, 5, 8-14, 16-18 and 22 in the list ofSURVEY ARTICLES AND BOOK CHAPTERS.

b) APPROXIMATE POLYNOMIAL GCD: papers 149, 182.

c) NUMERICAL COMPUTATION OF DETERMINANTS: papers 160, 180, 221.

d) RECOVERY OF A RATIONAL NUMBER FROM ITS NUMERICAL APPROXIMATION: paper 199.

e) NUMERICAL COMPUTATIONS WITH ERROR-FREE OUTPUT: papers 154, 224.

18. RANDOMIZED MATRIX ALGORITHMS: papers 214, 215, 217, 219, 221, 223, 225,228, 230, 232, 234, 237, 244, 245, 255, 257, 260, 265, 266.

19. DEGENERACY AND CONDITIONING: 220, 254, 259.

20. BOOLEAN COMPLEXITY OF ALGEBRAIC COMPUTATIONS: papers 26, 28, 31, 34, 40, 49, 53, 58, 73, 77, 246, 249, 250, 258, 262-264.

21. MANIPULATION WITH INTEGERS:

(a) BINARY SEGMENTATION: book 1, papers 87, 158, 224.

(b) RATIONAL RECONSTRUCTION, EUCLIDEAN ALGORITHM: papers 190, 196, 199.

22. LINEAR RECURRENCES: papers 142, 172.

10.2 BOOKS

1. How to Multiply Matrices Faster", Lecture Notes in Computer Science, vol. 179 (XI + 212 pages), Springer, Berlin (1984).

2. Polynomial and Matrix Computations", Volume 1: Fundamental Algorithms" (XVI + 415 pages) (by D. Bini and V. Y. Pan), in the series Progress in Theoretical Computer Science (R.V. Book editor), Birkh user, Boston (1994).

3. Structured Matrices and Polynomials: Uni ed Superfast Algorithms" (XXV + 278 pages), Birkh user/Springer, Boston/New York (June 2001).
4. Numerical Methods for Roots of Polynomials" (by J. M. McNamee and V. Y. Pan), Part 2 (XXII + 718 pages), Elsevier (2013).

10.3 CHAPTERS IN BOOKS AND SURVEY ARTICLES, including prefaces to Special Issues (items 1, 2, 3, 4-6, 8, 9 and 17 included new research results)

1. "On Methods of Computing the Values of Polynomials", Uspekhi Matematicheskikh Nauk, 21, 1 (127), 103 134 (1966). (Transl. Russian Mathematical Surveys, 21, 1 (127), 105 137 (1966).)
2. "How Can We Speed Up Matrix Multiplication?", SIAM Review, 26, 3, 393 415 (1984).
3. "Linear Systems of Algebraic Equations", in Encyclopedia of Physical Sciences andTechnology, 7, 304 329 (1987), (rst edition, Marvin Yelles, editor); 8, 779 804 (1992) (second edition), and 8, 617 642 (2001) (third edition, Robert A. Meyers, editor), Academic Press, San Diego, California.
4. "Complexity of Algorithms for Linear Systems of Equations", in Computer Algorithmsfor Solving Linear Algebraic Equations (The State of the Art), (E. Spedicato, editor), NATO ASI Series, Series F: Computer and Systems Sciences, 77, 27 56, Springer, Berlin (1991) and Academic Press, Dordrecht, the Netherlands (1992).
5. "Complexity of Computations with Matrices and Polynomials," SIAM Review, 34, 2,225 262 (1992).
6. "Parallel Solution of Sparse Linear and Path Systems", in Synthesis of Parallel Algorithms (J. H. Reif, editor), Chapter 14, pages 621 678, Morgan Kaufmann publishers, San Mateo, CA (1993).
7. "Algebraic Algorithms" (by A. Diaz, E. Kaltofen and V. Y. Pan), Chapter 10 in theComputer Science and Engineering Handbook (Allen B. Tucker, Jr., editor), pages 226 249, CRC Press Inc., Boca Raton, Florida (1997).
8. "Solving a Polynomial Equation: Some History and Recent Progress", SIAM Review,39, 2, 187 220 (1997).
9. Newton’s Iteration for Structured Matrices and Linear Systems of Equations" (by V.Y. Pan, S. Branham, R. Rosholt, and A. Zheng), SIAM volume on Fast Reliable Algorithms for Matrices with Structure (T. Kailath and A. H. Sayed, editors), chapter 7, pages 189 210, SIAM Publications, Philadelphia (1999).
10. Solving Polynomials with Computers", American Scientist, 86, 62 69 (JanuaryFebruary 1998).
11. "Computational Complexity of Solving Large Sparse and Large Special Linear Systems of Equations", pages 1 24, in "Algorithms for Large Scale Linear Algebraic Systems: Applications in Science and Engineering" (G. Winter Althaus and E. Spedicato, editors), NATO Advanced Science Institute Series, Kluwer Academic Publishers, Dordrecht, The Netherlands (1998).
12. "Some Recent Algebraic/Numerical Algorithms", Electronic Proc. IMACS/ACA’98 (1998): http:www-troja.fj .cvut.cz/aca98/sessions/approximate
13. "Algebraic Algorithms" (by A. Diaz, I.Z. Emiris, E. Kaltofen and V. Y. Pan), Chapter 16 in Handbook "Algorithms and Theory of Computations", pages from 16 1 to 16 27 (M. Atallah, editor), CRC Press Inc., Boca Raton, Florida (1999).
14. "Fast Fourier Transform and Its Applications" (by I. Z. Emiris and V. Y. Pan),Chapter 17 in Handbook "Algorithms and Theory of Computations", pages from 17 1 to 17 30 (M. Atallah, editor), CRC Press Inc., Boca Raton, Florida (1999).

15. Preface to the Special Issue on Algebraic and Numerical Algorithms" (by I. Z. Emiris, B. Mourrain, and V. Y. Pan), Theoretical Computer Science, 315, 2 3, 307 308 (2004). 16. "Algebraic Algorithms" (by A. Diaz, I.E. Emiris, E. Kaltofen and V. Y. Pan), Chapter 8 in the Computer Science and Engineering Handbook (Allen B. Tucker, editor), pages from 8 1 to 8 24, Chapman and Hall/CRC Press (2004).

17. Root nding with Eigen-solving" (by V. Y. Pan, D. Ivolgin, B. Murphy, R. E. Rosholt, Y. Tang, X. Wang, and X. Yan), pages 185 210 in Symbolic Numeric Computation (Dongming Wang and Lihong Zhi, editors), Birkh user, Basel/Boston (2007).

18. Preface to the Special Issue on Symbolic Numerical Algorithms" (by D. A. Bini, V. Y. Pan, and J. Verschelde), Theoretical Computer Science, 409, 2, 155 157 (2008).

19. Algebraic and Numerical Algorithms" (by I. Z. Emiris, V. Y. Pan, and E. Tsigaridas), in Algorithms and Theory of Computations Handbook", Second Edition, Volume 1 (1016 pages): General Concepts and Techniques, pages 1 34 in Chapter 17 (Mikhail J. Atallah and Marina Blanton, editors), CRC Press Inc., Boca Raton, Florida (2009).

20. "Fast Fourier Transform and Its Applications" (by I. Z. Emiris and V. Y. Pan),in Algorithms and Theory of Computations Handbook", Second Edition, Volume 1 (1016 pages): General Concepts and Techniques, pages 1 31 in Chapter 18 (Mikhail J. Atallah and Marina Blanton, editors), CRC Press Inc., Boca Raton, Florida (2009).

21. Preface to the Special Issue on Symbolic and Numerical Algorithms" (by I. S. Kotsireas, B. Mourrain, and V. Y. Pan), Theoretical Computer Science, 412, 16, 1443 1444 (2011).

22. Preface to the Special Issue on Symbolic and Numerical Algorithms" (by I. S. Kotsireas, B. Mourrain, V. Y. Pan, and L. Zhi), Theoretical Computer Science, 479, 1 3 (2013).

23. Algebraic Algorithms" (by I. Z. Emiris, V. Y. Pan, and E. Tsigaridas), Chapter 10 (pages from 10 1 to 10-40) of Computing Handbook (Third edition), Volume I: Computer Science and Software Engineering (Allen B. Tucker, Teo Gonzales, and Jorge L. Diaz-Herrera, editors), Taylor and Francis Group (2014).

24. "Fast Matrix Multiplication and Its Algebraic Neighborhood", DOI: 10.4213/sm8833 Matem. Sbornik, 208, 11, 90 138 (2017) and DOI:10.1070/SM8833: Sb. Math., 208, 11, 1661 1704 (2017).

10.4 RESEARCH PAPERS (in journals and refereed proceedings of conferences)

1. "On One Question by N.N. Luzin", Nauchnye Doklady Vysshey Schkoly, PhizikoMatematicheskie Nauki (in Russian), 4, 59 62 (1958).

2. "Some Schemes for the Evaluation of Polynomials with Real Coe cients", Doklady Akademii Nauk SSSR (in Russian), 127, 2, 266 269 (1959).

3. "On Approximation of Analytic Functions by Rational Ones", Uspekhi Matematicheskikh Nauk (in Russian), 16, 5 (101), 195 197 (1961).

4. "Some Schemes for the Evaluation of Polynomials with Real Coe cients" Problemy Kibernetiki (in Russian), (edited by A.A. Lyapunov), 5, 17 29 (1961). (Transl. Problems of Cybernetics, USSR, 5, 14 32, Pergamon Press (1961).)

5. "On Some Methods of Computing Polynomial Values", Problemy Kibernetiki (in Russian), (edited by A.A. Lyapunov), 7, 21 30 (1962). (Transl. Problems of Cybernetics, USSR, 7, 20 30, U.S. Dept. of Commerce (1962).)

6. "Schemes with Preconditioning for the Evaluation of Polynomials and a Program forAutomatic Preconditioning", Zhurnal Vychislitel’noy Matematiki i Matematicheskoy Fiziki (in Russian), 2, 1, 133 140 (1962). (Transl. from USSR Computational Mathematics and Mathematical Physics, 1, 137 146 (1963).).
7. "Methods for Computing Polynomials" (in Russian), Ph.D. thesis, Dept. of Mechanicsand Mathematics, Moscow State University (1964).
8. "The Evaluation of Polynomials of the Fifth and Seventh Degrees with Real Coefcients", Zhurnal Vychislitel’noy Matematiki i Matematicheskoy Fiziki (in Russian), 5, 1, 116 118 (1965). (Transl. USSR Computational Mathematics and Mathematical Physics, 5, 1, 159 161 (1965).)
9. "On Simultaneous Evaluation of Several Polynomials of Low Degree (Two to Five)",Zhurnal Vychislitel’noy Matematiki i Matematicheskoy Fiziki (in Russian), 6, 2, 352 357
(1966). (Transl. USSR Computational Mathematics and Mathematical Physics, 6, 2, 222
227 (1966).)
10. "On Methods of Computing the Values of Polynomials", Uspekhi MatematicheskikhNauk (in Russian), 21, 1 (127), 103 134 (1966). (Transl. Russian Mathematical Surveys, 21, 1 (127), 105 137 (1966).)
11. "Calculus of Rational Costs Based on Modern Economic Information" (by V. Belkin,A. Kronrod, Y. Nazarov, and V. Y. Pan), Ekonomika i Matematicheskie Metody (in Russian), Akademiya Nauk SSSR, 1, 5, 699 717 (1965).
12. "A Linear Model and Algorithm for Optimizing Foreign Trade", Tezisy Dokladov iVystupleniy na Simpoziume po Modelirovaniyu Narodnogo Khozyaistva, Institut Ekonomiki AN SSSR (Proceedings of the Symposium on Models of Public Economy, Institute of Economics, Academy of Sciences of USSR), (in Russian), Moscow, 29 37 (1970).
13. "On Solving a Distribution Problem with Upper Bounds on the Variables and aSimpli ed Criterion for Optimizing Foreign Trade", Trudy 4-oy Zimney Schkoly po Matematicheskomu Programmirovaniyu i Smezhnym Voprosam (in Russian), (Transactions of the 4-th Winter School on Mathematical Programming and Adjacent Problems), (edited by S. I. Zukhovitskiy), Iss. 5, 26 49, Drogobych (1972).
14. "On Schemes for the Evaluation of Products and Inverses of Matrices", UspekhiMatematicheskikh Nauk (in Russian), 27, 5 (167), 249 250 (1972). Also see translation into English: "Better Late Than Never: Filling a Void in the History of Fast Matrix Multiplication and Tensor Decompositions", arXiv:1411.1972 (6 Nov 2014)
15. "Models for Planning Costs with Optimization of Foreign Trade of Several Countries under the Economic Integration", Primenenie Ekonomiko Matematicheskikh Modeley i EVM pri Planirovanii i Prognozirovanii Tsen (in Russian), 5 (1973).
16. "A Model for the Optimization of Foreign Economic Relations Under the EconomicIntegration of the Socialist Countries", Ekonomika i Matematicheskie Metody (in Russian), Akademiya Nauk SSSR, 10, 2, 255 266 (1974). (Transl. USSR Trade and Services, 755, 1 16 (1974).)
17. "A Model for the Dynamics of Costs and Expenditures", Trudy Mezhdunarodnoy Konferentsii "Modelirovanie Ekonomicheskikh Protsessov" (Transactions of the International Conference, "Models of Economic Processes"), (in Russian), Erevan, 166 174 (1974).
18. "A Modi cation of a Balanced Model of Income Commodities into an EquilibriumModel" (by V. Belkin, V. Ivanter, N. Konstantinov, and V. Pan), Ekonomika i Matematicheskie Methody, Akademiya Nauk SSSR (in Russian), 11, 6, 1037 1049 (1975).
19. "Computational Complexity of Computing Polynomials over the Fields of Real and Complex Numbers", Proceedings of the Tenth Annual ACM Symposium on Theory of Computing (STOC’78), 162 172, ACM Press, New York (1978).

20. "Strassen’s Algorithm Is Not Optimal. Trilinear Technique of Aggregating, Unitingand Canceling for Constructing Fast Algorithms for Matrix Multiplication", Proceedings of the 19th Annual IEEE Symposium on Foundations of Computer Science (FOCS’78), 166 176, IEEE Computer Society Press, Long Beach, California (1978).
21. "Fields Extension and Trilinear Aggregating, Uniting and Canceling for the Acceleration of Matrix Multiplication", Proceedings of the 20th Annual IEEE Symposium on Foundations of Computer Science (FOCS’79), 28 38, IEEE Computer Society Press, Long Beach, California (1979).
22. "Methods of Aggregations" (by W. L. Miranker and V. Y. Pan), Linear Algebra andIts Applications, 29, 231 257 (1980).
23. "New Fast Algorithms for Matrix Operations", SIAM J. on Computing, 9, 2, 321 342 (1980).
24. "Convolution of Vectors over the Real Field of Constants", Journal of Algorithms,1, 297 300 (1980).
25. "New Combinations of Methods for the Acceleration of Matrix Multiplications", Computers and Mathematics (with Applications), 7, 73 125 (1981).
26. "The Bit Complexity of Arithmetic Algorithms", Journal of Algorithms, 2, 144 163 (1981).
27. "A Uni ed Approach to the Analysis of Bilinear Algorithms", Journal of Algorithms,2, 301 310 (1981).
28. "The Bit Operation Complexity of Matrix Multiplication and All Pair Shortest PathProblem", Computers and Mathematics (with Applications), 7, 5, 431 438 (1981).
29. "The Lower Bounds on the Additive Complexity of Bilinear Problems in Terms ofSome Algebraic Quantities", Information Processing Letters, 13, 2, 71 72 (1981).
30. "Trilinear Aggregating with Implicit Canceling for a New Acceleration of MatrixMultiplication", Computers and Mathematics (with Applications), 8, 1, 23 34 (1982).
31. "The Bit Operation Complexity of Approximate Evaluation of Matrix and Polynomial Products Using Modular Arithmetic", Computers and Mathematics (with Applications), 8, 2, 137 140 (1982).
32. "Fast Matrix Multiplication without APA Algorithms", Computers and Mathematics (with Applications), 8, 5, 343 366 (1982).
33. "Trilinear Aggregating is the Basis for the Asymptotically Fastest Known Algorithms for Matrix Multiplication", Conference Record, Second Conference on Foundations of Software Technology and Theoretical Computer Science (FST and TCS’82), 321 337, Indian Institute of Technology, Bangalore, India (December 1982).
34. "The Additive and Logical Complexities of Linear and Bilinear Algorithms", J. ofAlgorithms, 4, 1 34 (1983).
35. "The Projective Power Method for the Algebraic Eigenvalue Problem", Computersand Mathematics (with Applications), 9, 6, 735 745 (1983).
36. "Trilinear Aggregating Is the Basis for the Asymptotically Fastest Known Algorithmsfor Matrix Multiplication" (extended abstract), Methods of Operations Research, 45, 493 494 (1983).
37. "Trilinear Aggregating and the Recent Progress in the Asymptotic Acceleration ofMatrix Operations", Theoretical Computer Science, 33, 117 138 (1984).
38. "How Fast Can We Solve a System of Linear Inequalities in the Worst and in theAverage Case?", Methods of Operations Research, 51, 107 118 (1984).
39. "Fast Finite Methods for a System of Linear Inequalities", Computers and Mathematics (with Applications), 11, 4, 355 394 (1985).

40. "The Bit Complexity of Matrix Multiplication and of the Related Computations inLinear Algebra. The Segmented λalgorithms", Computers and Mathematics (with Applications), 11, 9, 919 928 (1985).
41. "On Application of Some Recent Techniques of the Design of Algebraic Algorithmsto the Sequential and Parallel Evaluation of the Roots of a Polynomial and to Some Other Numerical Problems", Computers and Mathematics (with Applications), 11, 9, 911 917
(1985).
42. "Fast Parallel Polynomial Division via Reduction to Polynomial Inversion Modulo aPower" (by D. Bini and V. Y. Pan), Information Processing Letters, 21, 79 81 (1985).
43. "On the Complexity of a Pivot Step of the Revised Simplex Algorithm", Computersand Mathematics (with Applications), 11, 11, 1127 1140 (1985).
44. "E cient Parallel Solution of Linear Systems" (by V. Y. Pan and J. Reif), Proc. 17th Annual ACM Symposium on Theory of Computing (STOC'85), 143 152, ACM Press, New York (1985).
45. "Improved Processor Bounds for Algebraic and Combinatorial Problems in RNC" (byZ. Galil and V. Y. Pan), Proc. 26th Annual IEEE Symposium on Foundations of Computer Science (FOCS'85), 490 495, IEEE Computer Society Press, Los Angeles, California (1985).
46. "Fast and E cient Algorithms for Sequential and Parallel Evaluation of Polynomial Zeros and of Matrix Polynomials," Proc. 26th Annual IEEE Symposium on Foundations of Computer Science (FOCS'85), 522 531, IEEE Computer Society Press, Los Angeles,
California (1985).
47. "Fast and E cient Parallel Algorithms for the Exact Inversion of Integer Matrices", Proc. Fifth Conference on Foundations of Software Technology and Theoretical Computer Science (FST and TCS'85), (edited by K. V. Nori), Lecture Notes in Computer Science, 206, 504 521, Springer, Berlin (1985).
48. "Algorithms for Polynomial Division" (by D. Bini and V. Y. Pan), Proc. EuropeanConference on Computer Algebra, Linz, Austria, Lecture Notes in Computer Science, 204, 1 3, Springer (1985).
49. "The Trade-o Between the Additive Complexity and Asynchronicity of Linear and Bilinear Algorithms", Information Processing Letters, 22, 11 14 (1986).
50. "Fast and E cient Parallel Linear Programming and Least Squares Computations" (by V. Y. Pan and J. Reif), VLSI Algorithms and Architectures, Lecture Notes in Computer Science, 227, 283 295, Springer, Berlin (1986).
51. "A Logarithmic Boolean Time Algorithm for Parallel Polynomial Division" (by D. Bini and V. Y. Pan), VLSI Algorithms and Architectures, Lecture Notes in Computer Science, 227, 246 251, Springer, Berlin (1986).
52. "Polynomial Division and Its Computational Complexity" (by D. Bini and V. Y.Pan), Journal of Complexity, 2, 179 203 (1986).
53. "Fast Parallel Algorithms for Polynomial Division over Arbitrary Field of Constants"(by D. Bini and V. Y. Pan), Computers and Mathematics (with Applications), 12A, 11, 1105 1118 (1986).
54. "Extension of the Parallel Nested Dissection Algorithm to the Path Algebra Problems" (by V. Y. Pan and J. Reif), Proc. Sixth Conference on Foundations of Software Technology and Theoretical Computer Science (FST and TCS'86), (New Delhi, India), Lecture Notes in Computer Science, 241, 470 487, Springer, Berlin (1986).
55. "Fast and E cient Linear Programming and Linear Least-Squares Computations" (by V. Y. Pan and J. Reif), Computers and Mathematics (with Applications), 12A, 12, 1217 1227 (1986).
56. "Parallel Nested Dissection for Path Algebra Computations" (by V. Y. Pan and J.Reif), Operations Research Letters, 5, 4, 177 184 (1986).

57. "E cient Parallel Linear Programming" (by V. Y. Pan and J. Reif), Operations Research Letters, 5, 3, 127 135 (1986).
58. "A Logarithmic Boolean Time Algorithm for Parallel Polynomial Division" (by D.Bini and V. Y. Pan), Information Processing Letters, 24, 233 237 (1987).
59. "Algebraic Complexity of Computing Polynomial Zeros", Computers and Mathematics (with Applications), 14, 4, 285 304 (1987).
60. "Complexity of Parallel Matrix Computations", Theoretical Computer Science, 54,65 85 (1987).
61. "Sequential and Parallel Complexity of Approximate Evaluation of Polynomial Zeros", Computers and Mathematics (with Applications), 14, 8, 591 622 (1987).
62. "Some Polynomial and Toeplitz Matrix Computations" (by V. Y. Pan and J. Reif),Proc. 28th Annual IEEE Symposium on Foundations of Computer Science (FOCS'87), 173 184, IEEE Computer Society Press, Los Angeles, California (1987).
63. "Improved Processor Bounds for Combinatorial Problems in RNC" (by Z. Galil and V. Y. Pan), Combinatorica, 8, 2, 189 200 (1988).
64. "E cient Algorithms for the Evaluation of the Eigenvalues of (Block) Banded Toeplitz Matrices" (by D. Bini and V. Y. Pan), Mathematics of Computation, 50, 182, 431 448 (1988).
65. "Computing the Determinant and the Characteristic Polynomial of a Matrix viaSolving Linear Systems of Equations", Information Processing Letters, 28, 2, 71 75 (1988).
66. "Fast and E cient Solution of Path Algebra Problems" (by V. Y. Pan and J. Reif), Journal of Computer and Systems Sciences, 38, 3, 494 510 (1989).
67. "Parallel Evaluation of the Determinant and of the Inverse of a Matrix" (by Z. Galiland V. Y. Pan), Information Processing Letters, 30, 41 45 (1989).
68. "Fast and E cient Parallel Evaluation of the Zeros of a Polynomial Having Only Real Zeros", Computers and Mathematics (with Applications), 17, 11, 1475 1480 (1989).
69. "Fast and E cient Parallel Solution of Dense Linear Systems" (by V. Y. Pan and J. Reif), Computers and Mathematics (with Applications), 17, 11, 1481 1491 (1989).
70. "Fast Evaluation and Interpolation at the Chebyshev Sets of Points", Applied Mathematics Letters, 2, 3, 255 258 (1989).
71. "On Some Computations with Dense Structured Matrices", Proc. Annual ACMSIGSAM International Symposium on Symbolic and Algebraic Computation (ISSAC'89), 34 42, ACM Press, New York (1989).
72. "Fast and E cient Parallel Inversion of Toeplitz and Block Toeplitz Matrices", Operator Theory: Advances and Applications, 40, 359 389, Birkh user, Basel, Switzerland (1989).
73. "On the Bit-Complexity of Discrete Solution of PDEs: Compact Multigrid" (by V.Y. Pan and J. Reif), Proc. 17th International Colloquium on Automata, Languages and Programming (ICALP'90), Springer's Lecture Notes in Computer Science, 443, 612 625 (1990).
74. "Parallel Least-Squares Solution of General and Toeplitz like Linear Systems", Proc.2nd Annual ACM Symposium on Parallel Algorithms and Architecture (SPAA'90), 244 253, ACM Press, New York (1990).
75. "Parallel Polynomial Computations by Recursive Processes" (by D. Bini and V. Y.Pan), Proc. Annual ACM International Symposium on Symbolic and Algebraic Computation (ISSAC'90), 294, ACM Press, New York (1990).
76. "On Computations with Dense Structured Matrices", Mathematics of Computation,55, 191, 179 190 (1990).

77. "The Bit-Complexity of the Discrete Solution of PDEs: Compact Multigrid" (by V. Y. Pan and J. Reif), Computers and Mathematics (with Applications), 20, 2, 9 16 (1990).
78. "Estimating the Extremal Eigenvalues of a Symmetric Matrix", Computers and Mathematics (with Applications), 20, 2, 17 22 (1990).
79. "The Modi ed Barrier Function Method for Linear Programming and Its Extensions", Computers and Mathematics (with Applications), 20, 3, 1 14 (1990).
80. "Parallel Complexity of Tridiagonal Symmetric Eigenvalue Problem" (by D. Bini andV. Y. Pan), Proc. 2nd Annual ACM-SIAM Symposium on Discrete Algorithms (SODA'91), 384 393, ACM Press, New York, and SIAM Publications, Philadelphia (1991).
81. "Improved Parallel Computations with Matrices and Polynomials" (by D. Bini, L.Gemignani, and V. Y. Pan), Proc. 18th International Colloquium on Automata, Languages and Programming (ICALP'91), Lecture Notes in Computer Science, 510, 520 531, Springer, Berlin (1991).
82. "Processor E cient Parallel Solution of Linear Systems over an Abstract Field" (by E. Kaltofen and V. Y. Pan), Proc. 3rd Annual ACM Symposium on Parallel Algorithms and Architectures (SPAA'91), 180 191, ACM Press, New York (1991).
83. "An Improved Newton Iteration for the Generalized Inverse of a Matrix, with Applications" (by V. Y. Pan and R. Schreiber), SIAM J. on Scienti c and Statistical Computing, 12, 5, 1109 1131 (1991).
84. "On the Evaluation of the Eigenvalues of a Banded Toeplitz Block Matrix" (by D.Bini and V. Y. Pan), J. of Complexity, 7, 408 424 (1991).
85. "The Parallel Computation of the Minimum Cost Paths in Graphs by Stream Contraction" (by V. Y. Pan and J. Reif), Information Processing Letters, 40, 79 83 (1991).
86. "Univariate Polynomial Division with a Remainder by Means of Evaluation and Interpolation" (by V. Y. Pan, E. Landowne, and A. Sadikou), Proc. of 3rd IEEE Symposium on Parallel and Distributed Processing, 212 217, IEEE Computer Society Press, Los Alamitos, California (1991).
87. "Extended Concept of Signi cant Digits and Lower Precision Computations", Applied Mathematics Letters, 5, 2, 3 6 (1992).
88. "Parametrization of Newton's Iteration for Computations with Structured Matricesand Applications", Computers and Mathematics (with Applications), 24, 3, 61 75 (1992).
89. "On Parallel Complexity of Integer Linear Programming, GCD and the Iterated ModFunction" (by Yu Lin-Kriz and V. Y. Pan), Proc. 3rd Annual ACM-SIAM Symposium on Discrete Algorithms (SODA'92), 124 137, ACM Press, New York and SIAM Publications, Philadelphia (1992).
90. "E cient Parallel Algorithms for Computing All Pair Shortest Paths in Directed Graphs" (by Y. Han, V. Y. Pan, and J. Reif), Proc. 4th Annual ACM Symposium on Parallel Algorithms and Architectures (SPAA'92), 353 362, ACM Press, New York (1992).
91. "Supere ective Slow-down of Parallel Computations" (by V. Y. Pan and F. P. Preparata), Proc. 4th Annual ACM Symposium on Parallel Algorithms and Architectures (SPAA'92), 402 409, ACM Press, New York (1992).
92. "Compact Multigrid" (by V. Y. Pan and J. Reif), SIAM J. on Scienti c and Statistical Computing, 13, 1, 119 127 (1992).
93. "Parallel Solution of Toeplitz like Linear Systems", J. of Complexity, 8, 1 21 (1992). 94. "A Fast, Preconditioned Conjugate Gradient Toeplitz Solver" (by V. Y. Pan and R. Schreiber), Computers and Mathematics (with Applications), 24, 7, 17 24 (1992).
95. "On Practical Algorithms for Accelerated Matrix Multiplication" (by J. Laderman, V. Y. Pan, and H.-X. Sha), Linear Algebra and Its Applications, 162 164, 557 588 (1992).

96. "Practical Improvement of the Divide-and-Conquer Eigenvalue Algorithms" (by D.Bini and V. Y. Pan), Computing, 48, 109 123 (1992).
97. "Polynomial Division with a Remainder by Means of Evaluation and Interpolation"(by V. Y. Pan, E. Landowne, and A. Sadikou), Information Processing Letters, 44, 149 153 (1992).
98. "Improving the Solution of the Symmetric Eigenvalue Problem and an Extension",Applied Mathematics Letters, 5, 6, 49 50 (1992).
99. "New Resultant Inequalities and Complex Polynomial Factorization", Proc. 1st Israel Symposium on Theory of Computing and Systems (ISTCS '92), Lecture Notes in Computer Science, 601, 122 136, Springer, Berlin (1992).
100. "Improved Parallel Computations with Toeplitz like and Hankel like Matrices" (by D. Bini and V. Y. Pan), Proc. of Annual ACM-SIGSAM International Symposium on Symbolic and Algebraic Computation (ISSAC'93), 193 200, ACM Press, New York (1993). 101. "The Power of Combining the Techniques of Algebraic and Numerical Computing: Improved Approximate Multipoint Polynomial Evaluation and Improved Multipole Algorithms" (by V. Y. Pan, J. Reif, and S. Tate), Proc. of 33rd Annual IEEE Symposium on Foundations of Computer Science (FOCS '92), 703 713, IEEE Computer Society Press, Los Alamitos, California (1992).
102. "Processor-E cient Parallel Solution of Linear Systems II. The Positive Characteristic and Singular Cases" (by E. Kaltofen and V. Y. Pan), Proc. of 33rd Annual IEEE Symposium on Foundations of Computer Science (FOCS'92), 714 723, IEEE Computer Society Press, Los Alamitos, California (1992).
103. "Improved Parallel Polynomial Division and Its Extension" (by D. Bini and V.Y. Pan), Proc. of 33rd Annual IEEE Symposium on Foundations of Computer Science (FOCS'92), 131 136, IEEE Computer Society Press, Los Alamitos, California (1992).
104. "Binary Segmentation for Matrix and Vector Operations", Computers and Mathematics (with Applications), 25, 3, 69 71 (1993).
105. "A New Approach to Fast Polynomial Interpolation and Multipoint Evaluation"(by V. Y. Pan, A. Sadikou, E. Landowne, and O. Tiga), Computers and Mathematics (with Applications), 25, 9, 25 30 (1993).
106. "Concurrent Iterative Algorithm for Toeplitz like Linear Systems", IEEE Trans.
on Parallel and Distributed Systems, 4, 5, 592 600 (1993).
107. "Fast and E cient Parallel Solution of Sparse Linear Systems" (by V. Y. Pan and J. Reif), SIAM J. on Computing, 22, 6, 1227 1250 (1993).
108. "Decreasing the Displacement Rank of a Matrix", SIAM J. on Matrix Analysis andApplications, 14, 1, 118 121 (1993).
109. "Generalized Compact Multigrid" (by V. Y. Pan and J. Reif), Computers and Mathematics (with Applications), 25, 9, 3 5 (1993).
110. "A New Algorithm for the Symmetric Tridiagonal Eigenvalue Problem" (by V. Y.Pan and J. Demmel), Journal of Complexity, 9, 387 405 (1993).
111. Improved Parallel Polynomial Division" (by D. Bini and V. Y. Pan), SIAM J. on Computing, 22, 3, 617 627 (1993).
112. "Improved Parallel Computations with Toeplitz like and Hankel like Matrices" (byD. Bini and V. Y. Pan), Linear Algebra and Its Applications, 188, 189, 3 29 (1993).
113. "The NC-Equivalence of Planar Integer Linear Programming and Euclidean GCD"(by D. Shallcross, V. Y. Pan, and Yu Lin-Kriz), Proc. 34th Annual IEEE Symposium on Foundations of Computer Science (FOCS'93), 557 564, IEEE Computer Society Press, Los Alamitos, California (1993).

114. "New Resultant Inequalities and Complex Polynomial Factorization", SIAM Journal on Computing 23, 5, 934 950 (1994).
115. "Improved Parallel Solution of a Triangular Linear System", Computers and Mathematics (with Applications), 27, 11, 41 43 (1994).
116. "New Techniques for Approximating Complex Polynomial Zeros", Proc. 5th AnnualACM-SIAM Symposium on Discrete Algorithms (SODA'94), 260 270, ACM Press, New York, and SIAM Publications, Philadelphia (1994).
117. "Optimum Parallel Computations with Band Matrices" (by V. Y. Pan, I. Sobze,and A. Atinkpahoun), Proc. 5th Annual ACM-SIAM Symposium on Discrete Algorithms (SODA'94), 649 658, ACM Press, New York, and SIAM Publications, Philadelphia (1994).
118. "Simple Multivariate Polynomial Multiplication", J. Symbolic Computation, 18,183 186 (1994).
119. "Parallel Solution of Toeplitz and Toeplitz like Linear Systems over Fields of SmallPositive Characteristic" (by E. Kaltofen and V. Y. Pan), Proceedings of First International Symposium on Parallel Symbolic Computation (PASCO'94), Linz, Austria (Sept. 1994), Lecture Notes Series in Computing, 5, 225 233, World Scienti c Publishing Company, Singapore (1994).
120. "Algebraic Improvement of Numerical Algorithms: Interpolation and Economization of Taylor Series", Mathematical and Computer Modeling, 20, 1, 23 26 (1994).
121. "An Algebraic Approach to Approximate Evaluation of a Polynomial on a Set ofReal Points", Advances in Computational Mathematics, 3, 41 58 (1995).
122. "Work-Preserving Speed-up of Parallel Matrix Computations" (by V. Y. Pan andF. P. Preparata), SIAM J. on Computing, 24, 4, 811 821 (1995).
123. "Parallel Computation of a Krylov Matrix for a Sparse and Structured Input",Mathematical and Computer Modeling, 21, 11, 97 99 (1995).
124. "Deterministic Improvement of Complex Polynomial Factorization Based on theProperties of the Associated Resultant", Computers and Mathematics (with Applications), 30, 2, 71 94 (1995).
125. "On Parallel Computations with Band Matrices" (by V. Y. Pan, I. Sobze, and A.Atinkpahoun), Information and Computation, 120, 2, 237 250 (1995).
126. "Optimal (up to Polylog Factors) Sequential and Parallel Algorithms for Approximating Complex Polynomial Zeros", Proc. 27th Annual ACM Symposium on Theory of Computing (STOC'95), 741 750, ACM Press, New York (1995).
127. "Weyl's Quadtree Algorithm for Unsymmetric Eigenvalue Problem", Applied Mathematics Letters, 8, 5, 87 88 (1995).
128. "A Fast, Preconditioned Conjugate Gradient Toeplitz and Toeplitz like Solver"(by V. Y. Pan, A. Zheng, O. Dias, and X. Huang), Computers and Mathematics (with Applications), 30, 8, 57 63 (1995).
129. "Optimal and Nearly Optimal Algorithms for Approximating Polynomial Zeros",Computers and Mathematics (with Applications), 31, 12, 97 138 (1996).
130. "Computing $x^n \bmod p(x)$ and an Application to Splitting a Polynomial into Factors over a Fixed Disc", Journal of Symbolic Computations, 22, 377 380 (1996).
131. "A New Approach to Parallel Computation of Polynomial GCDs and to RelatedParallel Computations over Fields and Integer Rings", Proc. 7th Annual ACM-SIAM Symposium on Discrete Algorithms (SODA'96), 518 527, ACM Press, New York, and SIAM
Publications, Philadelphia (1996).
132. "E ective Parallel Computations with Toeplitz and Toeplitz like Matrices Filledwith Integers", in Proc. of the AMS-SIAM Workshop "Mathematics of Numerical Analysis", Park City, Utah, July-August 1995

(J. Renegar, M. Shub, and S. Smale, editors), Lectures in Applied Math., 32, 593 641, American Mathematical Society Press, Providence, Rhode Island (1996).
133. "Parallel Computation of Polynomial GCD and Some Related Parallel Computations over Abstract Fields", Theoretical Computer Science, 162, 2, 173 223 (1996).
134. "On Isolation of Real and Nearly Real Zeros of a Univariate Polynomial and ItsSplitting into Factors" (by V. Y. Pan, M.-h. Kim, A. Sadikou, X. Huang, and A. Zheng), J. of Complexity, 12, 572 594 (1996).
135. "Grae e's, Chebyshev like, and Cardinal's Processes for Splitting a Polynomial into Factors" (by D. Bini and V. Y. Pan), J. of Complexity, 12, 492 511 (1996).
136. "Multidimensional Structured Matrices and Polynomial Systems" (by B. Mourrainand V. Y. Pan), Calcolo (Special Issue on Toeplitz Matrices: Structure, Algorithms and Applications), 33, 389 401 (1996).
137. "Techniques for Exploiting Structure in Matrix Formulae of the Sparse Resultant"(by I. Z. Emiris and V. Y. Pan), Calcolo (Special Issue on Toeplitz Matrices: Structure, Algorithms and Applications), 33, 353 369 (1996).
138. "E cient Parallel Algorithms for Computing All Pair Shortest Paths in Directed Graphs" (by Y. Han, V. Y. Pan, and J. Reif), Algorithmica, 17, 399 415 (1997).
139. "Solving Special Polynomial Systems by Using Structured Matrices and Algebraic Residues" (by B. Mourrain and V. Y. Pan), Proc. of the AMS-SIAM Conference on Foundation of Computational Mathematics (FoCM'97), (F. Cucker and M. Shub, editors), Rio-de-Janeiro, January 1997, 287 304, Springer (1997).
140. "Fast Multipoint Polynomial Evaluation and Interpolation via Computation withStructured Matrices" (by V. Y. Pan, A. Zheng, X. Huang, and Y. Yu), Annals of Numerical Math., 4, 483 510 (1997).
141. "Newton's Iteration for Inversion of Cauchy like and Other Structured Matrices"(by V. Y. Pan, A. Zheng, X. Huang, and O. Dias), J. of Complexity, 13, 108 124 (1997).
142. "Faster Solution of the Key Equation for Decoding the BCH Error-Correcting Codes", Proc. 29th ACM Symposium on Theory of Computing (STOC'97), 168 175, ACM Press, New York (1997).
143. "Computing Exact Geometric Predicates Using Modular Arithmetic with SinglePrecision" (by H. Br nnimann, I. Z. Emiris, V. Y. Pan, and S. Pion), Proc. 13th Annual ACM Symposium on Computational Geometry, 174 182, ACM Press, New York (1997).
144. "The Structure of Sparse Resultant Matrices" (by I. Z. Emiris and V. Y. Pan),Proc. Annual ACM International Symposium on Symbolic and Algebraic Computations (ISSAC'97), 189 196, ACM Press, New York (1997).
145. "Fast Rectangular Matrix Multiplication and Improving Parallel Matrix Computations" (by X. Huang and V. Y. Pan), Proc. Annual ACM International Symposium on Parallel Algebraic and Symbolic Computation (PASCO'97), 11 23, ACM Press, New York
(1997).
146. "Algebraic and Numerical Techniques for the Computation of Matrix Determinants" (by V. Y. Pan, Y. Yu, and C. Stewart), Computers and Mathematics (with Applications), 34, 1, 43 70 (1997).
147. "E cient Parallel Algorithms for Computing All Pair Shortest Paths in Directed Graphs" (by Y. Han, V.Y. Pan, and J. Reif), Algorithmica, 17, 399 415 (1997).
148. "New Fast Algorithms for Polynomial Interpolation and Evaluation on the Chebyshev Node Set", Computers and Mathematics (with Applications), 35, 3, 125 129 (1998).
149. "Approximate Polynomial Gcds, PadØ Approximation, Polynomial Zeros, and Bipartite Graphs", Proc. 9th Annual ACM-SIAM Symposium on Discrete Algorithms (SODA'98), 68 77, ACM Press, New York, and SIAM Publications, Philadelphia (1998).

150. New Transformations of Cauchy Matrices and Trummer's Problem" (by V. Y. Pan,M. Abu Tabanjeh, Z. Q. Chen, E. Landowne, and A. Sadikou), Computers and Mathematics (with Applications), 35, 12, 1 5 (1998).
151. Asymptotic Acceleration of Solving Polynomial Systems" (by B. Mourrain and V. Y. Pan), Proc. 30th Annual ACM Symposium on Theory of Computing (STOC'98), 488 496, ACM Press, New York (1998).
152. Planar Integer Linear Programming Is NC-equivalent to Euclidean GCD" (by D. F. Shallcross, V. Y. Pan, and Y. Lin-Kriz), SIAM Journal on Computing, 27, 4, 960 971
(1998).
153. Computing Matrix Eigenvalues and Polynomial Zeros Where the Output Is Real" (by D. Bini and V. Y. Pan), SIAM J. on Computing, 27, 4, 1099 1115 (1998).
154. Modular Arithmetic for Linear Algebra Computations in the Real Field" (by I. Z. Emiris, V. Y. Pan, and Y. Yu), J. of Symbolic Computation, 21, 1 17 (1998).
155. Controlled Iterative Methods for Solving Polynomial Systems" (by D. Bondyfalat, B. Mourrain, and V. Y. Pan), Proc. ACM Annual International Symposium on Symbolic and Algebraic Comp. (ISSAC'98), 252 259, ACM Press, New York (1998).
156. Transformations of Cauchy Matrices for Trummer's Problem and a Cauchy like Linear Solver" (by V. Y. Pan, M. Abu Tabanjeh, Z. Chen, S. Providence, and A. Sadikou), Proc. of 5th Annual International Symposium on Solving Irregularly Structured Problems in Parallel (IREGULAR'98), (A. Ferreira, J. Rolim, H. Simon, and S.-H. Teng, editors), Lecture Notes in Computer Science, 1457, 274 284, Springer, Berlin (1998).
157. Fast Rectangular Matrix Multiplication and Applications" (by X. Huang and V. Y. Pan), J. of Complexity, 14, 257 299 (1998).
158. Single Precision Computation of the Sign of Algebraic Predicates", Applied Mathematics Letters, 11, 6, 49 50 (1998).
159. A Uni ed Superfast Algorithm for Boundary Rational Tangential Interpolation Problem and for Inversion and Factorization of Dense Structured Matrices" (by V. Olshevsky and V. Y. Pan), Proc. 39th Annual IEEE Symposium on Foundation of Computer Science (FOCS'98), 192 201, IEEE Computer Society Press, Los Alamitos, California (1998).
160. Sign Determination in Residue Number Systems" (by H. Br nnimann, I. Z. Emiris,V. Y. Pan, and S. Pion), Theoretical Computer Science, 210, 1, 173 197 (1999).
161. Certi ed Computation of the Sign of a Matrix Determinant" (by V. Y. Pan and Y. Yu), Proc. 10th Annual ACM-SIAM Symposium on Discrete Algorithms (SODA'99), 715 724, ACM Press, New York, and SIAM Publications, Philadelphia (1999).
162. Approximate Real Polynomial Division via Approximate Inversion of Real Triangular Toeplitz Matrices" (by V. Y. Pan and Z. Q. Chen), Applied Mathematics Letters, 12, 1 2 (1999).
163. Parallel Matrix Multiplication on a Linear Array with a Recon gurable Pipelined Bus System" (by K. Li and V. Y. Pan), Proc. 13th International Parallel Processing Symposium and 10th Symposium on Parallel and Distributed Computing (IPPS/SPDP 1999), 31 35, IEEE Computer Soc. Press, Los Alamitos, California (April 1999).
164. The Complexity of the Matrix Eigenproblem" (by V. Y. Pan and Z. Chen), Proc. 31st Annual ACM Symposium on Theory of Computing (STOC'99), 507 516, ACM Press, New York (1999).
165. Superfast Computations with Singular Structured Matrices over Abstract Fields" (by V. Y. Pan, A. Zheng, M. Abu Tabanjeh, Z. Chen, and S. Providence), Proc. Second Workshop on Computer Algebra in Scienti c Computing (CASC'99), (V. G. Ganzha, E. W. Mayr, E. V. Vorozhtsov, Editors), 323 338, Springer (May 1999).

166. Polynomial and Rational Interpolation and Multipoint Evaluation (with StructuredMatrices)" (by V. Olshevsky and V. Y. Pan), Proc. 26th International Colloquium on Automata, Languages and Programming (ICALP'99), 1644, 585 594, Springer's Lecture Notes in Computer Science, Springer, Berlin (July 1999).
167. A New Approach to Bisection Acceleration for the Symmetric Tridiagonal Eigenvalue Problem" (by V. Y. Pan and E. Linzer), Numerical Algorithms, 22, 13 39 (1999).
168. Nearly Optimal Computations with Structured Matrices", Proc. of 11th Annual ACM-SIAM Symposium on Discrete Algorithms (SODA'2000), 953 962, ACM Press, New York, and SIAM Publications, Philadelphia (January 2000).
169. Approximating Complex Polynomial Zeros: Modi ed Quadtree (Weyl's) Construction and Improved Newton's Iteration", J. of Complexity, 16, 1, 213 264 (2000).
170. Multivariate Polynomials, Duality and Structured Matrices" (by B. Mourrain and V. Y. Pan), J. of Complexity, 16, 1, 110 180 (2000).
171. Lifting/Descending Processes for Polynomial Zeros and Applications" (by B. Mourrain and V. Y. Pan), J. of Complexity, 16, 1, 265 273 (2000).
172. New Techniques for the Computation of Linear Recurrence Coe cients", Finite Fields and Their Applications, 6, 93 118 (2000). Proc, version in ACM STOC 1997.
173. Superfast Algorithms for Cauchy like Matrix Computations and Extensions" (by V. Y. Pan and A. Zheng), Linear Algebra and Its Applications, 310, 83 108 (2000).
174. Matrix Structure, Polynomial Arithmetic, and Erasure-Resilient Encoding and Decoding", Proceedings of International Symposium on Symbolic and Algebraic Computations (ISSAC'2000), 266 271, ACM Press, New York (2000).
175. Parallel Complexity of Computations with General and Toeplitz like Matrices Filled with Integers", SIAM J. on Computing, 30, 1080 1125 (2000).
176. Computation of a Speci ed Root of a Polynomial System of Equations Using Eigenvectors" (by D. Bondyfalat, B. Mourrain, and V. Y. Pan), Linear Algebra and Its Applications, 319, 193 209 (2000).
177. A New Proximity Test for Polynomial Zeros", Computers and Mathematics (with Applications), 41, 12, 1559 1560 (2001).
178. A Homotopic Residual Correction Process", Proc. of the Second Conference on Numerical Analysis and Applications (L. Vulkov, J. Wasniewsky, and P. Yalamov, editors), Lecture Notes in Computer Science, 1988, 644 649, Springer, Berlin (2001).
179. Newton's Iteration for the Inversion of Structured Matrices" (by V. Y. Pan and Y. Rami), in "Structured Matrices: Recent Developments in Theory and Computation", 79 90 (edited by D. Bini, E. Tyrtyshnikov, and P. Yalamov), Nova Science Publishers, USA (2001).
180. Certi cation of Numerical Computation of the Sign of the Determinant of a Matrix" (by V. Y. Pan and Y. Yu), Algorithmica, 30, 708 724 (2001).
181. Parallel Matrix Multiplication on a Linear Array with a Recon gurable Pipelined Bus System" (by K. Li and V. Y. Pan), IEEE Transactions on Computing, 50, 5, 519 525
(2001).
182. Numerical Computation of a Polynomial GCD and Extensions", Information and Computation, 167, 2, 71 85 (2001).
183. Univariate Polynomials: Nearly Optimal Algorithms for Factorization and Root nding", Proc. of International Symposium on Symbolic and Algebraic Computation (ISSAC'01), 253 267, ACM Press, NY (2001).

184. Nearly Optimal Algorithms for Univariate Polynomial Factorization and Root nding II: Computing a Basic Annulus for Splitting", Trudy of the Steklov Mathematical Institute of the Russian Academy of Science (in English), volume 235, pp. 211 223, Moscow
(2001).
185. Asymptotic Acceleration of the Solution of Multivariate Polynomial Systems of Equations" (by B. Mourrain, V. Y. Pan, and O. Ruatta), Proceedings of the Smalefest 2000 (F. Cucker and M. Rojas, editors), Foundations of Computational Mathematics Series, 267 294, World Scienti c, New Jersey (2002).
186. Nearly Optimal Algorithms for Numerical Univariate Polynomial Factorization and Root nding I: Splitting a Polynomial into Factors over an Annulus", Proceedings of the Smalefest 2000 (F. Cucker and M. Rojas, editors), Foundations of Computational Mathematics Series, 325 353, World Scienti c, New Jersey (2002).
187. Structured Matrices and Newton's Iteration: Uni ed Approach" (by V. Y. Pan, Y. Rami, and X. Wang), Linear Algebra and Its Applications, 343/344, 233 265 (2002).
188. Randomized Acceleration of Fundamental Matrix Computations", Proc. of Symposium on Theoretical Aspects of Computer Science (STACS '02), Lecture Notes in Computer Science, 2285, 215 226, Springer, Berlin (2002).
189. Symbolic and Numerical Methods for Exploiting Structure in Constructing Resultant Matrices" (by I. Z. Emiris and V. Y. Pan), Journal of Symbolic Computation, 33, 393 413 (2002).
190. Acceleration of Euclidean Algorithm and Extensions" (by V. Y. Pan and X. Wang),Proc. International Symposium on Symbolic and Algebraic Computation (ISSAC '02), 207 213, ACM Press, New York (2002).
191. Univariate Polynomials: Nearly Optimal Algorithms for Numerical Factorization and Root Finding", J. of Symbolic Computation, 33, 5, 701 733 (2002).
192. Can We Optimize Toeplitz/Hankel Computations?" in Proc. the 5th InternationalWorkshop on Computer Algebra in Scienti c Computing (CASC'02), (edited by E. W. Mayr, V. G. Ganzha, and E. V. Vorozhtzov), 253 264, Technische University of M nchen, Germany
(2002).
193. Accelerated Solution of Multivariate Polynomial Systems of Equations" (by B.
Mourrain, V. Y. Pan, and O. Ruatta), SIAM J. on Computing, 32, 2, 435 454 (2003).
194. Inversion of Displacement Operators" (by V. Y. Pan and X. Wang), SIAM J. on Matrix Analysis and Applications, 24, 3, 660 677 (2003).
195. Matrix Structure and Loss-Resilient Encoding/Decoding", Computers and Mathematics (with Applications), 46, 493 499 (2003).
196. Acceleration of Euclidean Algorithm and Rational Number Reconstruction" (by X. Wang and V. Y. Pan), SIAM J. of Computing, 32, 2, 548 556 (2003).
197. Improved Computation of Determinants and Resultants" (by I. Z. Emiris and V. Y. Pan), Proc. of the 6th International Workshop on Computer Algebra in Scienti c Computing (CASC '03), (edited by E.W. Mayr, V.G. Ganzha, and E.V. Vorozhtzov), 81 94, Technische University of M nchen (2003).
198. Inverse Power and Durand-Kerner Iteration for Univariate Polynomial Root Finding" (by D. A. Bini, L. Gemignani and V. Y. Pan), Computers and Mathematics (with Applications), 47, 2/3, 447 459 (2004).
199. On Rational Number Reconstruction and Approximation" (by V. Y. Pan and X. Wang), SIAM J. on Computing, 33, 2, 502 503 (2004).
200. Iterative Inversion of Structured Matrices" (by V. Y. Pan, M. Van Barel, X. Wangand G. Codevico), Theoretical Computer Science, 315, 2 3, 581 592 (2004).

201. The Least-Squares Compression Policy for Newton like Iteration for Structured Matrices" (by G. Codevico, V. Y. Pan, M. Van Barel, X. Wang, and A. L. Zheng) in Proceedings of the 6th International Mathematica Symposium (IMS 2004), (edited by C.
Jacob and P. Mitic), Ban , Alberta, Canada (August 2004).
202. Improved Initialization of the Accelerated and Robust QR like Polynomial Root nding" (by D. A. Bini, L. Gemignani, and V. Y. Pan), Electronic Transactions on Numerical Analysis, 17, 195 205 (2004). (Proc. version in Proceedings of the 7th International Workshop on Computer Algebra in Scienti c Computing (CASC'04), St. Petersburg, Russia (2004), (edited by E. W. Mayr, V. G. Ganzha, and E. V. Vorozhtzov), 39-50, Technische Univ. M nchen, Germany (2004).)
203. An E cient Solution for Cauchy like Systems of Linear Equations" (by Z. Chen and V. Y. Pan), Computers and Mathematics with Applications, 48, 529 537 (2004).
204. Newton like Iteration Based on Cubic Polynomials for Structured Matrices" (by G. Codevico, V. Y. Pan, and M. Van Barel), Numerical Algorithms, 36, 365 380 (2004).
205. On Theoretical and Practical Acceleration of Randomized Computation of the Determinant of an Integer Matrix", Zapiski Nauchnykh Seminarov POMI (in English), Vol. 316, 163 187, St. Petersburg, Russia (2004).
206. Can the TPR1 Structure Help Us to Solve the Algebraic Eigenproblem?", in Proc.of the 16th Annual ACM-SIAM Symposium on Discrete Algorithms (SODA'05), 1069 1078, ACM Press, New York, and SIAM Publications, Philadelphia (2005).
207. Fast and Stable QR Eigenvalue Algorithms for Generalized Semiseparable Matrices and Secular Equation" (by D. A. Bini, L. Gemignani, and V. Y. Pan), Numerische Mathematik, 3, 373 408 (2005).
208. Improved Algorithms for Computing Determinants and Resultants" (by I. Z. Emirisand V. Y. Pan), J. of Complexity, 21, 1, 43 71 (2005).
209. Coe cient free Adaptation of Polynomial Root nders", Computers and Mathematics with Applications, 50, 263 269 (2005).
210. Amended DSeSC Power Method for Polynomial Root nding", Computers and Mathematics with Applications, 49, 9 10, 1515 1524 (2005).
211. Homotopic Residual Correction Algorithms for General and Structures Matrices" (by V. Y. Pan, M. Kunin, R. Rosholt, and H. Kodal), Mathematics of Computation, 75, 345 368 (2006).
212. Linking the TPR1, DPR1 and Arrow-head Matrix Structures" (by V. Y. Pan, B. Murphy, R. E. Rosholt, Y. Tang, X. Yan, and W. Cao), Computers and Mathematics with Applications, 52, 10-11, 1603 1608 (2006).
213. Real Root nding" (by V. Y. Pan, G. Qian, B. Murphy, R. E. Rosholt, and Y. Tang), Proceedings of the Third International Workshop on Symbolic Numeric Computation (SNC 2007), July 2007, London, Ontario, Canada (Jan Verschelde and Stephen Watt, editors), 161 169, ACM Press, New York (2007).
214. Null Space and Eigenspace Computation with Additive Preconditioning" (by V. Y.Pan and X. Yan), Proceedings of the Third International Workshop on Symbolic Numeric Computation (SNC'2007), July 2007, London, Ontario, Canada (Jan Verschelde and Stephen Watt, editors), 170 179, ACM Press, New York (2007).
215. The Schur Aggregation for Solving Linear Systems of Equations" (by V. Y. Pan, B. Murphy, R. E. Rosholt, and M. Abu Tabanjeh), Proceedings of the Third International Workshop on Symbolic Numeric Computation (SNC'2007), July 2007, London, Ontario, Canada (Jan Verschelde and Stephen Watt, editors), 180 188, ACM Press, New York (2007).
216. New Homotopic/Factorization and Symmetrization Techniques for Newton's and Newton's Structured Iteration", Computers Mathematics with Applications, 54, 721 729 (2007).

217. Additive Preconditioning and Aggregation in Matrix Computations" (by V. Y. Pan, D. Ivolgin, B. Murphy, R. E. Rosholt, I. Taj Eddin, Y. Tang, and X. Yan), Computers and Mathematics with Applications, 55, 8, 1870 1886 (2008).
218. Eigen-solving via Reduction to DPR1 Matrices" (by V. Y. Pan, B. Murphy, R. Rosholt, Y. Tang, X. Wang, and A. Zheng), Computers and Mathematics with Applications, 56, 166 171 (2008).
219. Additive Preconditioning for Matrix Computations" (by V. Y. Pan, D. Ivolgin, B. Murphy, R. E. Rosholt, Y. Tang, and X. Yan), in Proc. of the Third International Computer Science Symposium in Russia (CSR'2008), Lecture Notes in Computer Science (LNCS), 5010, 372 383 (2008).
220. Degeneration of Integer Matrices Modulo an Integer" (by V. Y. Pan and X. Wang),Linear Algebra and Its Applications, 429, 2113 2130 (2008).
221. Schur Aggregation for Linear Systems and Determinants" (by V. Y. Pan, D. Grady,B. Murphy, G. Qian, R. E. Rosholt, and A. Ruslanov), Theoretical Computer Science, Special Issue on Symbolic-Numerical Algorithms (D.A. Bini, V. Y. Pan, and J. Verschelde, editors), 409, pp. 255 268 (2008).
222. Fast Rectangular Matrix Multiplication and Some Applications" (by S. Ke, B. Zeng, W. Han, and V. Y. Pan), Science in China, Series A: Mathematics, 51, 3, 389 406,
2008.
223. Additive Preconditioning, Eigenspaces, and the Inverse Iteration" (by V. Y. Pan and X. Yan), Linear Algebra and Its Applications, 430, 186 203 (2009).
224. A New Error free Floating-Point Summation Algorithm" (by V. Y. Pan, B. Murphy, G. Qian, and R. E. Rosholt), Computers and Mathematics with Applications, 57, 560 564 (2009).
225. Preconditioning, Randomization, Solving Linear Systems, Eigen-Solving, and Root Finding" (by V. Y. Pan, G. Qian, and A. L. Zheng), Proc. International Symposium on Symbolic- Numerical Computations (SNC'2009), Kyoto, Japan, August 2009, (edited by Hiroshi Kai and Hiroshi Sekigawa), pp. 5 6, ACM Press, New York(2009).
226. Nearly Optimal Symbolic-Numerical Algorithms for Structured Integer Matrices and Polynomials" (by V. Y. Pan, B. Murphy, and R. E. Rosholt), Proc. of International Symposium on Symbolic-Numerical Computation (SNC'2009), Kyoto, Japan, August 2009, (edited by Hiroshi Kai and Hiroshi Sekigawa), 105 113, ACM Press, New York (2009).
227. Root Squaring with DPR1 Matrices", in Zapiski Nauchnykh Seminarov POMI (in English), volume 373 (edited by N. N. Vasiliev and A. M. Vershik), pp. 189 193 (2009).
228. Solving Homogeneous Linear Systems with Randomized Preprocessing" (by V. Y. Pan and G. Qian), Linear Algebra and Its Applications, 432, 3272 3318 (2010).
229. Uni ed Nearly Optimal Algorithms for Structured Matrices" (by V. Y. Pan, B. Murphy, and R. E. Rosholt), Operator Theory: Advances and Applications, 199, 359 375, Birkha ser, Basel (2010).
230. Additive Preconditioning for Matrix Computations" (by V. Y. Pan, D. Ivolgin, B. Murphy, R. E. Rosholt, Y. Tang, and X. Yan), Linear Algebra and Its Applications, 432, 1070 1089 (2010).
231. Newton's Iteration for Matrix Inversion, Advances and Extensions", pp. 364 381, in Matrix Methods: Theory, Algorithms and Applications (dedicated to the Memory of Gene Golub, edited by Vadim Olshevsky and Eugene Tyrtyshnikov), World Scienti c Publishing, New Jersey, ISBN-13 978-981-283-601-4, ISBN-10-981-283-601-2 (2010).
232. Advancing Matrix Computations with Randomized Preprocessing" (by V. Y. Pan,G. Qian, and A. L. Zheng), in Proc. of the Fifth International Computer Science Symposium in Russia (CSR'2010), Kazan, Russia, June 2010 (Farid Ablaev and Ernst W. Mayr, editors), Lecture Notes in Computer Science (LNCS), pages 303 314, Springer, Berlin (2010).

233. Real and Complex Polynomial Root Finding with Eigen Solving and Preprocessing" (by V. Y. Pan and A. L. Zheng), (edited by Stephen Watt), in Proc. International Symposium on Symbolic and Algebraic Computation (ISSAC'2010), pages 219 226, ACM Press, New York (July 2010).

234. Matrix Computations and Polynomial Root nding with Preprocessing" (by V. Y. Pan, G. Qian, A. L. Zheng, and Z. Chen), Linear Algebra and Its Applications, 434, 854 879 (2011).

235. New Progress in Real and Complex Polynomial Root Finding" (by V. Y. Pan and A. L. Zheng), Computers and Mathematics (with Applications), 61, 1305 1334 (2011).

236. Univariate Polynomial Root Finding by Arming with Constraints", in Proc. International Symposium on Symbolic-Numerical Computations (SNC'2011), San Jose, California, 2011 (edited by Marc Moreno Masa), 112 121, ACM Press, New York (2011).

237. Randomized Preconditioning of the MBA Algorithm" (by V. Y. Pan, G. Qian, andA. L. Zheng), in Proc. International Symposium on Symbolic and Algebraic Computation (ISSAC'2011), San Jose, California, June 2011 (edited by Anton Leykin), 281 288, ACM Press, New York (2011).

238. Nearly Optimal Solution of Rational Linear Systems of Equations with Symbolic Lifting and Numerical Initialization", Computers and Mathematics with Applications, 62, 1685 1706 (2011).

239. Root nding by Expansion with Independent Constraints" (by V. Y. Pan and A. L. Zheng), Computers and Mathematics with Applications, 62, 3164 3182 (2011).

240. A Note on the Paper by Murat Cenk and Ferruh Ozbudak Multiplication of Polynomials Modulo x^n", Theoretical Computer Science, 412 (2011) 3451 3462", T, Theoretical Computer Science, 428, page 91 (2012).

241. E cient Polynomial Root re ners: A Survey and New Record E ciency Estimate" (by J. M. McNamee and V. Y. Pan), Computers and Mathematics with Applications, 63, 239 254 (2012).

242. Root re ning for a Polynomial Equation", Proceedings of the 14th International Workshop on Computer Algebra in Scienti c Computing (CASC'2012), (V. P. Gerdt, V. Koepf, E. W. Mayr, and E. V. Vorozhtsov, editors), Lecture Notes in Computer Science, 7442, 271 282, Springer, Heidelberg (2012).

243. Real and Complex Polynomial Root nding via Eigen solving and Randomization" (by V. Y. Pan, G. Qian, and A. L. Zheng), Proceedings of the 14th International Workshop on Computer Algebra in Scienti c Computing (CASC'2012), (V. P. Gerdt, V. Koepf, E. W. Mayr, and E. V. Vorozhtsov, editors), Lecture Notes in Computer Science, 7442, 283 293, Springer, Heidelberg (2012).

244. Solving Linear Systems of Equations with Randomization, Augmentation and Aggregation" (by V. Y. Pan and G. Qian), Linear Algebra and Its Applications, 437, 2851 2876 (2012).

245. Randomized Preconditioning versus Pivoting" (by V. Y. Pan, G. Qian, and A. L. Zheng), Linear Algebra and Its Applications, 438, 4, 1883 1889 (2013).

246. On the Boolean Complexity of the Real Root Re nement" (by V. Y. Pan and E. P. Tsigaridas), in Proc. International Symposium on Symbolic and Algebraic Computations (ISSAC'2013), Boston, Massachusetts, June 2013 (M. Kauers, editor), 299 306, ACM Press, New York (2013).

247. Polynomial Evaluation and Interpolation and Transformations of Matrix Structures", Proceedings of the 15th International Workshop on Computer Algebra in Scienti c Computing (CASC'2013), (V. P. Gerdt, V. Koepf, E. W. Mayr, and E. V. Vorozhtsov, editors), Lecture Notes in Computer Science, 8136, 273 287, Springer, Heidelberg (2013).

248. Fast Approximate Computations with Cauchy Matrices, Polynomials and Rational Functions", Proc. of the Ninth International Computer Science Symposium in Russia (CSR'2014), (E. A. Hirsch et al., editors), Moscow, Russia, June 2014, Lecture Notes in Computer Science (LNCS), 8476, pp. 287 300, Springer International Publishing, Switzerland (2014).

249. Nearly Optimal Computations with Structured Matrices" (by Victor Y. Pan and Elias Tsigaridas), Proc. of the International Conference on Symbolic Numeric Computation (SNC 2014), (edited by S. Watt and J. Verschelde), 21 30, ACM Press, New York, 2014.
250. Accelerated Approximation of the Complex Roots of a Univariate Polynomial" (by Victor Y. Pan and Elias Tsigaridas), Proc. of the International Conference on Symbolic Numeric Computation (SNC 2014), (edited by S. Watt and J. Verschelde), pages 132 134, ACM Press, New York, 2014.
251. Global Newton Iteration over Archimedean and non-Archimedean Fields" (by Jonathan D. Hauenstein, Victor Pan, and Agnes Szanto), Proceedings of the 16th International Workshop on Computer Algebra in Scienti c Computing (CASC'2014), (V.P. Gerdt, V. Koepf, W.M. Zeiler, and E.V. Vorozhtsov, editors), Lecture Notes in Computer Science, 8660, 202 217, Springer, Cham (2014). Also April 17, 2014, arXiv:1404.5525 [math.NA]
252. Real Polynomial Root nding by Means of Matrix and Polynomial Iterations", Proceedings of the 16th International Workshop on Computer Algebra in Scienti c Computing (CASC'2014), (V.P. Gerdt, V. Koepf, W.M. Zeiler, and E.V. Vorozhtsov, editors), Lecture Notes in Computer Science, 8660, 335 349, Springer, Cham (2014).
253. Transformations of Matrix Structures Work Again", Linear Algebra and Its Applications, 465, 1 32 (2015).
254. Estimating the Norms of Circulant and Toeplitz Random Matrices and Their Inverses" (by V. Y. Pan, J. Svadlenka, and L. Zhao), Linear Algebra and Its Applications, 468, 197 210 (2015)
255. Random Multipliers Numerically Stabilize Gaussian and Block Gaussian Elimination: Proofs and an Extension to Low-rank Approximation" (by V. Y. Pan, G. Qian, and X. Yan), Linear Algebra and Its Applications, 481, 202 234 (2015).
256. Real Root Isolation by Means of Root Radii Approximation" (by V. Y. Pan and L. Zhao), Proceedings of the 17th International Workshop on Computer Algebra in Scienti c Computing (CASC'2015), (V. P. Gerdt, V. Koepf, and E. V. Vorozhtsov, editors), Lecture Notes in Computer Science, 9301, 347 358, Springer, Heidelberg (2015). Also arXiv:1501.05386 [math.NA] (36 pages, 12 tables), 22 January 2015, revised October 21, 2015.
257. Randomized Circulant and Gaussian Preprocessing" (by V. Y. Pan and L. Zhao), Proceedings of the 17th International Workshop on Computer Algebra in Scienti c Computing (CASC'2015), (V. P. Gerdt, V. Koepf, and E. V. Vorozhtsov, editors), Lecture Notes in Computer Science, 9301, 359 373, Springer, Heidelberg (2015).
258. Nearly Optimal Re nement of Real Roots of a Univariate Polynomial" (by V. Y. Pan and E. P. Tsigaridas), J. of Symbolic Computation, 74, 181 204 (2016)
259. How Bad Are Vandermonde Matrices?", SIAM Journal of Matrix Analysis and Applications, 37, 2, 676 694 (2016).
260. Low-rank Approximation of a Matrix: Novel Insights, New Progress, and Extensions" (by Victor Y. Pan and Liang Zhao), Proc. of the Eleventh International Computer Science Symposium in Russia (CSR'2016), (Alexander Kulikov and Gerhard Woeginger, editors), St. Petersburg, Russia, June 2016, Lecture Notes in Computer Science (LNCS), Volume 9691, 352 366, Springer International Publishing, Switzerland (2016).
261. Real Polynomial Root nding by Means of Matrix and Polynomial Iterations" (by Victor Y. Pan and Liang Zhao), Theoretical Computer Science, Special Issue on Symbolic Numerical Algorithms (Stephen Watt, Jan Verschelde, and Lihong Zhi, editors), 681, 101-116 (2017), http://dx.doi.org/10.1016/j.tcs.2017.03.032.

262. Accelerated Approximation of the Complex Roots and Factors of a Univariate Polynomial" (by V. Y. Pan and E. P. Tsigaridas), Theoretical Computer Science, Special Issue on Symbolic Numerical Algorithms (Stephen Watt, Jan Verschelde, and Lihong Zhi, editors), 681, 138 145 (2017). http://dx.doi.org/10.1016/j.tcs.2017.03.030.
263. Nearly Optimal Computations with Structured Matrices" (by V. Y. Pan and E. P. Tsigaridas), Theoretical Computer Science, Special Issue on Symbolic Numerical Algorithms (Stephen Watt, Jan Verschelde, and Lihong Zhi, editors)), 681, 117 137, 2017. http://dx.doi.org/10.1016/j.tcs.2017.03.031.
264. Simple and Nearly Optimal Polynomial Root Finding by Means of Root Radii Approximation", Springer International Publishing (I.S. Kotsireas and E. Martinez- Moro, editors), Applications of Computer Algebra, Springer, Chapter 23, Proceedings in Mathematics and Statistics, 198 AG (2017). Chapter DOI.10.1007/978-3-319-56932-1_23
265. New Studies of Randomized Augmentation and Additive Preprocessing" (by VictorY. Pan and Liang Zhao), Linear Algebra and Its Applications, 512, 256 305 (2017). Available at http://dx.doi.org/10.1016/j.laa.2016.09.035.
266. "Numerically Safe Gaussian Elimination with No Pivoting" (by Victor Y. Pan andLiang Zhao), Linear Algebra and Its Applications, 527, 349 383 (2017).
http://dx.doi.org/10.1016/j.laa.2017.04.007
267. Fast Approximate Computations with Cauchy Matrices and Polynomials", Mathematics of Computation, 86, 2799 2826, 2017. DOI: https://doi.org/10.1090/mcom/3204
268. An E cient Computation of Generalized Inverse of a Matrix" (by V.Y. Pan, F.
Soleymani and L. Zhao), Applied Mathematics and Computation, 316, 89 101, 2018. http://dx.doi.org/10.1016/j.amc.2017.08.010
269. Implementation of a Near-Optimal Complex Root Clustering Algorithm" (by R. Imbach, V.Y. Pan, C. Yap), Proc. of International Congress on Math Software (ICMS 2018), 235 244, 2018.
270. Old and New Nearly Optimal Polynomial Root-Finders", In: Proceedings of the 21st International Workshop on Computer Algebra in Scienti c Computing (CASC'2019), (M. England, W. Koepf, T.M. Sadikov, W.M. Seiler, and E. V. Vorozhtsov, editors), Lecture Notes in Computer Science, 11661, 393 411, Springer, Nature Switzerland (2019), doi: 10.1007/978-3-030-26831-2 and arXiv:1805.12042 Submitted on 30 May 2018
271. Root-Finding with Implicit De ation" (with R. Imbach, C., Yap, I.S. Kotsireas, V. Zaderman), In: Proceedings of the 21st International Workshop on Computer Algebra in Scienti c Computing (CASC'2019), (M. England, W. Koepf, T.M. Sadikov, W.M. Seiler, and E. V. Vorozhtsov, editors), Lecture Notes in Computer Science, 11661, 236 245, Springer, Nature Switzerland (2019), doi: 10.1007/978-3-030-26831-2 and arxiv:1606.01396, submitted on 21 May (2019)
272. Sublinear Cost Low Rank Approximation via Subspace Sampling" (by V. Y. Pan, Q. Luan, J. Svadlenka, and L. Zhao), In LNCS 11989, Book: Mathematical Aspects of Computer and Information Sciences (MACIS 2019), D. Salmanig et al (Eds.), Springer Nature Switzerland AG 2020, Chapter No: 9, pages 1 16, Springer Nature Switzerland AG 2020 Chapter DOI:10.1007/978-3-030-43120-4_9
273. CUR LRA at Sublinear Cost Based on Volume Maximization" (by Q. Luan, V. Y. Pan), LNCS 11989, In Book: Mathematical Aspects of Computer and Information Sciences (MACIS 2019), D. Salmanig et al (Eds.), Springer Nature Switzerland AG 2020, Chapter No: 10, pages 1 17, Chapter DOI:10.1007/978-3-030-43120-4_10
274. New Practical Advances in Polynomial Root Clustering" (by R. Imbach, V. Y. Pan), LNCS 11989, In Book: Mathematical Aspects of Computer and Information Sciences (MACIS 2019), D. Salmanig et al (Eds.),

Springer Nature Switzerland AG 2020, Chapter No: 11, pages 1 16, Chapter DOI:10.1007/978-3-030-43120-4_11

275. New Progress in Univariate Polynomial Root- nding" (by R. Imbach, V. Y. Pan), Proceedings of ACM-SIGSAM ISSAC 2020, ACM ISBN 978-1-4503-7100-1/20/07 https://doi.org/10.1145/3373207.3403979

276. Acceleration of Subdivision Root-Finding for Sparse Polynomials", In: Computer Algebra in Scienti c Computing (CASC'20), Springer Nature Switzerland AG 2020, F. Boulier et al. (Eds.): CASC 2020, LNCS 12291, Ch. 27, pp. 1 17, 2020 https://doi.org/10.1007/978 − 3 − 030 − 60026 − 6_27

277. Faster Numerical Univariate Polynomial Root-Finding by Means of Subdivision Iterations" (by Qi Luan, Victor Y. Pan, Wongeun Kim, and Vitaly Zaderman), In: Computer Algebra in Scienti c Computing (CASC'20), Springer Nature Switzerland AG 2020,
F. Boulier et al. (Eds.): CASC 2020, LNCS 12291, Ch. 25, pp. 1 16, 2020 https://doi.org/10.1007/978 − 3 − 030 − 60026 − 6_25

278. Root Radii and Subdivision for Polynomial Root-Finding" (by R. Imbach, V. Y. Pan), In: Computer Algebra in Scienti c Computing (CASC'21), Springer Nature Switzerland AG 2021, F. Boulier et al. (Eds.): CASC 2021, LNCS 12865, pp. 1 21, 2021. *https* : *//doi.org/*10.1007/978 − 3 − 030 − 85165 − 1_9

279. New Progress in Sparse Polynomial Root- nding" (by V.Y. Pan), In: Sirius University of Science and Technology, Sirius Mathematics Center, International Conference:
Computer Algebra in Scienti c Computing, September 13 17, 2021, pages 85 91, Sirius Federal Territory, 2021.

280. Accelerated subdivision for clustering roots of polynomials given by evaluation oracles" (by R. Imbach, V. Y. Pan), In: Computer Algebra in Scienti c Computing (CASC'22), Springer Nature Switzerland AG 2022, F. Boulier et al. (Eds.): CASC 2022, LNCS 13366, pp. 143 164, 2022. arXiv preprint 2206.08622 (2022)

281. Fast approximation of polynomial zeros and matrix eigenvalues" (by Pan, V.Y., Go, S., Luan, Q., Zhao, L.), accepted by 13th International Symposium on Algorithms and Complexity (CIAC 2023), Springer's Lecture Notes in Computer Science, Springer (2023)

11 TALKS AND PRESENTATIONS AT PROFESSIONAL MEETINGS SINCE 1991

1991

2nd Annual ACM-SIAM Symposium on Discrete Algorithms (SODA'91), San Francisco, California, January 1991. Refereed paper was accepted by Program Committee.

Fifth Biennial Copper Mountain Conference on Multigrid Methods, Copper Mountain, Colorado, April 1991. Refereed paper was accepted by Program Committee.

18th International Colloquium on Automata, Languages and Programming (ICALP'91), Madrid, Spain, July 1991. Refereed paper was accepted by Program Committee.

3rd Annual ACM Symposium on Parallel Algorithms and Architectures (SPAA'91), Hilton Head, South Carolina, July 1991. Refereed paper was accepted by Program Committee.

4th SIAM Conference on Applied Linear Algebra, Minneapolis, Minnesota, September 1991. Two talks at mini-symposia.

3rd IEEE Symposium on Parallel and Distributed Algorithms, Dallas, Texas, December 1991. Refereed paper was accepted by Program Committee.

1992

3rd Annual ACM-SIAM Symposium on Discrete Algorithms, Orlando, Florida, January 1992. Refereed paper was accepted by Program Committee.
Israel Symposium on the Theory of Computing and Systems (ISTCS'92), Haifa, Israel, May 1992. Refereed paper was accepted by Program Committee.
4th Annual ACM Symposium on Parallel Algorithms and Architectures (SPAA'92), San Diego, California, June-July 1992. Two refereed papers were accepted by Program Committee.
33rd Annual IEEE Conference on Foundations of Computer Science (FOCS'92), Pittsburgh, Pennsylvania, October 1992. Three refereed papers were accepted by Program Committee.
Second Biennial Copper Mountain Conference on Iterative Methods, Copper Mountain, Colorado, April 1992. Refereed paper was accepted by Program Committee.

1993

Panamerican Workshop for Applied and Computational Mathematics, Caracas, Venezuela, January 1993. Three refereed papers were accepted by Program Committee.
Workshop on Applicable Algebra, Obervolfach, Germany, February 1992. Invited talk (30 minutes).
Annual ACM International Symposium on Symbolic and Algebraic Computations (ISSAC'93), Kiev, Ukraine, July 1993. Refereed paper was accepted by Program Committee.
3rd SIAM Conference on Linear Algebra, Seattle, Washington, August 1993. Invited talk at mini-symposium (30 minutes).
884th Meeting of the American Mathematical Society, Syracuse, New York, September 1993. Invited talk (30 minutes).
34th Annual IEEE Conference on Foundations of Computer Science, Palo Alto, California, November 1993. Refereed paper was accepted by Program Committee.
Workshop on Parallel Algorithms, DIMACS, Rutgers University, New Jersey, November 1993. Invited talk (30 minutes).

1994

5th Annual ACM-SIAM Symposium on Discrete Algorithms (SODA'94). Two refereed papers were accepted by Program Committee.
Third Biennial Colorado Conference on Iterative Methods (CCIM'94), Breckenridge, Colorado, April 1994. Refereed paper was accepted by Program Committee.
5th SIAM Conference on Applied Linear Algebra, Snowbird, Utah, June 1994. Three refereed papers were accepted by Program Committee.
First Intern. Symposium on Parallel Algebraic and Symbolic Computation (PASCO'94), Linz, Austria, September 1994. Refereed paper was accepted by Program Committee.
35th Annual IEEE Conference on Foundation of Computer Science (FOCS'94), Santa Fe, New Mexico, November 1994. Refereed paper was accepted by Program Committee.

1995

Annual ACM Symposium on Theory of Computing (STOC'95), Las Vegas, Arizona, May 1995. Refereed paper was accepted by Program Committee.
25th AMS-SIAM Summer Seminar on Mathematics of Numerical Analysis, Park City, Utah, July-August 1995. Invited plenary talk (1 hour).
Seminar on Real Computation and Complexity, Schloss Dagstuhl, Germany, November 1995. Invited talk (45 minutes).

1996

7th Annual ACM-SIAM Symposium on Discrete Algorithms (SODA'96), Atlanta, Georgia, January 1996. Refereed paper was accepted by Program Committee.

Fourth Biennial Copper Mountain Conference on Iterative Methods, Copper Mountain, Colorado, April 1996. Refereed paper was accepted by Program Committee.

NATO Advanced Study Workshop on Algorithms for Sparse Large Scale Linear Systems, Las Palmas de Grand Canaria, Spain, June 1996. Invited Talk (1 hour).

Workshop on Symbolic - Numeric Algebra for Polynomials (SNAP'96), INRIA Sophia Antipolis, France, July 1996. Invited talk (45 minutes).

International Conference on Structured Matrices, Cortona, Italy, September 1996. Two refereed papers were accepted by Program Committee.

1997

International Conference on Foundation of Computational Mathematics (FoCM), Rio de Janeiro, Brazil, January 1997. Invited semi-plenary talk (50 minutes) and invited talk (30 minutes).

FRISCO Open Workshop 97, INRIA Sophia Antipoles, France, March 1997. Invited talk (20 minutes).

The 29th Annual ACM Symposium of Theory of Computing (STOC'97), El Paso, Texas, May 1997. Refereed paper was accepted by Program Committee.

The 13th Annual ACM Symposium on Computational Geometry, Nice, France, June 1997. Refereed paper was accepted by Program Committee.

Faddeev Memorial International Algebraic Conference, St. Petersburg, Russia, June 1997. Invited talk (45 minutes).

Annual ACM International Symposium on Symbolic and Algebraic Computation (ISSAC'97), Maui, Hawaii, August 1997. Refereed paper was accepted by Program Committee.

Second ACM International Symposium on Parallel Symbolic Computation (PASCQ'97), Maui, Hawaii, August 1997. Refereed paper was accepted by Program Committee.

1998

9th Annual ACM-SIAM Symposium on Discrete Algorithms (SODA'98), January 1998, San Francisco, California. Refereed paper was accepted by Program Committee.

Fifth Biennial Copper Mountain Conference on Iterative Methods, March 1998. Copper Mountain, Colorado. Refereed paper was accepted by Program Committee.

933rd AMS Meeting, April 1998, Philadelphia. Invited talk at mini-symposium.

30th Annual ACM Symposium on Theory of Computing (STOC'98), May 1998, Dallas, Texas. Refereed paper was accepted by Program Committee.

Kurosh Memorial Algebraic Conference, June 1998, Moscow, Russia. Invited talk at mini-symposium.

International Seminar on Real Computation and Complexity, June 1998, Dagstuhl, Germany. Invited talk (45 minutes).

SIAM Annual Meeting, July 1998, Toronto, Canada. Invited talk at mini-symposium.

Annual International Conference IMACS on Application of Computer Algebra (ACA), August 1998, Praha, Czech Republic. Two invited talks at two mini-symposia.

Annual ACM International Symposium on Symbolic and Algebraic Computations (ISSAC'98), August 1998, Rostock, Germany. Refereed paper was accepted by Program Committee.

5th International Symposium on Solving Irregularly Structured Problems Parallel (IRREGULAR'98), August 1998, Berkeley, California. Refereed paper was accepted by Program Committee.
MSRI Workshop on Solving Systems of Equations, September 1998, Berkeley, California. Invited talk (30 minutes).
39th Annual IEEE Conference on Foundations of Computer Science (FOCS'98), October 1998, Palo Alto, California. Refereed paper was accepted by Program Committee.

1999

10th Annual ACM-SIAM Symposium on Discrete Algorithms (SODA'99), January 1999, Baltimore, Maryland. Refereed paper was accepted by Program Committee.
13th International Parallel Processing Symposium and 10th Symposium on Parallel and Distributed Computing (IPPS/SPDP'99), San Juan, Puerto Rico, April 1999. Refereed paper was accepted by Program Committee.
31st Annual ACM Symposium on Theory of Computing (STOC'99), May 1999, Atlanta, Georgia. Refereed paper was accepted by Program Committee.
2nd International Workshop on Computer Algebra in Scienti c Computing (CASC'99), June 1999, Munich, Germany. Invited lecture (45 minutes).
Annual International Conference IMACS on Application of Computer Algebra (ACA), June 1999, El Escorial, Madrid, Spain. Two invited lectures at two mini-symposia.
1999 AMS-IMS-SIAM Summer Research Conference on Structured Matrices in Operator Theory, Numerical Analysis, Control, Signal and Image Processing, June-July 1999, Boulder, Colorado. Invited lecture (45 minutes).
Annual International Colloquium on Automata, Languages, Programming (ICALP), July 1999, Praha, Czech Republic. Refereed paper was accepted by Program Committee.
International Symposium on Foundations of Computational Mathematics (FoCM'99), July 1999, Oxford, England. Four invited talks at three mini-symposia.

2000

11th Annual ACM-SIAM Symposium on Discrete Algorithm (SODA'2000), January 2000, San Francisco. Refereed paper was accepted by Program Committee.
Sixth Biennial Copper Mountain Conference on Iterative Methods (Copper'2000), April 2000, Copper Mountain, Colorado. Refereed paper was accepted by Program Committee.
2nd Conference on Numerical Analysis and Applications (NAA'2000), June 2000, Rousse, Bulgaria. Invited plenary talk (1 hour).
Annual International Conference IMACS on Application of Computer Algebra (ACA), June 2000, St. Petersburg, Russia. Invited plenary talk (50 minutes) and invited talk at mini-symposium.
14th International Symposium on Mathematical Theory of Network and Systems (MTNS'2000), June 2000, Perpignan, France. Invited talk at mini-symposium.
The Smalefest Conference in Hong Kong, July 2000. Two papers were refereed and accepted for the proceedings.
Annual ACM International Symposium on Symbolic and Algebraic Computation (ISSAC'2000), August 2000, St. Andrew's, Scotland. Refereed paper was accepted by Program Committee.

2001

International Conference on Complex Analysis and Applications. Moscow, Russia, June 2001. Invited talk (45 minutes) and a refereed paper accepted for the proceedings.
SIAM Annual Meeting, San Diego, California, July 2001. Invited talk at minisymposium.
Annual ACM International Symposium on Symbolic and Algebraic Computations (ISSAC'2001), London, Ontario, Canada, July 2001. Refereed paper was accepted by the Program Committee.
2001 AMS-IMS-SIAM Summer Research Conference on Fast Algorithms in Mathematics, Computer Science, and Engineering. S. Hadley, Massachusetts, August 2001. Invited Lecture (45 minutes).

2002

Annual International Symposium on Theoretical Aspects of Computer Science (STACS). March 2002, Juan Les Pins, France. Refereed paper was accepted by the Program Committee.
Seventh Biennial Copper Mountain Conference on Iterative Methods (Copper'2002), March-April 2002, Copper Mountain, Colorado. Refereed paper was accepted by the Program Committee.
International Conference on Structured Matrices, May-June 2002, Hong Kong, China. Invited talk at a mini-symposium.
First Joint Meeting of the American Mathematical Society and Unione Matematica Italiana (AMS/UMI'2002), Pisa, Italy, June 2002. Invited talks at a session.
Annual International Conference IMACS on Application of Computer Algebra (ACA), Volos, Greece, June 2002. Two invited talks at two mini-symposia.
Annual ACM International Symposium Symbolic and Algebraic Computation (ISSAC'2002), Lille, France, July 2002. Refereed paper was accepted by the Program Committee.
International Symposium on Foundations of Computational Mathematics (FoCM'2002), Minneapolis, Minnesota, August 2002. Two invited talks at a mini-symposium.
5th Annual Conference on Computer Algebra in Scienti c Computing (CASC'2002), Yalta, Crimea, Ukrain, September 2002. Refereed paper was accepted by the Program Committee.

2003

International Seminar on Matrix Methods and Operator Equations, Moscow, Russia, June 2003. Invited talk.
Workshop on Nonlinear Approximation in Numerical Analysis, Moscow, Russia, June 2003. Invited talk.
SIAM Conference on Linear Algebra (LA'03), Williamsburg, Virginia, July 2003. Invited talk at a mini-symposium and a contributed talk.
9th Annual International Conference on Applications of Computer Algebra (ACA'2003), Raleigh, North Carolina, July 2003. An invited talk at mini-symposium.
6th Annual Conference on Computer Algebra in Scienti c Computing (CASC'2003), Passau, Germany, September 2003. Refereed paper was accepted by the Program Committee.

2004

Eighth Biennial Copper Mountain Conference on Iterative Methods (Copper'2004), March-April 2004, Copper Mountain, Colorado. Refereed paper was accepted by the Program Committee.
Mathematics of Computer Algebra and Analysis (MOCAA'2004). A talk by invitation by Program Committee.
16th International Symposium on Mathematical Theory of Network and Systems (MTNS'2004), July 2004, Leuven, Belgium. Refereed paper was accepted by the Program Committee.

6th Annual Conference on Computer Algebra in Scienti c Computing (CASC'2003), July 2004, St. Petersburg, Russia. Refereed paper was accepted by the Program Committee.

6th International Mathematica Symposium (IMS 2004), August 2004, Ban , Canada, Refereed paper was accepted by the Program Committee.

2nd International Conference on Structured Numerical Linear Algebra Problems: Algorithms and Applications (Cortona 2004), September 2004, Cortona, Italy. Invited talk (30 minutes).

2005

International Conference on Matrix Methods and Operator Equations, Moscow, Russia, June 2005. Invited talk (30 minutes).

16th Annual ACM-SIAM Symposium on Discrete Algorithm (SODA'2005), January 2005, Vancouver, Canada. Refereed paper was accepted by Program Committee.

International Conference on Foundation of Computational Mathematics (FoCM'2005), July 2005, Santander, Spain. Two invited talks (50 minutes and 25 minutes) at two minisymposia.

International Workshop on Symbolic-Numeric Computation, July 2005, Xi'an, China. Invited plenary talk (one hour).

Conference on Applications of Computer Algebra, July-August 2005, Nara, Japan. Invited talk at mini-symposium.

2006

Nineth Biennial Copper Mountain Conference on Iterative Methods (CMCIM'06), April 2006, Copper Mountain, Colorado. Refereed paper was accepted by the Program Committee.

International Conference on Algebraic Computational Geometry, Nice, France, June 2006. Invited talk (30 minutes).

Conference on Applications of Computer Algebra, Varna, Bulgaria, June 2006. Two invited talks at a mini-symposium.

SIAM Annual Meeting, Boston, Massachusetts, July 2006. Refereed paper was accepted by the Program Committee.

2007

The 6th International Congress on Industrial and Applied Mathematics (ICIAM'2007), Zurich, Switzerland, July 2007. Invited talk at a mini-symposium.

2nd International Conference on Matrix Methods and Operator Equations, Moscow, Russia, July 2007. Invited talk (30 minutes).

International Workshop on Symbolic-Numerical Computations (SNC'2007), London, Ontario, Canada, July 2007. Three refereed papers were accepted by the Program Committee. 2008

Tenth Biennial Copper Mountain Conference on Iterative Methods (CMCIM'06), April 2008, Copper Mountain, Colorado. Refereed paper was accepted by the Program Committee.

Third International Computer Science Symposium in Russia (CSR'2008), June 2008, Moscow, Russia. Refereed paper was accepted by the Program Committee.

The XIX International Workshop on Operator Theory and its Applications, July 2008, Williamsburg, Virginia. Invited talk at a mini-symposium.

Structured Linear Algebra Problems: Analysis, Algorithms, and Applications, Cortona, Italy, September, 2008. Invited talk, 30 minutes.

2009

International Conference on Polynomial Computer Algebra, St. Petersburg, Russia, April 2009. Invited Speaker.

The 3rd International Workshop on Symbolic-Numeric Computation (SNC 2009), Kyoto, Japan, August 2009. Invited talk (1 hour) and a refereed paper was accepted by the Program Committee.

SIAM Conference on Applied Linear Algebra, Oct. 26-29, Seaside, California, Oct. 2629. Two invited talks at two mini-symposia.

2010

International Conference on Polynomial Computer Algebra, St. Petersburg, Russia, April 2010. Invited Speaker.

The Fifth International Computer Science Symposium in Russia (CSR'2010), June 2010, Kazan, Russia. Refereed paper was accepted by the Program Committee.

The 16-th ILAS Conference, Pisa, Italy, June 2010. Invited talk, 30 minutes.

Annual ACM International Symposium on Symbolic and Algebraic Computation (ISAAC'2001), Munich, Germany, July 2010. Refereed paper was accepted by the Program Committee.

2011

Annual ACM SIGSAM International Symposium on Symbolic and Algebraic Computation (ISAAC'2011), San Jose, CA, June 8-11, 2011. Refereed paper was accepted by the Program Committee.

The 4th International Workshop on Symbolic-Numeric Computation (SNC'2011), San Jose, CA, June 7-9, 2011. Refereed paper was accepted by the Program Committee.

3rd International Conference on Matrix Methods in Mathematics and Applications, Moscow, Russia, June 22-25, 2011. Plenary talk (1 hour) and invited talk (30 minutes).

The 7th International Congress on Industrial and Applied Mathematics (ICIAM'2011), Vancouver, British Columbia, Canada, July 18-22, 2011. Invited talk at a mini-symposium (30 minutes).

2012

SIAM International Conference on Linear Algebra, Valencia, Spain, June 18-22, 2012. Invited talk at a mini-symposium (30 minutes).

14th Annual Conference on Computer Algebra in Scienti c Computing (CASC'2012), September 3-6, 2012, Maribor, Slovenia. Two refereed papers were accepted by the Program Committee.

2nd International Conference on Structured Numerical Linear Algebra Problems: Algorithms and Applications (Leuven 2012), September 10-14, 2012, Leuven, Belgium. Invited talk (30 minutes).

2013

The 17-th ILAS Conference, Providence, R.I., June 3-7, 2013. Four invited talks at three mini-symposia (30 minutes each).

Annual ACM SIGSAM International Symposium on Symbolic and Algebraic Computation (ISAAC'2013), Boston, Massachusetts, June 23-26, 2013. Refereed paper was accepted by the Program Committee.

15th Annual Conference on Computer Algebra in Scienti c Computing (CASC'2013), September 9-13, 2013, Berlin, Germany. A refereed paper was accepted by the Program Committee.

2014

The Ninth International Computer Science Symposium in Russia (CSR'2014), June 2014, Moscow, Russia. Refereed paper was accepted by the Program Committee.

The 5th International Workshop on Symbolic-Numeric Computation (SNC'2014), July 2014, Shanghai, China. Two refereed paper were accepted by the Program Committee.

16th Annual Conference on Computer Algebra in Scienti c Computing (CASC'2014), September 8-12, 2014, Warsaw, Poland. Two refereed papers have been accepted by the Program Committee.
3rd International Conference on Structured Numerical Linear Algebra Problems: Algorithms and Applications, September 8-12, 2012, Kalamata, Greece. Invited talk (30 minutes).

2015

The 26th International Workshop on Operator Theory and its Applications, July 6 10, 2015, Tbilisi, Georgia. Invited talk at a mini-symposium (30 minutes).

Conference on Applications of Computer Algebra, Kalamata, Greece, July 20-23, 2016. Invited talk at a mini-symposium (30 minutes).

17th Annual Conference on Computer Algebra in Scienti c Computing (CASC'2014), September 10 14, 2015, Aachen, Germany. Two refereed papers have been accepted by the Program Committee.

4th International Conference on Matrix Methods in Mathematics and Applications (MMMA-2015), August 24 28, 2015, Skolkovo/Moscow, Russia. Invited talk at a minisymposium (30 minutes).

2015 SIAM Conference on Applied Linear Algebra, October 26 30, 2015, Atlanta, Georgia, USA. Invited talk at a mini-symposium (30 minutes).

2016

The Eleventh International Computer Science Symposium in Russia (CSR'2016), June 2016, Moscow, Russia. Refereed paper was accepted by the Program Committee.

Milestones in Computer Algebra (MICA 2016), July 16 18, 2016, University of Waterloo, Canada. Invited talk at a mini-symposium (30 minutes).

Workshop on Fast Direct Solvers, November 12 13, 2016, Purdue University, West Lafayette, Indiana. Invited talk (50 minutes).

2017

SIAM Conference on Computational Science and Engineering, February March 2017, Atlanta, Georgia, USA. Invited talk at a mini-symposium (30 minutes).
INdAM Meeting Structured Matrices in Numerical Linear Algebra: Analysis, Algorithms and Applications, Cortona, Italy, September 4-8, 2017. Invited talk (30 minutes).

2019

SIAM Conference on Computational Algebraic Geometry (SIAM ACG 2019), Bern, Switzerland, July 2019 (invited talk at a minisymposium (30 minutes).

The 5th International Conference on Matrix Methods in Mathematics and Applications (MMMA 2019) Moscow, Russia, August 19-23, 2019 (an invited talk, 30 minutes).

Computer Algebra in Scienti c Computing, Moscow (CASC 2019), Russia, August 26-30, 2019 (2 contributed talks, 30 minutes each, accepted by the Program Committee).

Biennial Conference on Mathematical Aspects of Computer and Information Sciences (MACIS 2019), Gebze-Istanbul, Turkey, November 13-15, 20193 (3 contributed talks, 30 minutes each, accepted by the Program Committee).

2020

Annual ACM SIGSAM International Symposium on Symbolic and Algebraic Computation (ISSAC'2020), Kalamata, Greece, July 20 23, 2020. Refereed paper was accepted by the Program Committee.

Computer Algebra in Scienti c Computing (CASC 2020), Linz, Austria, September 14 18, 2020 (2 contributed talks, 30 minutes each, accepted by the Program Committee).

Polynomial Computer Algebra (PCA'2020), 12 - 17 Oct., 2020, Euler International Mathematical Institute, Saint Petersburg, Russia (a contributed talk, 30 minutes, accepted by the Program Committee).

2021

SIAM Conference on Applied Linear Algebra (LA21), Virtual Conference Originally scheduled in New Orleans, Lousiana, USA; May 17 - 21, 2021 (an invited talk at a Minisymposium, 20 minutes).
Computer Algebra in Scienti c Computing (CASC 2021), Sochi, Russia, September 13 17, 2021 (2 contributed talks, 30 minutes each, accepted by the Program Committee).

2022

Computer Algebra in Scienti c Computing (CASC 2022), Gebze, Turkey, August 22

26, 2022 (a contributed talk, 30 minute, accepted by the Program Committee).

13th International Conference on Algorithms and Complexity, 13 16 June, 2023 Larnaca, Cyprus (a contributed talk, 30 minutes, accepted by the Program Committee).

25th Conference of International Linear Algebra Society, June 20-24, 2023, Madrid, Spain, contributed talk, 30 minutes, accepted by the Program Committee).

2023

10th International Congress on Industrial and Applied Mathematics (ICIAM 2023), Tokyo : August 20-25, 2023 Invited talk at Minisymposium on Randomized methods for solving linear systems and eigenvalue problems (30 minute talk, accepted by the Program Committee)

12 RESEARCH

I will begin with my research Manifesto, then will brie y cover my education and my research in ten major subject areas of Computer Science and Computational Mathematics (omitting my work of 1965 75 in Economics in the USSR and a number of my more sporadic research excursions into other areas). I will end with a summary and concluding remarks. I will use the acronyms listed in Section 12.16 and followed by the list of the references cited in this section, and I will also refer to my works cited in Section 10 PUBLICATIONS (COMPLETE LIST).

12.1 MANIFESTO

I have been working in Mathematics, Computational Mathematics, and Computer Science for more than ve decades, facing research challenges and seeking new insights and novel methods. I was thrilled whenever I discovered new keys that opened challenging scienti c locks, particularly when a single key opened a number of locks, as this was the case with my techniques of active operation/linear substitution, trilinear aggregation, and transformation of matrix structures.

My work has contributed to the creation of the elds of the Complexity of Algebraic Computations and Algebraic Multigrid and to signi cantly advancing some other research areas such as Computations with Structured Matrices, Symbolic-Numerical Computations, and Fast and Processor-E cient Parallel Algorithms. My techniques, insights, concepts and de nitions are commonly used, sometimes as folklore.

I was lucky to reveal a number of important but hidden links among apparently distant subjects. I helped bring together research in various areas of computing such as symbolic computations, numerical computations, theoretical computer science, and applied linear algebra in many cases I achieved synergy.

I am grateful for recognition and support of my research by leading experts, foundations, journals, professional societies, research centers, and universities. The National Science Foundation (NSF) has awarded me with Grants for over $2,500,000 for 1980 2021, including Special Creativity Extension Award from the Numeric, Symbolic, and Geometric Computation Program of the CCR Division in the Directorate CISE of NSF in 1993 and over $1,000,000 in grants for 2016 2021. My Awards from Professional Sta Congress of the City University of New York (PSC CUNY) for 1989 201218 exceed $130,000.

I was encouraged by enthusiastic reviews and citations of my work in books, journals, and magazines and by designation of a Fellow of American Mathematical Society of 2013 For Contributions to the Mathematical Theory of Computation".

According to Google Scholar, I published four books (1623+LXXIV pages overall), over 20 surveys in journals and book chapters, over 180 research articles in journals and over 100 in refereed conference proceedings and was cited over 12,000 times. Almost all my publications are in Computer Science and Computational and Applied Mathematics.

I have also disseminated my research ndings in my lectures at the universities, research centers, and professional conferences worldwide as well as through my research reports, the Internet, and personal communication.

I guided 26 students to their PhD Defenses in Math and Compute Science (two in 2020) and published dozens of papers jointly with my current and former students; with some of them more than a decade after their defense.

12.2 Education and research areas

My scienti c destiny was decided in the 59th high school in Moscow, Russia, celebrated for having excellent teachers in mathematics. I was among many of its graduates who went to the famous MechMat Department of Moscow State University (MGU), headed by Andrey Nikolaevich Kolmogorov. He was one of the greatest mathematician of his time, and so was his student Vladimir Igorevich Arnold, also a graduate from the 59th school in Moscow.

My adviser Anatoli Georgievich Vitushkin, a renowned expert in the theory of functions of real and complex variables and a member of the Russian Academy of Sciences, was among Kolmogorov’s distinguished disciples. He also worked with a versatile scientist Alexander Semenovich Kronrod and like Kolmogorov and Kronrod had broad scienti c interests.

From 1956 to 1961 I enjoyed learning mathematics in the MechMat Department of MGU. My rst journal paper appeared in 1958 and was on the real function theory, but at that time Vitushkin guided me into research in Computational Mathematics, and from 1959 to 1964 almost all my publications as well as my PhD Thesis were in that eld. I defended the thesis in 1964, and then up to the Fall of 1976 had been making living by working in Economics rather than Mathematics because job market in the USSR was quite restricted for people of Jewish ethnicity, like myself. In 1976 I emigrated to the USA and since 1977 have been working entirely in Computer Science and Computational Mathematics.

12.3 My rst scienti c breakthrough: polynomial evaluation

In 1962, by introducing a novel technique of active operation/linear substitution, I proved optimality of the classical algorithm for polynomial evaluation, commonly called Horner's. This gave positive answer to a question asked by Alexander Markowich Ostrowsky in 1955. Volker Strassen and Shmuel Winograd adopted my technique for proving the optimality of the classical algorithms for some fundamental matrix computations (see [BM75, Section
2.3]).

My work has been surveyed in my paper [P66] and in the most fundamental Computer Science book [K81/97] by Donald E. Knuth, which cites my work and that of Richard P. Brent most extensively among all its cited authors. The paper [P66] has been highly recognized in the West, has led to the emergence of the eld of Complexity of Algebraic Computations, and made me known as "polynomial Pan".

12.4 My second scienti c breakthrough: fast matrix multiplication by means of trilinear decomposition and aggregation

Matrix multiplication (hereafter referred to as MM) is one of the central subjects of the theory and practice of computing, and the scienti c world was tremendously impressed in 1969, when Strassen decreased the classical exponent 3 of MM to $\log_2 7 \approx 2.808$, that is, performed MM by using less than cubic time. In my book and my review article in SIAM Review in 1984, both much cited at that time, I praised his discovery as well as his subsequent extensive work on algebraic computations, while he himself has been attracted to this eld by my paper [P66] and has paid tribute to my work in his chapters, both called "Pan's method", in [S72] and [S74].

Further progress toward performing MM in quadratic time was expected to come shortly, but all attempts to decrease the exponent 2.808 de ed worldwide e ort for almost a decade, until I decreased it in 1978. This work of 1978 was recognized worldwide as a long-awaited breakthrough.

I quote the following excerpt from a letter by Donald E. Knuth with his permission:

I am convinced that his research on matrix multiplication was the most outstanding event in all of theoretical computer science during 1978. The problem he solved, to multiply $n \times n$ matrices with less than $O(n^{\log_2 7})$ operations, was not only a famous unsolved problem for many years, it also was worked on by all of the leading researchers in the eld, worldwide. Pan's breakthrough was based on combination of brilliant ideas, and there is no telling what new avenues this will open."

Indeed my techniques prompted fast new progress, with my participation. I have become widely known as "matrix Pan" and to the experts as "matrix and polynomial Pan".

I devised my fast MM algorithms by means of

(i) reducing the bilinear problem of matrix multiplication to the equivalent problem oftrilinear (tensor) decomposition and

(ii) nontrivially exploiting cyclic symmetry in the tensor of matrix multiplication.

In [P78] I called my combination of the two techniques trilinear aggregation in [P78], but I introduced it already in the paper [P72] (in Russian), translated into English only in 2014, in arXiv:1411.1972, and little known in the West until 1978.

Actually my trilinear aggregation technique of 1972 was a historic landmark on a wider scale. It produced the rst nontrivial decomposition of a tensor and the associated trilinear form that de ned a new e cient algorithm for matrix computations. Subsequently tensor decomposition has become a popular tool for devising highly e cient matrix algorithms in many areas of scienti c computing. Says Eugene E. Tyrtyshnikov, a renowned expert in tensor decomposition:

We should be especially grateful to Victor Pan for the link between the bilinear algorithms and trilinear tensor decompositions. Although it looks simple and even might be regarded as a folklore by now, this observation still has its creator, and by all means and for all I know it is due to the work of Victor Pan."

Lately experts pointed me out that Richard Brent had report of 1970 where he also expressed matrix multiplication as a similar tensor decomposition. Not to diminish the value of that work, it has not gone beyond stating the link of MM to tensors but showed no application to devising new faster algorithms. Also in my extensive discussions of fast MM with all leading experts from the 1970s and throughout the 1990s Brent's report was never cited and apparently was not known; certainly it was not known in the Soviet Union in 1972, when I published my paper [P72].

Since 1978 my trilinear aggregation has been routinely employed by myself and my successors for devising new fast MM algorithms. After the stalemate from 1969 to 1978 the MM exponent was decreased a number of times in 1979 1981 and then again twice in 1986, reaching the record value 2.376 in [CW86/90]. It was decreased again in 2010 2014, but only nominally. Every decrease relied on amazing novel techniques built on the top of the previous ones, always employing the reduction of the MM problem to trilinear aggregation, frequently by default, as this has been pointed out on page 255 of the celebrated paper [CW86/90] about its immediate predecessor [S86]: Strassen uses the following basic trilinear identity, related to Victor Pan's trilinear aggregation" (1978)."

As Arnold Sch nhage has written at the end of the introduction of his seminal paper [S81], however, all these exponents of MM have been just "of theoretical interest". They hold only for inputs "beyond any practical size", and "Pan's estimates of 1978 for moderate" input sizes were "still unbeaten". Actually in [P79], [P80], [P81], and [P82], I successively decreased my record exponent for all feasible MM (that is, for MM of moderate sizes $n \times n$, say, up to $n \leq 1{,}000{,}000{,}000$). My exponent of [P82], below 2.7734, still remains the record in 2021. All smaller exponents rely on ignoring the curse of recursion they have been obtained only at the end of a long recursive processes, whose each recursive step squared the input size. The resulting algorithms beat the classical MM only for inputs of immense sizes.

My algorithms promise to be highly e cient in practice: the implementations by Igor Kaporin of an algorithm from [P84a] in [K99] and of that of [LPS92] in [K04] use substantially smaller computer memory and are more stable numerically than Strassen's algorithm.

I surveyed the progress up to the date in [P84b] and [P84a]. In both cases I focused on the decrease of the exponent of MM because this was the focus of the research community in 1984; presently I pay more attention to the acceleration of feasible MM.

In [HP98], jointly with my student Xiaohan Huang, I accelerated rectangular MM, which implied new record asymptotic complexity estimates for the computations of the composition and factorization of univariate polynomials over nite elds.

12.5 Hierarchical aggregation as a springboard for the Algebraic Multigrid (1980). Compact Multigrid (1990 1993)

In [MP80], jointly with Willard L. Miranker, I introduced hierarchical aggregation/disaggregation processes, substantially responsible for the emergence of the popular eld of Algebraic Multigrid.

Jointly with John H. Reif, in SPAA 1990, SIAM J. of Scienti c and Statistical Computing 1992 and CAMWA 1990 and 1993, I proposed a simple but novel acceleration technique of Compact Multigrid.

12.6 Parallel algebraic and graph algorithms (1985 2001)

Throughout the years of 1985 2001, prompted by high recognition of my joint paper with Reif at STOC 1985, I proposed, both by myself and jointly with coauthors, a variety of new e cient parallel algorithms and in particular a number of fast and processor-e cient parallel algorithms for computations with matrices, polynomials, and graphs. They relied on a number of our novel nontrivial techniques; I regularly presented my work at the most competitive conferences in this eld such as ACM STOC, IEEE FOCS, ICALP, and ACMSIAM SODA and published them in leading journals such as SICOMP, JCSS, Algorithmica, and Information and Computation. The study of processor e ciency is critical for the practice of parallel computation but was a novelty in 1985 for the researchers in the Theory of Computing.

a) Fast and processor e cient algorithms for matrix and polynomial computations. In STOC 1985, jointly with Reif, I introduced fast and processor e cient parallel algorithms for the solution of dense and sparse linear systems of equations. The algorithm for sparse linear systems of equations has been implemented on the supercomputers of NASA and Thinking Machines Corp. By myself and jointly with coauthors I continued working on parallel matrix and polynomial computations for more than a decade. We proposed nontrivial novel techniques, extended the list of the known fast and processor e cient parallel algorithms, and improved the known complexity bounds for the following fundamental computational problems: (i) the solution of general and structured linear systems of equations with integer input (see my papers in TCS 1987, IPL 1989, and SICOMP 2000) and over abstract elds (see my paper in CAMWA 1992 and my joint papers with Dario A. Bini and Luca Gemignani in ICALP 1991 and with Erich Kaltofen in SPAA 1991 and FOCS 1992), (ii) the computation of polynomial greatest common divisors (GCDs), least common multiples, and PadØ approximations (see my papers in CAMWA 1992 and TCS 1996), (iii) polynomial division (see my joint papers with Bini in J. of Complexity 1986, FOCS 1992, and SICOMP 1993), and (iv) the computation of the determinant, the characteristic polynomial, and the inverse of a matrix (see my joint papers with Zvi Galil in IPL 1989 and Xiohan Huang in J. of Complexity 1998). In 1985 86 part of my work on parallel algorithms was covered in the magazines Science, Science News, and Byte.

b) Graph algorithms. By myself and jointly with coauthors, I published a number of fast and processor e cient parallel algorithms for the computation of matching and paths in graphs. They relied on combining some novel techniques and nontrivial known reductions to matrix computations. I published these results in FOCS 1985 and Combinatorica 1988 jointly with Galil, in JCSS 1989, IPL 1991, and SICOMP 1993 jointly with Reif, in SICOMP 1995 jointly with Franco Preparata, in Algorithmica of 1997 jointly with Yijie Han and Reif, and in my own chapter in the Handbook on Computer Science of 1993.

c) In my joint works with David Shallcross and my student Yu Lin Kriz, publishedin SODA 1992, FOCS 1993, and SICOMP 1998, I proved NC-equivalence of the integer GCD and planar integer linear programming problems, which was a well-known theoretical challenge.

12.7 Univariate polynomial root- nding (1985 2017). Nearly optimal solution of a four millennia old problem

Univariate polynomial root- nding has been central in mathematics and computational mathematics for four millennia. It was studied already on Sumerian clay tablets and Egyptian papyrus scrolls but also has modern applications to signal processing, nancial mathematics, control theory, computational algebraic geometry, computer algebra and geometric modeling.

Hundreds of e cient algorithms have been proposed for its solution. Two-part book published with Elsevier, by John M. McNamee in 2007 (354 pages) and jointly by J.M. McNamee and myself in 2013 (728 pages), covers nearly all of them up to the date, in a unique comprehensive coverage of this popular subject area. Since 1985 I have been doing research in that area and in the related areas of computation of approximate polynomial GCDs, matrix eigenvalues and eigenvectors, and the solution of a system of multivariate polynomial equations. Next I brie y outline some of my results. See further information in my papers cited below in parts (a) (g) and the papers (individual and joint with my students) in FOCS 1985 and 1987, CAMWA 1985, 1987, 1995, 1996, 2011 (two papers), and 2012 (two papers, one of them joint with McNamee), SICOMP 1994, J. of Complexity 1996 and 2000 (four papers), JSC 1996, ISSAC 2010 and 2011, and SNC 2011 and 2014 (two papers).

a) In STOC 1995 (and also in CAMWA 1996, ISSAC 2001, and JSC 2002) I combinedthe advanced techniques by Sch nhage and by Andy C. Ne and Reif with my novelties in exploiting the geometry of the complex plane, precision control by using PadØ approximation, and recursive lifting and descending. As a result I have substantially accelerated the known algorithms. My divide-and-conquer algorithm of STOC 1995 approximates all roots of a univariate polynomial nearly as fast as one can access the input coe cients in record and (up to a polylogarithmic factor) optimal Boolean time. I have surveyed my work up to the date in SIAM Review 1997 [P97] and more informally in American Scientist 1998 [P98]. I cover it in some detail in JSC 2002 [P02] and Chapter 15 of my book of 2013, joint with McNamee and already cited.

b) Hermann Weyl's Quad-tree construction of 1924 enables the solution of a univariatepolynomial equation in roughly quartic arithmetic time. James Renegar decreased the time bound to cubic in 1987, and I reached quadratic arithmetic time bound in J. of Complexity 2000. Most of the computations of my algorithm require low precision, which suggested that the extension of this work can yield nearly optimal Boolean time. This involved substantial technical challenges, eventually simpli ed in the process of studying the so called subdivision root- nders for real and complex root- nding; in the complex case they were precisely the Quad-tree construction. In [BSSXY16] and [BSSY18] Ruben Becker, Michael Sagralo , Vikram Sharma, and Chee Yap obtained nearly optimal complex subdivision root- nder. Their work boosted interest to that direction because the approach promises to be highly e cient in practice. Our paper [IPY18] has presented the rst implementation of this algorithm. In my papers in CASC 2019 (two papers), CASC 2020 (two papers), ISSAC 2020 (joint with Imbach), and in arXiv preprint 1805.12042, I presented a novel version of subdivision root- nder, which is signi cantly faster than [BSSXY16] and [BSSY18]. This acceleration becomes dramatic in the case where an input polynomial is given by a black box for its evaluation, which includes highly important classes of sparse polynomials, polynomials in Bernstein basis, and ones given by recurrence (such as Mandelbrot's polynomials) or in compressed form, such as $c_1(x-a)^d + c_2(x-b)^d$.

c) Approximation of the real roots of a polynomial is an important goal because in many applications, for example, to algebraic optimization, only r real roots are of interest and because frequently they are much less numerous than all n complex roots. In my joint papers with my students in SNC 2007, CAMWA 2011, CASC 2012 and 2014, and TCS 2017 I accelerated the known algorithms for this problem by a factor of n/r.

d) My algorithm in ISSAC 2013 and JCS 2016 (joint with Elias P. Tsigaridas) is nearlyoptimal for a more narrow goal of real polynomial root-re ning rather than root- nding. Likewise my algorithm (also joint with him) in SNC 2014 and TCS 2017 re nes all complex roots at a nearly optimal Boolean complexity bound.

e) Together with Bini in J. of Complexity 1996, with Bini and Gemignani in CAMWA2004, ETNA 2004, and Numerische Mathematik 2005, with McNamee in CAMWA 2012, by myself in CAMWA 2005, and jointly with my present and former students in ISSAC 2010, CAMWA 2011, LAA 2011, CASC 2012, SNC 2014, TCS 2017, and a chapter in the SNC volume of 2007, published by Birkh user, I proposed novel matrix methods for polynomial root- nding. Unlike many previous companion matrix methods, we preserve and exploit the structure of the associated companion and generalized companion matrices and yield numerically stable solution, while keeping the arithmetic cost at a low level.

I further extended these algorithms to the solution of the eigenproblem for a general matrix in SODA 2005 and CAMWA 2006 and 2008.

f) Jointly with Bini, I proposed and elaborated upon an algorithm that approximates all eigenvalues of a real symmetric tridiagonal matrix by using nearly optimal Boolean time. This is a popular and important problem of matrix computations. We proposed the rst algorithm of this kind, presented in some detail in SODA 1991 and then in Computing 1992 and SICOMP 1998.

g) Computation of approximate polynomial GCDs has important applications to control and signal processing. My papers in SODA 1998 and Information and Computation of 2001 yielded a new insight into this computational problem by exploiting its links to polynomial root- nding, matching in a graph, and PadØ approximation.

h) Both divide and conquer and Quad-tree (subdivision) root- nders involve isolationof some sets of polynomial roots from each other. In particular the isolation of roots lying in a xed disc on the complex plane from the other roots implies quadratic (rather than linear) convergence of Newton's iterations right from the start. The computational cost of achieveing isolation can be considerable, however, and the paper [PT13] proposed to decrease it by means of testing isolation by action, that is, by means of applying Newton's iterations and then verifying isolation by monitoring the behavior of the iterations. This recipe was later adopted in [BSSY18]. In divide and conquer algorithm of [P95] one has to increase the isolation of the roots in the unit disc $D(0,1) = \{x : |x| \leq 1\}$ or in the annuli $\{x : 1/q \leq x| \leq q\}$ for a xed $q > 1$ and achieves this by means of repeated squaring of the roots. This process is not costly but recursively increases the approximation errors. [P95] counters such a de ciency by combining the recursive lifting process of repeated squaring with recursive descending.

12.8 A system of multivariate polynomial equations (1996-2005). Best Paper Award

My joint papers with Bernard Mourrain in Calcolo 1996, STOC 1998 and J. of Complexity (Best Paper Award for 2000), with Didier Bondifalat and Mourrain in ISSAC 1998 and LAA 2000, with Mourrain and Olivier Ruatta in SICOMP 2003, and with Ioannis Z. Emiris in ISSAC 1997, JSC 2002, CASC 2003, and J. of Complexity 2005 introduced and analyzed a number of novel and now popular techniques and algorithms for the approximation of the roots of dense and sparse systems of multivariate polynomials. The algorithms exploits the structure of the associated matrices.

12.9 Matrix structures: uni cation and bene ts (1987 2017)

This area is highly important for both theory and practice of computing. It was studied already in the 19th century and with increased intensity in the recent decades because of important applications to a variety of areas of modern computing, including the hot subject of handling Big Data.

My contributions can be traced back to 1987 and include the results in the following directions, besides the applications to polynomial root- nding, already cited.

a) Uni cation of structured matrix computations by using their displacement representation and the transformation of matrix structures. The four most popular matrix structures of Toeplitz, Hankel, Vandermonde, and Cauchy types have di erent features, which allow di erent computational bene ts. In particular, the Cauchy matrix structure, unlike the three other ones, is invariant in both row and column interchange and allows approximation by rank structured matrices, which can be very e ciently handled by means of the Fast Multipole Method one of the Top 10 Algorithms of the 20th century [C00].

The matrices of all four classes share, however, an important feature: they can be represented in compressed form through their displacements of low rank. Every matrix M can be expressed via its displacements $AM-MB$ and $M-AMB$ under mild restriction on operator matrices A and B, and for each of the four classes of structured matrices and a proper pair of operator matrices of shift and/or diagonal scaling, the displacement has small rank and therefore can be represented with fewer parameters, typically with $O(n)$ parameters for an $n \times n$ structured matrix, having n^2 entries. By properly exploiting this representation and using advanced techniques, one can dramatically decrease the amount of computer memory and time required in computations with such matrices.

The approach was proposed in [KKM79] by Thomas Kailath, Sun-Yuan Kung, and Martin Morf, who demonstrated its power by multiplying by a vector an $n \times n$ Toeplitz-like matrix (having structure of Toeplitz type) by using $O(n)$ memory cells and $Q(n\log n)$ ops (oating point arithmetic operations). The MBA divide-and-conquer algorithm of 1980 by Morf and by Robert R. Bitmead and Brian D. O. Anderson has extended this KKM 1979 progress to the inversion of Toeplitz-like matrices and the solution of Toeplitz-like linear system of equations, and the natural challenge was the extension of these algorithms of 1979 and 1980 to the computations with important classes of matrices having structures of the three other types.

I contributed to further progress with my two books of 1994 (with Bini) and 2001 and dozens of papers by myself and joint with coauthors.

In ISSAC 1989 and MC 1990, I uni ed fast computations with the four listed matrix classes in a rather unexpected way. Namely, I observed that one can transform matrix structure at will by transforming the associated operator matrices, and moreover can do this just by multiplying a given structured matrix by Hankel and Vandermonde matrices and their transposes. By applying such transformations of matrix structures one can extend any successful algorithm for the inversion of the structured matrices of any of the four classes to the inversion of the matrices of the three other classes, and similarly for solving linear systems of equations.

Moreover one can always use the simple reversion matrix as a Hankel multiplier and frequently can use the matrix of the discrete Fourier transform or its Hermitian transpose as a Vandermonde multiplier. In some cases such transformations enable dramatic improvement of the known algorithms.

For example, in 1989 cubic time was required for the inversion of Cauchy-like matrices and for the Nevanlinna Pick fundamental problem of rational approximation, closely linked to this task. My transformations immediately decreased the known cubic upper bounds on the time-complexity of these highly important computational problems to nearly linear.

Unlike the multiplication algorithm of [KKM79], the MBA inversion algorithm is numerically unstable, however, and this limits applications of the latter recipe. Later, however, my approach has become basic for a stream of highly e cient practical numerical algorithms for Toeplitz linear systems of equations: the algorithms begin computations with the converse reduction to the Cauchy-like case and then exploit either the invariance of Cauchy structure in row and column interchange (cf. [GKO95]) or the link of this structure to the rank structure of matrices and consequently to the Fast Multipole Method (cf. [CGS07], [MRT05], [XXG12]). In view of such a link one is challenged to extend my approach to the uni cation of computations

with matrices having displacement and rank structures, which could be highly important for both theory and practice of matrix computations. Recent progress towards meeting this uni cation challenge was reported in [BT17].
In 2013 2017 I extended my method to Vandermonde and Cauchy matrix-by-vector multiplication, the solution of Vandermonde and Cauchy linear systems of equations, and polynomial and rational interpolation and multipoint evaluation. For all these classical problems, the known numerical algorithms, running with bounded precision (for example, the IEEE standard double precision), required quadratic arithmetic time, and I decreased it to nearly linear (see my papers in CASC 2013, LAA 2015 and MC 2017). For another application of my techniques, in [P16] I formally supported empirical observation of many researchers (which remained with no proof for decades) that a Vandermonde matrix is ill-conditioned (that is, close to singular) unless it is close (up to scaling) to the matrix of discrete Fourier transform, whose knots are nearly equally spaced on or near the unit circle centered in the origin.
b) For alternative and more direct uni cation of computations with structured matrices of the four classes, one can express them in terms of operations with the displacements. The MBA algorithm of 1980 does this for Toeplitz-like matrices. I extend it to Cauchy-like matrices rst jointly with my student Ai-Long Zheng in LAA 2000 (submitted in 1996) and then jointly with Vadim Olshevsky in FOCS 1998. In SODA 2000 and in chapter 5 of my book of 2001 I extended the MBA algorithm in a uni ed way for computations with various structured matrices.
c) E cient algorithms for structured matrices and links to polynomial and rational computations. In SIAM Review 1992 [P92], CAMWA 1992, 1993 (jointly with my students), TCS 1996, and Annals of Numerical Mathematics 1997 (by myself), and ICALP 1999, jointly with Olshevsky, I presented new e cient algorithms for various fundamental computations with structured matrices such as computing their ranks, characteristic and minimum polynomials, bases for their null spaces, and the solutions of structured linear systems of equations. Furthermore I have also extended successful methods for computations with structured matrices to some fundamental computations with polynomials and rational functions. Conversely, in SNC 2014 and TCS 2017, jointly with Tsigaridas, I deduced nearly optimal estimates for the Boolean complexity of some fundamental computations with Vandermonde and Cauchy matrices by reducing these computations to the ones for polynomials and modifying the known fast algorithms for the latter problems.

12.10 Newton’s iterations for general and structured matrix inversion

Newton’s iterations reduce matrix inversion to matrix multiplications, which is attractive for parallel computations and for computations with structured matrices. My paper with Robert Schreiber in SISSC 1991 presents nontrivial initialization policies for these iterations and their variations that enhance performance. In Chapter 6 of my book of 2001 and in my paper with my students in MC 2006 I improved performance of the iterations by applying homotopy continuation techniques.
In the case of structured matrices the main challenge is the slow-down of the computations due to the recursive increase of the displacement rank of the approximations to the inverse computed in the iterative process. I recalled, however, that displacement rank of the inverse is shared with the input matrix, and so in J. of Complexity 1992, IEEE Transaction on Parellel and Distributed Systems 1993, and SIMAX 1993 I proposed and elaborated upon a remedy by means of recursive re-compression, that is, by recursively compressing displacements of the computed approximations (by means of truncation of their SVDs). My resulting superfast solution algorithms run in nearly linear arithmetic time and allow processor e cient parallel implementation and uni cation over various classes of structured matrices. I presented these

results in Chapter 6 of my book of 2001, my paper of 2010 in Matrix Methods: Theory, Algorithms and Applications, and with coauthors in LAA 2002, TCS 2004, Numerical Algorithms 2004, and MC 2006.

12.11 Computation of the determinant of a matrix

This classical problem has important applications in modern computing, for example, to the computation of convex hulls and resultants, with further link to the solution of multivariate polynomial systems of equations.

a) In TCS 1987 (Appendix) and IPL 1988 I reduced the computation of the determinantof a matrix to the solution of linear systems of equations and then applied p-adic lifting to yield the solution e ciently. By extending this approach John Abbott, Manuel Bronstein and Thom Manders in ISSAC 1999, Wayne Eberly, Mark Giesbrecht and Gilles Villard in FOCS 2000, and myself jointly with Emiris in JSC 2003 obtained some of the most e cient known symbolic algorithms for the computation of the determinant of a matrix and the resultant of a polynomial system.

b) I published novel algorithms for computing determinants in TCS 1999, jointly withthree coauthors from INRIA, France, and in Algorithmica of 2001, jointly with my student Yanqiang Yu. The algorithms perform computations with single or double IEEE standard precision, based on algebraic techniques (in the TCS paper) and on numerical techniques (in the Algorithmica paper), use small arithmetic time, and certify the output. The TCS paper has accelerated the computations by means of output sensitive and randomization methods, novel in this context.

12.12 Synergy of symbolic and numerical computations

Numerical and symbolic algorithms are the backbone of modern computations for Sciences, Engineering, and Signal and Image Processing, but historically these two subject areas have been developed quite independently of one another, while combination of symbolic and numerical techniques can be highly bene cial.

Since the early 1990s I have been promoting such bene ts as an organizer of conferences, as a member of their Program Committees, and as the Managing Editor of four Special Issues of TCS on this subject in 2004, 2008, 2011 and 2013. Perhaps even stronger impact into this direction was from my books of 1994 (joint with Dario Bini), 2001 (by myself), and 2013 (joint with John M. McNamee) and from my surveys in SIAM Review 1992 and 1997, in NATO ASI Series published by Springer 1991, Academic Press 1992, and Kluwer 1998, in the electronic proceeding of IMACS/ACA 1998, and in my chapters (with co-authors) in four Handbooks of 1999, 2004, 2009, and 2014, as well as from dozens of my research papers. For example, the Special Issue of TCS on Symbolic-Numerical Algorithms in 2017 published my three joint papers two with Tsigaridas and one with my student Liang Zhao out of 13 papers of that Issue.

12.13 Randomized preprocessing (2007 2017). Addition of chaos stabilizes fundamental numerical matrix computations

Since 2007 I have been working on randomized pre-processing of matrix computations. I have contributed a new direction, new insight, and novel techniques to the popular area of randomized matrix computations. See my papers (some joint with my students) in SNC 2007 (two papers) and 2009, CSR 2008, 2010, and 2016, TCS 2008, CAMWA 2009, LAA of 2009, 2010 (two papers), 2011, 2012, 2013, 2015, and 2017 (two papers), ISSAC 2011, CASC 2015, and reports in arXiv: 1611.01391 and 1710.07946.

I have advanced the known numerical algorithms for both nonsingular and homogeneous singular linear systems of equations. In particular I proved that, with a probability near one, randomized multiplicative preprocessing numerically stabilizes Gaussian elimination with no pivoting (GENP) and block Gaussian elimination, and I obtained similar results for any nonsingular and well-conditioned (possibly sparse and structured) multiplicative preprocessor and for Gaussian random input. This should embolden the search for new e cient sparse and structured multipliers, and jointly with my students I proposed some new classes of them. Our extensive tests with real world inputs were in good accordance with our formal analysis. My work on this subject with my students appeared in TCS 2008, LAA 2012, LAA 2013, LAA 2015 and LAA 2017 (see [PZ17] and the references therein). GENP with randomized pre-processing should be practically valuable because pivoting (row/column interchange) is communication intensive and because Gaussian elimination is most used algorithm in matrix computations. Some implementation of GENP applied to an input preprocessed with ad hoc random multipliers appeared in a series of papers by Mark Baboulin et al. beginning in 2012. My study should help re ne such implementations and provide formal support for this approach.

12.14 Superfast and accurate low rank approximation

By extending our techniques I obtained substantial progress for low rank approximation (hereafter referred to as LRA) of a matrix. This is a central problems of modern computing because of its highly important applications to numerical linear algebra, machine learning, neural networks, and Big Data mining and analysis. In CSR 2016 (jointly with my student Liang Zhao) I proposed a new insight into this subject, provided formal support for the empirical power of various known sparse and structured multipliers and de ned some new classes of e cient multipliers.

Jointly with my students, I studied computation of LRA at sublinear cost, that is, by using much fewer ops and memory cells than the input matrix has entries. I call such algorithms superfast. They are indispensable in modern computations that access and handle matrices with billions entries, representing Big Data too Big to access and handle otherwise.

It is easy to prove that any superfast algorithm fails to compute accurate LRA of the worst case input. We also proved, however, that with a high probability the well-known Cross-Approximation (C-A) iterations compute accurate LRAs superfast in the case of (i) a small-norm perturbation of a random matrix of low rank and (ii) any input matrix allowing LRA (that is, having low numerical rank) and pre-processed with a Gaussian random multiplier.

I began our LRA study by trying to prove the e ciency of C-A iterations, which has been consistently observed empirically, and indeed I have provided some missing formal support for this empirical phenomenon as well as for the e ciency of some known randomized algorithms for LRA, but I have simpli ed and accelerated these algorithms. Jointly with my present and former students I proposed another major class of recursive LRA algorithms based on random sketching and proved their convergence to LRA under some rather mild assumptions on the decay of singular values of an input matrix. I have also introduced new insight into this subject and novel techniques for LRA. I published some results of this work jointly with my present and former students in [LP20] and [PLSZ20] and in arXiv:1906.04929 (April 3 2021) and arXiv:1906.04327 (April 22 2021).

12.15 Concluding remarks

Throughout my career my work has advanced the state of the art of various fundamental subjects of Computational Mathematics and Computer Science such as computations with general and structured matrices, polynomials, integers and graphs, for example, polynomial evaluation, interpolation, division, factorization, and root- nding, solution of general and structured linear systems of equations, computation of linear recurrences, matching and paths in graphs, and the sign and the value of the determinant of a matrix.

While devising new e cient algorithms, I proposed novel techniques and new insights and revealed hidden links among various subject areas and computational problems, for example, (i) between the techniques of Symbolic and Numerical Computation, (ii) between the methods for low rank approximation (LRA) proposed and developed by researchers in Computer Science and Numerical Linear Algebra, (iii) between matrix multiplication and tensor decomposition, (iv) among matrices with various structures, and (v) between LRA and Fast Multiplole method (FMM). This list can be readily extended.

Most of my novelties have been immediately recognized, e.g., my results on polynomial evaluation in [P66], on fast and processor e cient parallel algorithms in [PR85] (joint with John H. Reif), and on nearly optimal polynomial root- nding in [P95], but each of my trilinear aggregation of 1972 and my transformation of matrix structures of 1989 waited for six years before they became widely known and appreciated. Likewise the value of my contribution of 2000 to the quadtree root- nding is only now becoming recognized, but even in such cases it was rewarding to witness the progress in the eld resulted from my e ort.

My long survey [P66] attracted attention of Volker Strassen, Shmuel Winograd, and other renowned researchers, who extended my work into a new eld of Algebraic Complexity of Computations.[5] Their work in turn attracted me to this eld again. For another striking example of cross-fertilization, my renewed interest to this eld was prompted by the concise but far-fetching exposition in the book [BM75] by Allan Borodin and Ian Munro, which was the rst book in Math in English that I have read after moving to the USA in 1977. In 1979 I learned from Borodin that his interest to the eld was largely inspired by my paper [P66].

My paper with Willard L. Miranker [MP80] was pioneering for the eld of the Algebraic Multigrid, now popular.

My survey in SIAM Review in 1992, my book with Dario Bini, published by Birkh user in 1994, and dozens of my subsequent research papers (individual and joint with Dario Bini and with my students) have demonstrated synergy in combining the techniques of Symbolic and Numerical Computations.

My book with Dario Bini (1994) is called "Polynomial and Matrix Computations" and includes a number of new research results by the authors. It covers its title subjects both thoroughly and comprehensively according to its reviews (see some excerpts below) and was frequently cited, as well as my three other books (also devoted to polynomial and matrix computations) and my surveys in SIAM Review on matrix multiplication (1984), polynomial and matrix computations (1992), and polynomial root- nding (1997). Google Scholar lists over 12,000 citations of my work overall.

Excerpts from SIGACT News, ACM Press, 26, 2, pages 26 27, June 1995, by Steve Tate: We are now greeted with the release of a book covering the basic, foundational material of the algebraic algorithm eld, written by the authors who are leading researchers in the eld and are responsible for many of the current best algorithms......For researchers in the eld of algebraic algorithms, this is a must-have" book, both as a

[5] The next and seminal paper [W67] in this subject area begins with: Introduction.-In reference [1], V. Ya. Pan summarized the results about the minimum number of multiplications and additions required to compute a polynomial. In particular, Pan proved that the minimum number of multiplications/divisions required to compute $P_n(x)x = a_0 + a_1x + \cdots + a_nx^n$ is n. The theorem of this note includes this result of Pan's as a special case, and also shows that the minimum number of multiplications/divisions required to compute the product of an $n \times n$ matrix by a vector is $m \cdot n$."

reference and the review of basic material......In conclusion, for researchers in the eld of algebraic computing, I highly recommend this book as an essential addition to your bookshelf."
Excerpts from SIGSAM Bulletin, ACM Press, 30, 3, pages 21 23, September 1996, by Ioannis Emiris and Andre Galligo: The book covers an impressive range of algorithmic issues in Theoretical Computer Science, Symbolic Computer Algebra and Numerical Computation, and the presence of several latest methods and results makes it exciting to read. It would be useful to a specialist in any of the above areas who wishes to undergo a rigorous study of polynomial or matrix operations for large problems using exact or approximate arithmetic......The book is outstanding......We would strongly recommend this book as a reference for graduate course in symbolic computation or computer algebra. It can also supplement the reading in a course on scienti c computing, computer science theory or applied mathematics. In conclusion, the book by Bini and Pan is an excellent companion for researchers and advanced students. Given, moreover, that it is a handy reference book, it should be present in every good library."

12.16 Acronyms

"CACS" stands for "Proceedings of Conference on Applications of Computer Algebra" "CAMWA" stands for "Computers and Mathematics (with Applications)"
"CSR" stands for "Proceedings of Computer Science in Russia"
"FOCS" stands for "Proceedings of IEEE Symposium on Foundations of Computer Science"
"ICALP" stands for "Proceedings of International Colloquium on Automata Languages and Programming"
"IPL" stands for "Information Processing Letters"
"ISSAC" stands for "Proceedings of ACM International Symposium on Symbolic and Algebraic Computation"
"JCSS" stands for "Journal of Computer and System Sciences"
"JSC" stands for "Journal of Symbolic Computation"
"LAA" stands for "Linear Algebra and Its Applications"
"LNCS" stands for "Lecture Notes in Computer Science"
"MC" stands for "Mathematics of Computation"
"SICOMP" stands for "SIAM Journal on Computing"
"SIMAX" stands for "SIAM Journal on Matrix Analysis and Applications"
"SNC" stands for "Symbolic-Numerical Computations" or "Proceedings of Workshop on Symbolic-Numerical Computations"
"SODA" stands for "Proceedings of ACM-SIAM Symposium on Discrete Algorithms"
"SPAA" stands for "Proceedings of ACM Symposium on Parallel Algorithms and Architecture"
"STOC" stands for "Proceedings of ACM Symposium on Theory of Computing"
"TCS" stands for "Theoretical Computer Science"

References

[BM75] Allan Borodin and Ian Munro, The Computational Complexity of Algebraic and Numeric Problems, American Elsevier, NY, 1975.

[BSSY18] Ruben Becker, Michael Sagralo , Vikram Sharma, and Chee Yap, A NearOptimal Subdivision Algorithm for Complex Root Isolation based on the Pellet Test and Newton Iteration, Journal of Symbolic Computation, 86, 51 96, May June 2018.

[BT17] Bernhard Beckermann and Alex Townsend, On the singular values of matrices with displacement structure, SIAM J. on Matrix Analysis and Applications, 2017.

[C00] Barry A. Cipra, The Best of the 20th Century: Editors Name Top 10 Algorithms, SIAM News, 33, 4, 2, May 16, 2000.

[CGS07] Shivkumar Chandrasekaran, Ming Gu, X. Sun, Jianlin Xia, and Jiang Zhu, A Superfast Algorithm for Toeplitz Systems of Linear Equations, SIAM. J. on Matrix Analysis and Applications, 29, 4, 1247 1266, 2007.

[CW90/86] Don Coppersmith and Shmuel Winograd, Matrix Multiplicaton via Arithmetic Progressions. J. of Symbolic Computations, 9, 3, 251 280, 1990. Proc. version in 19th ACM Symposium on Theory of Computing (STOC 1987), 1 6, ACM Press, New York, NY, 1987. Also Research Report RC 12104, IBM T.J. Watson Research Center, August 1986.

[GKO95] Israel Gohberg, Thomas Kailath, and Vadim Olshevsky, Fast Gaussian Elimination with Partial Pivoting for Matrices with Displacement Structure, Mathematics of Computation, 64, 1557 1576, 1995.

[GL13] Gene H. Golub and Cleve F. Van Loan, Matrix Computations, The Johns Hopkins University Press, Baltimore, Maryland, 2013 (fourth edition).

[HP97] Xiohan Huang and Victor Y. Pan, Fast Rectangular Matrix Multiplication and Applications, Journal of Complexity, 14, 2, 257 299, 1998. Proc. version in Proc. of ACM International Symposium on Parallel Algebraic and Symbolic Computation (PASCO'97), 11 23, ACM Press, NY, 1997.

[IPY18] RØmi Imbach, Victor Y. Pan, and Chee Yap, Implementation of a Near-Optimal Complex Root Clustering Algorithm, Proc. of International Congress on Math Software (ICMS 2018), 2018.

[K81/97] Donald E. Knuth, The Art of Computer Programming, volume 2, Addison-Wesley, 1981 (second edition), 1997 (third edition).

[K99] Igor Kaporin, A Practical Algorithm for Faster Matrix Multiplication, Numerical Linear Algebra with Applications, 6, 8, 687 700, 1999.

[K04] Igor Kaporin, The Aggregation and Cancellation Techniques as a Practical Tool for Faster Matrix Multiplication, Theoretical Computer Science, 315, 2 3, 469 510, 2004.

[KKM78] Thomas Kailath, Sun-Yuan Kung, and Martin Morf, Displacement Ranks of Matrices and Linear Equations, Journal of Mathematical Analysis and Applications, 68, 395 407, 1979. See also Thomas Kailath, Sun-Yuan Kung, and Martin Morf, Displacement Ranks of Matrices and Linear Equations, American Mathematical Society (New Series), 1, 5, 769 773, 1979.

[LPS92] Julian Laderman, Victor Y. Pan, and Xuan-He Sha, On Practical Algorithms for Accelerated Matrix Multiplication, Linear Algebra and Its Applications, 162 164, 557 588, 1992.

[LP20] Qi Luan and Victor Y. Pan, CUR LRA at Sublinear Cost Based on Volume Maximization, LNCS 11989, In Book: Mathematical Aspects of Computer and Information Sciences (MACIS 2019), D. Salmanig et al

(Eds.), Springer Nature Switzerland AG 2020, Chapter No: 10, pages 1 17, Chapter DOI:10.1007/978-3-030-431204_10

[MP80] Willard L. Miranker and Victor Y. Pan, Methods of Aggregations, Linear Algebra and Its Applications, 29, 231 257, 1980.

[MRT05] Per Gunnar Martinsson, Vladimir Rokhlin, and Marc Tygert, A Fast Algorithm for the Inversion of Toeplitz Matrices, Computers and Mathematics with Applications, 50, 741 752, 2005.

[P66] Victor Y. Pan, On Methods of Computing the Values of Polynomials, Uspekhi Matematicheskikh Nauk, 21, 1(127), 103 134, 1966. [Transl. Russian Mathematical Surveys, 21, 1(127), 105 137, 1966.]

[P72] Victor Y. Pan, On Schemes for the Evaluation of Products and Inverses of Matrices (in Russian), Uspekhi Matematicheskikh Nauk, 27, 5 (167), 249 250, 1972.

[P78] Victor Y. Pan, Strassen's Algorithm Is Not Optimal: Trilinear Technique of Aggregating for Fast Matrix Multiplication, Proc. of the 19th Annual IEEE Symposium on Foundations of Computer Science (FOCS'78), 166 176, IEEE Computer Society Press, Long Beach, California, 1978.

[P79] Victor Y. Pan, Fields Extension and Trilinear Aggregating, Uniting and Canceling for the Acceleration of Matrix Multiplication, Proceedings of the 20th Annual IEEE Symposium on Foundations of Computer Science (FOCS'79), 28 38, IEEE Computer Society Press, Long Beach, California, 1979.

[P80] Victor Y. Pan, New Fast Algorithms for Matrix Operations, SlAM J. on Computing, 9, 2, 321 342, 1980, and Research Report RC 7555, IBM T.J. Watson Research Center, February 1979.

[P81] Victor Y. Pan, New Combinations of Methods for the Acceleration of Matrix Multiplications, Computers and Mathematics (with Applications), 7, 1, 73 125, 1981.

[P82] Victor Y. Pan, Trilinear Aggregating with Implicit Canceling for a New Acceleration of Matrix Multiplication, Computers and Mathematics (with Applications), 8, 1, 23-34, 1982.

[P84a] Victor Y. Pan, How Can We Speed up Matrix Multiplication? SIAM Review, 26, 3, 393 415, 1984.

[P84b] Victor Y. Pan, How to Multiply Matrices Faster, Lecture Notes in Computer Science, 179, Springer, Berlin, 1984.

[P92] Victor Y. Pan, Complexity of Computations with Matrices and Polynomials, SIAM Review, 34, 2, 225 262, 1992.

[P95] Victor Y. Pan, Optimal (up to Polylog Factors) Sequential and Parallel Algorithms for Approximating Complex Polynomial Zeros, Proc. 27th Annual ACM Symposium on Theory of Computing (STOC'95), 741 750, ACM Press, New York, 1995.

[P97] Victor Y. Pan, Solving a Polynomial Equation: Some History and Recent Progress, SIAM Review, 39, 2, 187 220, 1997.

[P98] Victor Y. Pan, Solving Polynomials with Computers, American Scientist, 86, 62 69, January-February 1998.

[P02] Victor Y. Pan, Univariate Polynomials: Nearly Optimal Algorithms for Numerical Factorization and Root Finding, J. of Symbolic Computation, 33, 5, 701 733, 2002.

[P16] Victor Y. Pan, How Bad Are Vandermonde Matrices? SIAM Journal of Matrix Analysis and Applications, 37, 2, 676 694, 2016.

[PLSZ20] Victor Y. Pan, Qi Luan, John Svadlenka, and Liang Zhao, Sublinear Cost Low Rank Approximation via Subspace Sampling, In LNCS 11989, Book: Mathematical

Aspects of Computer and Information Sciences (MACIS 2019), D. Salmanig et al (Eds.), Springer Nature Switzerland AG 2020, Chapter No: 9, pages 1 16, Springer

Nature Switzerland AG 2020 Chapter DOI:10.1007/978-3-030-43120-4_9

[PR85] Victor Y. Pan and John H. Reif, E cient Parallel Solution of Linear Systems, Proc. 17th Annual ACM Symposium on Theory of Computing (STOC'85), 143 152, ACM Press, New York, 1985.

[PT13] Victor Y. Pan and Elias P. Tsigaridas, On the Boolean Complexity of the Real Root Re nement, in Proc. International Symposium on Symbolic and Algebraic Computations (ISSAC'2013), Boston, Massachusetts, June 2013 (M. Kauers, editor), 299 306, ACM Press, New York, 2013.

[PZ17] Pan, V.Y., Zhao, L.: Numerically Safe Gaussian Elimination with No Pivoting, Linear Algebra and Its Applications, 527, 349 383 (2017) http://dx.doi.org/10.1016/j.laa.2017.04.007

[S69] Volker Strassen, Gaussian Elimination Is Not Optimal, Numerische Mathematik, 13, 354 356, 1969.

[S72] Volker Strassen, Evaluation of Rational Functions, pages 1 10, in Proceedings of a Symposium on the Complexity of Computer Computations, March 20-22, 1972, the IBM Thomas J. Watson Research Center, Yorktown Heights, New York, USA. Raymond E. Miller and James W. Thatcher editors. The IBM Research Symposia Series, Plenum Press, New York 1972, ISBN 0-306-30707-3

[S74] Volker Strassen, Some Results in Algebraic Complexity Theory, Proceedings of the International Congress of Mathematicians, Vancouver, 1974 (Ralph D. James, editor), Volume 1, Canadian Mathematical Society, pages 497 501, 1974.

[S81] Arnold Sch nhage, Partial and Total Matrix Multiplication. SIAM J. on Computing, 10, 3, 434 455, 1981.

[S86] Volker Strassen, The Asymptotic Spectrum of Tensors and the Exponent of Matrix Multiplication. Proceedings of the 27th Annual IEEE Symposium on Foundations of Computer Science (FOCS'86), 49-54, IEEE Computer Society Press, 1986.

[W67] Shmuel Winograd, On the Number of Multiplications Required to Compute Certain Functions, Proc. of National Academy of Sciences, 58, 1840 42, 1967.

[XXCB14] Yuanzhe Xi, Jianlin Xia, Stephen Cauley, and Venkataramanan Balakrishnan, Superfast and stable structured solvers for Toeplitz least squares via randomized sampling, SIAM J. Matrix Anal. Appl., 35 (2014), pp. 44-72

[XXG12] Jianlin Xia, Yuanzhe Xi and Ming Gu, SIAM J. on Matrix Analysis and Applications, 33, 837 858, 2012.

EVA SOFIANOS

esofianos@gmail.com 914-269-8747

twitter.com/EvaSofianos linkedin.com/in/eva-sofianos meetup.com/gdg-bronx

PROFESSIONAL SUMMARY

Energetic, disciplined and highly motivated professional with 20 years of experience in IT, diligently leading teams to success. Currently, a Computer Science Faculty Member and Deputy Chair of Computer Science at City University of New York (CUNY) Lehman College. Industry experience prior to transitioning to academia, includes implementing and customizing Cloud Solutions using International Business Machines (IBM) owned tools such as TPM, TSAM, and ISDM, which led to co-authoring an IBM RedBook. Earlier career focus includes web application development as well as system administration on Windows and Linux servers. Additional experience includes various research projects at IBM, the American Museum of Natural History (AMNH), and CUNY. Enjoys building communities and organizing events, as founder of the Google Developer Group (GDG) Bronx Chapter which has grown to over 3000 members. Served as North East Regional GDG Mentor, and Women Techmakers (WTM) Ambassador. Interpersonal and communication skills have been refined through years of working with a variety of executives, employees, businesses, clients, and students. Ability to consistently meet deadlines is demonstrated by handling multiple demanding projects, embracing challenges and exhibiting perseverance in conquering all obstacles.

EMPLOYMENT

CITY UNIVERSITY OF NEW YORK, HERBERT H. LEHMAN COLLEGE (June 2005 - Present)

Lecturer(01/2018-Present), *Substitute Lecturer*(08/2016-12/2017), *Adjunct*(06/2005-12/2006, 09/2013-08/2016)

- Design, develop, and deliver curricular lessons for students in the Computer Science Department
- Prepare exams, assign homework, review and correct them while providing appropriate feedback
- Serve as a faculty mentor for student organizations (SHPE, WiCS)
- Advise students on future course electives and selection of major field of study
- C

ourses taught: (June 2005 – December 2006), (September 2013 - Present)

- CMP 567 Programming Methods I for Educators (HTML, CSS, JavaScript, Java)
- CMP 485 Technical Interview Preparation (Skill Sharpening both soft and hard, plus portfolio preparation)
- CMP 430 Mobile Application Development for Android (Java, XML)
- CMP 342 Internet Programming (Java, Javascript, HTML, CSS)
- CMP 338 Data Structures & Algorithms (ADTs, Data Structures, Analysis of Algorithms using Java)
- CMP 168/326 Programming Methods II (Problem Solving and Programming using Java)
- CMP 167/CMP230 Programming Methods I (Structured Computer Programming using Java / Python)
- MAT 171 Precalculus Problem Solving for Economics and Life Sciences (Functions, Graphs, Matrices)
- MAT 132 Statistics/Data Analysis with Excel (Probability, Random Sampling, Confidence Intervals)

- MAT 104 College Algebra (Rational Expr., Logarithms, Quadratics, Complex Numbers, Trigonometry)

GOOGLE, Mountain View CA (June 2021 - July 2021)

Faculty in Residence

- Learned how to create an immersive learning environment via project oriented and project based curriculum
- Learned about Google's interview process and how to better prepare students for such interviews
- Prepared and developed plans for projects that could be used as part of the curriculum at Lehman College
- Collaborated with faculty and other Googlers to review and revise curriculum plans

CITY UNIVERSITY OF NEW YORK, HERBERT H. LEHMAN COLLEGE, Bronx NY (January 2016 - May 2017)

SCHOOL OF CONTINUING AND PROFESSIONAL STUDIES (SCPS)

Web Development Program Manager (March 2017 - May 2017)

- Prepared and developed program's social media marketing materials
- Recruited, screened, and selected the students as well as businesses for the program
- Planned and implemented the curriculum (HTML, CSS, JavaScript)
- Shared program curriculum and instructional modules for reuse
- Planned, organized and hosted technical networking events at CUNY On The Concourse (COTC)
- Prepared report summarizing outcomes, feedback and achievements from January 2016 - May 2017

Technology and Media Manager (July 2016 - September 2016)

- Developed technology and media marketing plan in line with organizational objectives
- Developed a new process to streamline managing the school's social media presence
- Managed and ran the social media campaigns and day-to-day activities
- Created a maintenance procedure manual for the platforms

Innovation Lab Startup Program Manager (January 2016 - June 2016)

- Planned, prepared, and hosted technical networking events geared toward startups and entrepreneurs
- Recruited trainees and aspiring entrepreneurs
- Planned and implemented training
- Guided participants on technical and business related topics
- Mentored participants on the value of entrepreneurship and networking

AMERICAN MUSEUM OF NATURAL HISTORY, New York, NY (July 2016 – August 2016)

Lead Instructor BridgeUp STEM summer program

- Presented lessons for Brown Scholars participating in the BridgeUp STEM program
- Managed team of instructors who assisted with hands on lessons
- Course included the following topics:
  - Programming using Python
  - Command Line Interface (CLI)

- Collaboration and code sharing using GitHub
- SQLite DB
- Data Analysis & Visualization using Python libraries and online tools such as Plotly

STATE UNIVERSITY OF NEW YORK, FARMINGDALE

(September 2015 – July 2018)

Part-time Assistant Professor of Computer Science

- Prepared and presented lessons in the Business Computing Systems Department
- Prepared exams, assigned homework, reviewed and corrected them while providing appropriate feedback
- Courses taught:
  - BCS 421 Android Mobile Application Development (design and develop native Android applications)
  - BCS 130 Website Development I (design and develop websites using HTML and CSS)
  - BCS 102 Computer Concepts & Applications (basic concepts and Microsoft Word, Excel, PowerPoint)

TRAINCUBE, Bronx NY (May 2015 – June 2017)

Co-Founder and Instructor

- Prepared marketing material and promote upcoming sessions
- Recruited, screened, interviewed, and matched small businesses technical needs' with trainees
- Planned, prepared and presented hands on curriculum customized for the technical problem of the business
- Mentored and managed trainees through development of their solution
- Connected trainees with small business owners to establish a relationship and encourage entrepreneurship
- Courses taught:
  - Full Stack Website Development & Hosting (MEAN stack - Mongo, Express, Angular, Node)
  - Android Native Mobile Application Development (Java, XML, Android Studio, Material Design)
  - Full Stack Website Development & Hosting (HTML, CSS, JavaScript, Firebase)

IBM Global Services Department, Poughkeepsie, NY (December 2006 – December 2015)

Advisory IT Specialist - Cloud Computing

- IBM Customer Account American Express - ISDM (Team Lead) (January 2012 - December 2015)
  - Designed and developed custom solutions to handle the intricacies of the client's cloud environment
  - Designed and implemented Agile solutions to meet the client's needs as the environment changed
  - Advised team members and executives on customization options, requirements and best practices
  - Performed installation, configuration and customization of ISDM and all related components
  - Performed research and analysis required to install, and configure ISDM 7.2.2 and later ISDM 7.2.4

    - Integrated the tool with VSS Admin and VMware ESX 4.0, 4.1, 5.0, 5.5 in multiple vCenter environments
    - Developed a notification system using SMTP to notify personnel of image customization failures
    - Trained team members and developed documentation for knowledge transfer and steady state support
    - Developed templates to provision Windows and Linux images that met stringent security standards
    - Established a process to remediate inconsistencies in the client's build process and remediate flaws
    - Languages/scripts used: Jython, Java, Perl, bash scripts on Linux, batch files on Windows
    - Co-Authored IBM Redbook Implementation Services for Private Clouds Implementation Guide
- IBM Customer Account Capital One – TPM (Team Lead) (May 2008 - July 2011)
    - Developed Jython and bash scripts to discover servers, install software, and schedule tasks via TPM
    - Performed troubleshooting and remediation of problematic software installations on servers
    - Packaged Windows software patches & products and automated their distribution using TPM
    - Handled all aspects of change orders from planning to closure using HPSM, while coordinating the schedules of team members, and change freezes to maximize the use of available resources
    - Tracked progress, created forecasts, generated pivot tables and reports in Excel for upper management
    - Trained new team members on TPM and HPSM, change order planning, implementation and tracking
    - Oversaw team members, assigned tasks, and coordinated team schedules to assure 24/7 availability

- IBM Customer Account -American Express MRO (Team Lead) (November 2007 - June 2008)
    - Received awards and accolades for my contributions to the successful outcome of this project
    - Automated the weekly creation of tables and charts for the AMEX MRO Project by developing Excel Add-ins and VBA Macros
    - Improved automation process to achieve 20 hours saved every week for each project manager
- IBM Services Connectivity Support (Team Lead) (January 2007 - June 2008)
    - Defined and developed a new procedure for ISC Level 2 Problem Ticket Support that enabled 3 team members to provide the same level and quality of support formerly provided by 100 people
    - Created a technical guide used to implement the new procedure
    - Led ongoing ISC Level 2 Problem Ticket Support (opening, analyzing, transferring, and closing tickets)
    - Provided ISC Problem Ticket Support for the TIM instance of ManageNow! by analyzing incoming tickets and modifying code integrations accordingly
    - Contributed to the successful completion and testing of weekend night changes
    - Trained team members to continue providing ISC Level 2 support using the new procedure
- IBM Daylight Savings Time (DST) (December 2006 - March 2007)
    - Received awards and accolades for my contributions to the successful outcome of this project
    - Worked with upper management to develop and implement a procedure consisting of continuous monitoring and analysis of contents on over 750 websites in order to distribute the newest information to IBMers working on the 2007 DST project
    - Performed administrative tasks on various DST related wikis to assure information was accurate

IBM T.J. WATSON RESEARCH CENTER, Hawthorne, NY (December 2003 – December 2006)

Co-op Software Engineer

- Designed, developed, deployed, and maintained a web based contact storage application with a GUI
- Designed, developed, deployed, and maintained a web application with a GUI to remotely monitor and control web servers and applications using RMI, and SMTP assuring high availability with rapid response time
- Developed and maintained web apps using Java Beans, Servlets, JSP, XML, JavaScript using WSAD/RAD
- Used IBM DB2, Java and JDBC driver with SQL to perform CRUD database operations for various applications
- Deployed intranet web applications using Apache Tomcat web server
- Performed Natural Language Processing (NLP) by parsing and tagging XML documents
- Developed RSS feeds using Java, XML and SAX
- Performed system administration on Windows and Linux servers, by developing batch files and bash scripts

AMERICAN MUSEUM OF NATURAL HISTORY, New York, NY (September 2003 – December 2004)

Mathematics & Biology Research Assistant

- Collected data from specimens using 3DX Microscribe and Cyberware 3030 RGB Laser Surface Scanner
- Prepared data for analysis by removing size differences using Generalized Procrustes Analysis
- Analyzed the data to create one exemplar specimen for each species using standard deviation methods and specialized computer software such as SPSS and various internal applications
- Presented research results at CUNY Poster Session as “Visualizing Ancestors on a Phylogenetic Tree”

ALLSECTOR TECHNOLOGY GROUP, New York, NY (September 2003 – December 2003)

Programming/Analyst Intern

- Reconstructed web based database using ColdFusion
- Manipulated existing database information to create user-friendly applications

WEILL MEDICAL CENTER OF CORNELL UNIVERSITY, New York, NY (June 2003 – September 2003)

Programming Intern

- Used Java Servlets, JSP, JDBC, SQL, HTML, along with MS Access to develop user friendly GUIs
- Developed the user portion to allow interaction with the database in a controlled and systematic manner
- Developed the administrative portion of the project to track and record users’ actions

GRANTS & AWARDS

Center for Inclusive Computing - Northeastern Data Grant (January 2021)

Data Collection and Analysis to Improve Student Success

CUNY, Lehman College - Academic Programs and Educational Effectiveness (January 2020)

Student Success Course Redesign Initiative (SSCRI) - High DWIF/High Enrollment General Education Courses

WiTNY (July 2018)

Developing and Introducing a new CS0 course into Pathways to recruit and retain students (especially female)

CTC - CEWP - JPMorgan Chase Grant (April 2018)

Redesigning and Enhancing Project-based Learning in Android Mobile Application Development

PUBLICATIONS & POSTERS

IBM RedBook (February 2013)

Implementation Services for Private Clouds - An Implementation Guide for Practitioners and Architects

ISBN-10: 0738438359, ISBN-13: 9780738438351, IBM Form Number SG24-8123-00, pages 550

CUNY Poster Session (December 2004)

Visualizing Ancestors on a Phylogenetic Tree

EDUCATION

CITY UNIVERSITY OF NEW YORK, LEHMAN COLLEGE, New York (December 2006)

M.S. Computer Science [GPA 3.67]

CITY UNIVERSITY OF NEW YORK, LEHMAN COLLEGE, New York (December 2004)

B. S. Computer Information Systems & Management, *minor Computer Science [GPA 3.95] Summa Cum Laude*

SCHOLARSHIPS & AWARDS

National Science Foundation Scholarship, Computer Science & Mathematics Lehman College Scholarship, National Supermarket Association Scholarship, Lehman Scholars Program, Golden Key Honor Society, Dean's List, Presidential Scholar, IBM BRAVO Awards, IBM THANKS Awards, IBM Top Contributor Award

PROFESSIONAL DEVELOPMENT

Certificate in Effective College Instruction (May 2021)

The only college teaching credential endorsed by the American Council on Education (ACE), Association of College and University Educators (ACUE), Lehman College CUNY, National https://api.badgr.io/public/assertions/m_soPakEQXmKhF1gW9qQyg

Workshop on Preparation for Teaching Online (May 2021)

Training to better serve our students and to be in compliance with the New York State Education Department requirements for distance education offerings as well as the Interregional Guidelines for the Evaluation of Distance Education Council of Regional Accrediting Commissions by the National Council for State Authorization Reciprocity Agreements (NC-SARA)

SERVICE

Committee Member, ***Catalyzing Upward Mobility & Economic Prosperity CUNY*** (January 2022 - June 2022)

Committee Member, ***Computer Science Major Gateways Committee*** (March 2021 - May 2021)

Committee Member, ***Lehman College Senate*** (Fall 2020 - Present)

Committee Member, ***Education Policy Committee - Computer Science*** (May 2020 - Present)

Committee Member, ***Education Policy Committee - Computer Information Systems*** (May 2020 - Present)

Deputy Chairperson, ***Computer Science Department Lehman College*** (Spring 2019 - Present)

Committee Member, ***Search and Hiring Committee for Dean of NSS Lehman College*** (September 2019 - April 2020)

SHPE - Lehman College, CUNY **Faculty Advisor Lehman College Chapter** (January 2019 - Present)

WiCS - Lehman College, CUNY **Faculty Advisor Lehman College ACM-W Chapter** (September 2018 - Present)

Google Developer Group (GDG) **Northeast USA Regional Mentor,** (March 2018 - June 2021)

Lehman College, CUNY **Computer Science Course Assessment Coordinator** (January 2018 - Present)

Holy Trinity Greek Orthodox Church Hicksville Girls Basketball Team - **Assistant Coach** (October 2018 - Present)

Girl Scouts of Nassau County Troop #1214 **Chapter Founder and Leader** (September 2017 - June 2018)

Lehman Scholars Program - Lehman College, CUNY **Faculty Mentor for thesis** (September 2017 - December 2017)

AMNH Hackathon **Mentor for the only team of underage participants, leading them to victory** (November 2015)

Code Interactive weare.ci **Mentor for programming students** (September 2015 - March 2016)

OpportunityNetwork.org **Mentor for students pursuing technical careers** (May 2016)

Google Developer Group (GDG) Bronx Chapter **Founder, Organizer with over 3000 members** (May 2015 - Present)

Annual NSBE Hackathon - Lehman College, CUNY **Judge or Mentor annually** (November 2014 - Present)

IBM Girls Go Tech Know Camp **Mentor for teams using Lego Mindstorms, Arduino Wearables** (June, July 2014)

PRESENTATIONS, PODCASTS, WORKSHOPS

CCSCNE 2021 Northeastern Region Conference **Panel Session - CUNY 2x Tech Faculty Workshop** (April 2021)

Enjoy the Vue Podcast Interview NY **Teaching CS & Preparing Students to Enter the workforce** (February 2021)

CUNY 2x Tech TTP Faculty Workshop NY **Panel Speaker on Best Pedagogical Practices** (January 2021)

GDG North America Academy 20' CA **Speaker on Team Building** (February 2020)

Thunder Nerds Podcast Interview 19' FL **Interview Behind the Scenes of DevFest** (November 2019)

DevFest 19' MA **Keynote Speaker on Salary Negotiation** (November 2019)

Thunder Nerds Podcast Interview 19' FL **Computer Science in the Bronx** (August 2019)

GDG North America Academy 19' CO **Speaker on Team Building** (July 2019)

GDG North America Academy 19' CO **Speaker on Growing Developer Groups** (July 2019)

Lehman College STEM Careers **Panel Speaker on technical career paths** (March 2019)

GDG North America Summit 19' **Speaker on Nailing the Technical Interview** (March 2019)

DevFest 18' CT, 19' FL, 19' ME **Speaker on Nailing the Technical Interview** (October 2018 - February 2019)

Lehman College STEM Careers **Panel Speaker on technical career paths** (October 2018)

CTRLSHE.org **Speaker on Android development** (March 2016)

Udacity NYC Meetup **Panel Speaker on technical career paths** (January 2016)

Curriculum Vitae

Name: Steven Fulakeza **College:** Lehman Col

RECOMMENDATION FOR:

APPOINTMENT:

PROMOTION:

REAPPOINTMENT:

REAPPOINTMENT WITH TENURE: X

OTHER (Sabbatical Leave, Designation ECP, etc.):

TITLE: Lecturer

DEPARTMENT: Computer Science

EFFECTIVE DATE:

SALARY RATE: $91,027.00

Initial Appointment Date: 08/27/2018

Tenure Date:

HIGHER EDUCATION (in reverse chronological order)

Institution	Dates Attended	Degree & Major	Date Conferred
Lehman College	08/14 - 12/16	M.S. Computer Science	01/01/17
Lehman College	08/08 - 05/13	B.S. Computer Science	05/30/13

EXPERIENCE (in reverse chronological order)

A. Teaching (at Lehman and any other institutions)

Institution	Dates	Rank	Department
Lehman College	08/20 - present	Lecturer	Computer Science
Lehman College	01/18 - 07/18	Substitute Lecturer	Computer Science
Bronx Community College	06/18 - 08/18	Adjunct Lecturer	Mathematics & Computer Science
Lehman College	10/15 - present	Adjunct Lecturer	Computer Science

B. Employment/Others

Employer/ Institution	Dates	Position	Department/Unit
City College of New York	10/14 - 01/18	Web Developer	Information Technology
John Jay College of Criminal Justice	08/12 - 11/14	Web Developer	Information Technology

ACADEMIC AND PROFESSIONAL HONORS (since last personnel action, with dates received, in reverse chronological order)

- City College of New York Behind the Scenes Information Technology, Excellence, Achievement and Merit Award (2017)
- CUNY Tech Innovations Award (2016)

PUBLICATIONS/CREATIVE WORKS (since last personnel action, in reverse chronological order)

Peer-Reviewed

Non-Peer-Reviewed

PUBLICATIONS / CREATIVE WORKS (prior to last personnel action, in reverse chronological order)

Peer-Reviewed

Non-Peer-Reviewed

PRESENTATIONS (prior to last personnel action, in reverse chronological order)

- MCSLC White Boarding Algorithms Treehouse (04/12/2019)
- STEM Careers - Technical Career Paths (03/08/2019)
- Data Visualization with Python Workshop (04/28/2018)
- Git & GitHub Workshop (11/03/2018)
- Creating your Digital Portfolio: A Pathway to Showing Your Skills to Any Employer in the World (11/16/2016)

PH.D. DISSERTATION/THESIS TITLE:

UNPUBLISHED WORK (Supported by evidence, including unpublished Ph.D. or Master's Thesis)

a. Works accepted for publication/Exhibition/Production

b. Works submitted for publication, exhibition and production

c. Works in progress

- Manuscripts in preparation
- Research in progress
- Exhibitions / Productions in preparation

GRANTS

- Current
    - CUNY - NYC Tech Talent Pipeline Career Success Course Innovation Grant - CMP 167, CMP 167, & CMP 338 Course Curriculum Re-Development, $10,000 2022 - 2023
    - CUNY - NYC Tech Talent Pipeline Career Success Course Innovation Grant - CMP 420 Course Curriculum Re-Development, $10,000 2022 - 2023
    - Northwestern University, Center for Inclusive Computing Programming Sequence Data Grant, $60,000 2020 - present

- Completed

- Applied but not funded
    - CUNY - NYC Tech Talent Pipeline Career Success Course Innovation Grant - CMP 426 Course Curriculum Re-Development, $10,000

SERVICE TO DEPARTMENT

- (AY 2021 – 2023) Senator – Lehman College Senate.
- (AY 2020 - 2023) Department of Computer Science Education Policy Committee (EPC) for Computer Information System.
- (AY 2020 - 2023) Co-creating a new graduate certification program and co-designing courses for certifying K-12 teachers to teach computer science (CMP 566, CMP 567, CMP 568, and CMP 569)
- (AY 2018 - 2019) Served as the Co-Advisor for Women in Computer Science (WiCS). Oversaw the initial registration and establishment of the club with the Office of Campus Life, Student Government Association and Association of Computing Machinery (ACM). Worked with the student co-founders to come up with club events such as Tech Create her, and Coding & Beyond panel discussion.
- (AY 2019 - 2020) Served as the Co-Advisor for Women in Computer Science (WiCS).
- (AY 2019 - 2020) Served as the Advisor for the Women in Cybersecurity (WiCyS). Oversaw the initial registration with the Office of Campus Life. Assisted the club in securing $500 seed money from Women in CyberSecurity (WiCyS) national organization for club events. (https://www.wicys.org/).
- Participated in mentoring students at the Lehman Hackathon in the Fall 2018.
- Participated in mentoring students at the Lehman Hackathon in the Fall 2019.
- Co-created a new course (Server-side Web Application Development): Currently awaiting approval.
- Participated in meeting with Major League Hacking (MLH) to discuss channels for establishing a partnership for conducting computer science and computer information systems workshops using their industry-standard-robust resources and platforms.
- (Spring 2019) Participated in Mathematics and Computer Science Learning Center Technical Interview Whiteboarding session.
- Served as a mentor and judge at the Spring 2019 Women in Computer Science Tech Create her event.
- Mentored and advised Computer Science and Computer Information Systems student.

- Revised CMP 420 through the introduction of Project-Based Learning and online quizzes for active learning.
- Secured a grant to improve CMP 420 curriculum
- Served as an Independent Study (CMP 485) course Advisor:
    a. Winter 2020: 3 students
    b. Spring 2020: 2 students
- (AY 2018 – 2019) Served on a student grade appeal committee. Other members included: Gwang Jung and Liang Zhao.
- (Summer 2020) Attended Break Through Tech New York's summer faculty workshop where we discussed NSF's Broadening Participation in Computing (BPC) initiative.
- (AY 2020-2021) Co-applied for in a $60,000 grant from Northwestern Khoury College of Computer Science Center for Inclusive Computing for broadening participation.
- (AY 2020 – 2021) Served on a student grade appeal committee. Other members included: Gwang Jung and Eva Sofianos.
- Advising and supervising students for Independent Studies

SERVICE TO SCHOOL

- (2018 - 2021) Instructor for the Tech Talent Pipeline (TPP) Bootcamp training Program at Lehman College. Introduced more Git and GitHub materials in the lessons to help meet industry standards and adopted the use video resources to help deliver content to students as part of Project-Based-Learning and active learning.
    - https://lehmancuny2x.com/index.html
- (AY 2019 - 2020) Co-wrote a $16,000 tech-fee proposal for two new classroom multimedia lecterns and two teaching computers to be placed in Gillet Hall room 333 and Gillet Hall room 311. This was done in conjunction with Marvin Florin.
- (AY 2020 – 2021) Committee Member Project Exito – Lehman College in partnership with Guttman Community College : Guttman CC to Lehman Four-Year Transfer Maps Meeting
- (2018) Instructor for the Tech Talent Pipeline (TPP) Bootcamp Program at Lehman College
- Co-created a Pre-Bootcamp Program for the TPP Bootcamp to help students improve technical and soft skills prior to joining bootcamp training.
- (2019) Instructor for the Tech Talent Pipeline (TPP) Bootcamp Training Program at Lehman College.
    - http://comet.lehman.cuny.edu/sfulakeza/su19/tpp/index.php
- (AY 2018 - 2020) Lehman College CUNY2X/TPP & Bootcamp Committee.
- (AY 2018 – 2019) NSS & Computer Science Workshops Committee.
- Co-organizer and co-designer/co-facilitator for the Git & GitHub Workshop in the Fall 2018 semester.
- Facilitated technical-career-oriented workshop (AY 2018 - 2019).
- Tech Panelist "A conversation about the gap between academics and the tech industry" (2019).
- Co-facilitated pre-TPP Bootcamp Whiteboarding sessions in the Spring 2019 semester.
- (AY 2017 - 2018).Served as mentor for TPP/CUNY2X students and helped develop a TPP pre-bootcamp program
- (AY 2018 - 2019) Mentored TPP/CUNY2X students.
- (AY 2019 - 2020) Mentored TPP/CUNY2X students.
- Served on the Hiring Committee for the CUNY2X Career Advisor in June 2019.

SERVICE TO LEHMAN COLLEGE

- Served as a college representative to meet members from the Middle States Commission on Higher Education in the immediate past reaccreditation process.
- Served on the Faculty and Staff Success taskforce for the development of 2020 - 2025 Strategic Plan for the college.
- Worked with the International Student and Scholar Office create an animated video that presents services that are provided by the office. Provided technical support at their events
- (AY 2018 - 2019) Co-designed Introduction to Scratch Programming program for the Mount Vernon Steam Academy (MVSA) (middle school and high school students) for the Mount Vernon Pipeline to College Partnerships Program (PCOPP). PCOPP is a grant from New York State Education Department (NYSED) that aims to engage STEM students and faculty from Lehman College and MVSA (AY 2018 - 2019).
    - https://www.mtvernoncsd.org/site/Default.aspx?PageType=3&DomainID=4&PageID=1&ViewID=6446ee88-d30c-497e-9316-3f8874b3e108&FlexDataID=6475
    - http://comet.lehman.cuny.edu/sfulakeza/sp19/scratch/index.php
- (AY 2019 - 2020) Co-designed and facilitated Web Development Programming for the Mount Vernon Steam Academy students through the Mount Vernon Pipeline to College Partnerships Program (PCOPP)
- (AY 2020 - 2021) Facilitated Web Development Programming for the Mount Vernon Steam Academy students through the Mount Vernon Pipeline to College Partnerships Program (PCOPP)

SERVICE TO CUNY

- (AY 2022 – 2023) Served a Faculty Facilitator for the CUNY New Lecturer Initiative
- Served as a faculty mentor to the Tech-in-Residence corps at the 2018 Training Workshop at CUNY Central. The corps are industry professionals hired to teach CS and CIS courses and share their industry experience with our students. Assisted with their syllabi, best practices and pedagogy styles.

SERVICE TO THE PROFESSION

- (AY 2018 – 2019) Participated in Professors Open Source Software Experience (POSSE) Workshops and Humanitarian Free and Open Source Software (HFOSS) Project from the Fall of 2018 to the Winter of 2019
    - http://teachingopensource.org/
    - http://teachingopensource.org/hfoss/
    - http://foss2serve.org/index.php/Main_Page
    - http://foss2serve.org/index.php/POSSE_2019-01_Participants
- Member Association for Computing Machinery (ACM)
- Member Institute of Electrical and Electronics Engineers (IEEE)

COMMUNITY SERVICE

- Served as a judge at the Tri-County Science & Technology Fair (04/27/2019)

TEACHING

A. COURSES TAUGHT

Course Code	Course Title	Semester First Taught	Lehman or GC
CIS 166	Computer Programming for Information Processing I	Fall 2016	Lehman
CIS 244	Introduction to Database Management	Fall 2017	Lehman
CIS 246	E-Commerce	Fall 2015	Lehman
CMP 167	Programming Methods I	Spring 2019	Lehman
CMP 168	Programming Methods II	Fall 2019	Lehman
CMP 326	Programming Methods II	Spring 2019	Lehman
CMP 334	Computer Organization	Summer 2020	Lehman
CMP 405	Introduction to Networks	Fall 2019	Lehman
CMP 420	Database Systems	Summer 2019	Lehman
CMP 426	Operating Systems	Fall 2018	Lehman
CMP 485	Independent Study	Winter 2020	Lehman
CMP 697	Operating Systems	Fall 2018	Lehman
CMP 758	Database Systems	Summer 2020	Lehman
CMP 128	Programming through Web Development	Spring 2021	Lehman
CMP 794	Supervised Internship in Computer Science	Summer 2021	Lehman
CMP 798	Independent Study in Computer Science	Spring 2021	Lehman
LSP 481	Honors Tutorial	Spring 2021	Lehman

B. **<u>STUDENT EVALUATIONS</u> (since last personnel action, in reverse chronological order; excerpts of representative comments may be attached as a separate document)**

Course	**Semester**	**SETL 5.1 mean**	**SETL 5.1 median**	**SETL 5.2 mean**	**SETL 5.2 median**	**Number of responses**
CMP 167-05	Spring 2022	1.25	1	2.25	2	8
CMP 334-01	Spring 2022	1.2	1	1.5	1	10
CMP 420-ZG81	Spring 2022	2.08	1	2.38	2	13
CMP 426-ZI81	Spring 2022	1	1	1.2	1	5
CMP 426-C401	Spring 2022	1.4	1	1.6	1	10
CIS 166-H1LB	Fall 2021	1.8	2	2.21	2	25
CIS 166-H1LC	Fall 2021	1.95	2	2.35	2	20
CMP 128-H01	Fall 2021	1.13	1	1.38	1	8
CMP 405-H81B	Fall 2021	1.2	1	1.1	1	10
CMP 405-H81C	Fall 2021	1.25	1	1.25	1	8
CMP 426-H81	Fall 2021	1.33	1	1.6	1	15
CMP 697-H81	Fall 2021	1	1	1	1	1
CMP 743-H81	Fall 2021	1.5	1.5	1.5	1.5	2
CMP 128-A01	Spring 2021	1.9	1.5	1.7	1	10
CMP 426-A01	Spring 2021	1.64	1	1.82	2	22
CMP 168-A01	Fall 2020	1.44	1	1.53	1	17
CMP 405-A1	Fall 2020	1.46	1	1.59	1	35
CMP 426-A01	Fall 2020	1.54	1	1.68	1	28
CMP 697-A01	Fall 2020	1	1	1	1	2
CMP 426-01	Fall 2019	1.85	1	2.23	2	13
CMP 405-01LC	Fall 2019	1.9	1.5	1.7	1.5	10
CMP 405-01LB	Fall 2019	1.38	1	1.38	1	8

CMP 167-02	Fall 2019	1.7	1	2.1	2	10
CMP 167-01	Fall 2019	1.9	1.5	2.2	2	10
CMP 420-01C	Summer 2019	1.38	1	1.25	1	8
CMP 326-01	Summer 2019	1.4	1	2	2	5
CMP 697-ZG81	Spring 2019	1	1	1	1	3
CMP 426-01	Spring 2019	1.59	1	1.79	1	29
CMP 167-03	Spring 2019	1.67	1.5	1.83	1.5	6
CMP 167-02	Spring 2019	1.33	1	1.67	1.5	6
CMP 697-01	Fall 2018	2	2	3	3	1
CMP 426-01	Fall 2018	1.81	2	2.1	2	21
CMP 326-ZF81	Fall 2018	1.25	1	1.5	1	4
CMP 326-01	Fall 2018	1	1	1.25	1	4
CMP 167-1FLC	Fall 2018	1.89	2	2.05	2	19
CIS 166-81LC	Spring 2018	1.78	2	1.89	2	9
CMP 167-04LC	Spring 2018	2.09	2	2.09	2	11
CMP 326-C401	Spring 2018	2.29	2	2.29	2	7
CMP 326-ZG81	Spring 2018	1.5	1	1.5	1	6

Mingxian Zhong ***Curriculum Vitae***

Contact Information

Department of Computer Science
Lehman College, CUNY
Gillet Hall, Room 200B
250 Bedford Park Blvd W, Bronx, NY 10468
E-mail: mingxian.zhong@lehman.cuny.edu

Research Interests

Graph theory, Scheduling, Algorithms, Combinatorial Optimization.

Current Position

Assistant Professor, Department of Computer Science Lehman College, CUNY.
Member of the Doctoral Faculty, Department of Computer Science The Graduate Center, CUNY.

Education

Columbia University, New York, NY
Ph.D., Operations Research, 2018

- Thesis:"Some problems in Graph Theory and Scheduling"
- Advisors: Maria Chudnovsky and Clifford Stein

M.S., Operations Research, 2013
Wuhan University, Wuhan, China
B.S., Economics and Mathematics, 2011

Journal Articles

- Better 3-coloring algorithms: excluding a triangle and a seven vertex path (with F. Bonomo, M. Chudnovsky, J. Goedgebeur, P. Maceli, O. Schaudt and M. Stein), *Theoretical Computer Science*, *850(2021)*, *98-115*.
- List 3-coloring graphs with no $P_6 + rP_3$ (with M. Chudnovsky, S. Huang and S. Spirkl), *Algorithmica*, *83(2021)*, *216–251*.
- List-three-coloring P_t-free graphs with no induced 1-subdivision of $K_{1,s}$ (with M. Chudnovsky and S. Spirkl), *Discrete Mathematics*, *11(2020)*, *Article 112086*.
- Obstructions for three-coloring graphs without induced paths on six vertices (with M. Chudnovsky, J. Goedgebeur and O. Schaudt), *Journal of Combinatorial Theory*, *Ser. B*, *140 (2020)*, *45-83*.
- Obstructions for three-coloring and list three-coloring H-free graphs (with M. Chudnovsky, J. Goedgebeur and O. Schaudt), *SIAM Journal on Discrete Mathematics*, *34 (2020)*, *431-469*.
- Approximately Coloring Graphs Without Long Induced Paths (with M. Chudnovsky, O. Schaudt, S. Spirkl and M. Stein), *Algorithmica*, *81 (2019)*, *3186-3199*.
- Scheduling When You Don't Know the Number of Machines (with C. Stein), *ACM Transactions on Algorithms*, *November 2019*, *Article No.9*.
- Triangle-free graphs with no six-vertex induced path, (with M. Chudnovsky, P. Seymour, S. Spirkl and M. Zhong) *Discrete Mathematics*, *341 (2018)*, *2179-2196*.
- Three-coloring and list three-coloring graphs without induced paths on seven vertices (with F. Bonomo, M. Chudnovsky, P. Maceli, O. Schaudt and M. Stein), *Combinatorica*, *38 (2018)* , *779-801*.
- 4-coloring P_6-free graphs with no induced 5-cycles (with M. Chudnovsky, P. Maceli and J. Stacho), *Journal of Graph Theory*, *84 (2017)*, *262-285*.

Conference Proceedings

- Complexity of C_k-coloring in hereditary classes of graphs (with M. Chudnovsky, S. Huang, P. Rzazewski and S. Spirkl), *Proceedings of the 27th Annual European Symposium on Algorithms, 2019*.
- Four-coloring P_6-free graphs (with M. Chudnovsky and S. Spirkl), *Proceedings of SODA 2019, 1239-1256*.
- Scheduling When You Don't Know the Number of Machines (with C. Stein), *Proceedings of SODA 2018, 1261-1273*.
- Approximately Coloring Graphs Without Long Induced Paths (with M. Chudnovsky, O. Schaudt, S. Spirkl and M. Stein), *Proceedings of WG 2017, 193-205*.
- Obstructions for three-coloring graphs without induced paths on six vertices (with M. Chudnovsky, J. Goedgebeur and O. Schaudt), *Proceedings of SODA 2016, 17741783*.

Invited Talks

"Complexity of Ck-coloring in Hereditary Classes of Graphs"
• SIAM Conference on Discrete Mathematics, Virtual — July 2021
"Two recent results on coloring graphs with forbidden induced subgraphs"
• BCC Math Seminar, Bronx, NY — May 2019
"Four-coloring P_6-free graphs "

• SODA 2019, San Diego, CA — Jan 2019

• Graph Theory Day 77, Graden City, NY "Coloring graphs with forbidden induced subgraphs" — April 2019

• New York Number Theory Seminar, New York, NY — Nov 2018

• New York Combinatorics Seminar, New York, NY — Nov 2018
"Obstructions for list three-coloring H-free graphs"

• SiGMa 2017, Waterloo, Canada — July 2017

• INFORMS Annual Meeting 2017, Houston, TX — Oct 2017

• SIAM Conference on Discrete Mathematics, Denver, CO — June 2018
"Scheduling When You Don't Know the Number of Machines"

• INFORMS Annual Meeting 2016, Nashville, TN — Nov 2016

• MAPSP 2017, Seeon Abbey, Germany — June 2017

• SODA 2018, New Orleans, LA — Jan 2018
"Three-coloring graphs without induced paths on seven vertices"

• SIWAG 2016, Pogliano, Italy — Sep 2016
"Recent progress on three-coloring graphs without long induced paths"
• AMS Fall Eastern Sectional Meeting, Brunswick, ME — Sep 2016

Teaching Experience

Instructor at Lehman College

- CMP 232 Elementary Discrete Structures & Applications Fall 2018, Spring 2019, Fall 2019, Spring 2020, Fall 2020, Fall 2021, Spring 2022
- CMP 464/CMP 775 Combinarial and Graph Algorithms Spring 2020, Spring 2021, Spring 2022
- CMP 464/CMP 747 Linear Programming and Operations Research Fall 2021

Teaching Assistant at Columbia University

- IEOR 4407: Game Theory Models of Operations Research Fall 2013
- IEOR 4630: Asset Allocation Spring 2014
- IEOR 4403: Quantitative Corporation Finance Fall 2014, 2015, 2016
- IEOR 4106: Intro to OR: Stochastic Models Spring 2015
- IEOR 6614: Optimization II Spring 2016, 2017
- IEOR 3608: Foundations of Optimization Fall 2017
- IEOR 4405: Production Scheduling Spring 2018

Academic Services

Co-organizer of New York Combinatorics Seminar, Spring 2019 - present.
Reviewer for Journals and Conferences including: Journal of Combinatorial Theory, Series B, Journal of Graph Theory, ACM Transactions on Algorithms, Algorithmica, Discrete Mathematics, European Journal of Combinatorics, Journal of Combinatorial Optimization, Journal of Automata, Languages, and Combinatorics Expert Systems
With Applications ACM-SIAM Symposium on Discrete Algorithms (SODA), SIAM Symposium on Simplicity in Algorithms (SOSA), Annual European Symposium on Algorithms (ESA).

Professional Services

EPC member for the major of Computer Information Systems, Spring 2020-present Computer Science Graduate Advisor, Spring 2022 - present.

Grants

PSC-CUNY Award # 63407-00 51

Curriculum Vitae

Name: Sameh Fakhouri **College:** Lehman College

Recommendation For

Appointment: **Promotion:**

Reappointment: x **Reappointment With Tenure:**

Other (Designation as Vice President, Dean, etc.):

Title: Lecturer **Department:** Computer Science

Effective Date: **Salary Rate:**

Higher Education

A. Degrees

Institution	Dates Attended	Degree And Major	Date Conferred
New York University	1/81 – 6/83	M.S. Computer Science	June 1983
Columbia University	9/75 – 6/79	B.S. Electrical Engineering	June 1979

B. Addition Higher Education and/or Education In Progress

21 Graduate credits in Education at Lehman College, towards fulfilling teacher certification requirements 6/06 – 5/10

3 Graduate credits in Special Education at The College of Saint Rose, towards fulfilling teacher certification requirements 10/09

6 Graduate credits in Mathematics at Lehman College, towards fulfilling teacher certification requirements 9/09 – 5/10

Experience

A. Teaching

Intitution	Dates	Rank	Department
Lehman College	9/12 – Present	Lecturer	Mathematics & Computer Science
Lehman College	9/11 – 5/12	Substitute Lecturer	Mathematics & Computer Science
Lehman College	1/11 – 5/11	Adjunct Lecturer	Mathematics & Computer Science
South Bronx Preparatory High School	1/08 – 6/11	Teacher	Mathematics
Theatre Arts Productions Company High School	9/07 – 1/08	Teacher	Mathematics
Essex Street Academy	9/06 – 6/07	Teacher	Mathematics
Iona College	9/88 – 6/89	Adjunct Lecturer	Computer & Information Systems

B. Other

Institution	Dates	Rank	Department
IBM T.J. Watson Research Center	10/89 – 6/05	Advisory Software Engineer	Advanced Distributed Systems
IBM DSD/TPF	9/84 – 9/89	Staff Software Engineer	DSD/TPF
Logos General Systems	6/83 – 8/84	Manager Engineering Systems	Development
Nicolet Scientific	6/79 – 7/82	Electronics Engineer	Development

Academic and Professional Honors:

Publications (Last Five Years Only):

Ph.D. Thesis Title:

Publications (Prior to Last Five Years):

A. Papers

Eilam, T., Appleby, K., Breh, J., Breiter, G., Daur, H., Fakhouri, S., Hunt, G., Lu, T., Mummert, L., Miller, S., Pershing, J., Wagner, H. ***Using a Computing Utility Framework to Develop Utility Systems***. IBM Systems Journal, Volume 43, Number 1, 2004.

Fakhouri, S., Goldszmidt, G., Gupta, I., Kalantar, M., Pershing, J. ***GulfStream - A System for Dynamic Topology Management in Multi-Domain Server Farms*** in *IEEE International Conference on Cluster Computing (Cluster 2001)*, October 2001

Appleby, K. Fakhouri, S. Fong, L. Goldszmidt, G. Kalantar, M. Krishnakumar, S. Pazel, D.P. Pershing, J. Rochwerger, B. ***Océano SLA based management of a computing utility***. Proceedings of 2001 International Symposium on Integrated Network Management. Page 14-18. May 2001

Sameh Fakhouri, William F. Jerome, Vijay K. Naik, Ajay Raina, Pradeep Varma: ***Active Middleware Services in a Decision Support System for Managing Highly Available Distributed Resources***. Middleware 2000: 349-371

Farnam Jahanian, Sameh Fakhouri, Ragunathan Rajkumar***: Processor Group Membership Protocols: Specification, Design, and Implementation***. Symposium on Reliable Distributed Systems 1993: 2-11

B. Patents

Fakhouri, S., Jerome, W., Kummamuru, K., Naik; V., Pershing, Jr., J., Raina, A., Varma, P., Badovinatz, P., Kumar, V. **MANAGING A CLUSTER OF NETWORKED RESOURCES AND RESOURCE GROUPS USING RULE-BASED CONSTRAINTS IN A SCALABLE CLUSTERING ENVIRONMENT**, US 7,464,147. Issued December 9, 2008.

Knop, F., Jurgensen, D., Chang, C.S., Fakhouri, S., Race, T. **TOPOLOGY PROPAGATION IN A DISTRIBUTED COMPUTING ENVIRONMENT WITH NO TOPOLOGY MESSAGE TRAFFIC IN STEADY STATE**, US 6,885,644 B1. Issued April 26, 2005

Chandra, T., Fakhouri, S., Fong, L., Jerome, W., Krishnakumar, S., Naik, V., Pershing, J., Turek, J. **CONTROLLING A NUMBER OF INSTANCES OF AN APPLICATION RUNNING IN A COMPUTING ENVIRONMENT**, US 6,782,408 B1. Issued August 24, 2004

Chandra, T., Fakhouri, S., Fong, L., Jerome, W., Krishnakumar, S., Naik, V., Pershing, J. **METHOD, SYSTEM AND PROGRAM PRODUCTS FOR MANAGING THE CHECKPOINTING/RESTARTING OF RESOURCES OF A COMPUTING ENVIRONMENT**, US 6,594,779 B1. Issued July 15, 2003

Unpublished Work (Supported by Evidence):

a. Works Accepted For Publication

b. Works Submitted For Publication

c. Work In Progress

Grants Received

a. Multiple

b. Individual

Service To The College

1. Lehman College Technology Fee Committee 2012-2023.
2. Department Of Mathematics & Computer Science Education Policy Committee (EPC) 2013-2014.
3. Department Of Mathematics & Computer Science Education Policy Committee (EPC) 2014-2015.
4. Department Of Mathematics & Computer Science Education Policy Committee (EPC) 2015-2016.
5. School Curriculum Committee (SCC), School of Natural and Social Sciences. Spring 2015.
6. Served as course coordinator for CIS-166 & CMP-230 Spring 2013.
7. Served as course coordinator for CIS-166 & CMP-230 Fall 2013.
8. Served as the internship coordinator:
    - **a.** Spring 2013 – 18 students.
    - **b.** Summer 2013 – 1 student.
    - **c.** Fall 2013 – 15 students.
    - **d.** Spring 2014 – 8 Students.
    - **e.** Fall 2014 – 19 Students
    - **f.** Spring 2015 – 17 Students
    - **g.** Fall 2015 – 10 Students
    - **h.** Spring 2016 – 30 Students
    - **i.** Fall 2016 – 19 Students
    - **j.** Spring 2017 –13 Students
    - **k.** Fall 2017 –34 Students
    - **l.** Spring 2018 –23 Students
    - **m.** Fall 2018 –33 Students
    - **n.** Spring 2019 –18 Students
    - o. Summer 2019 – 2 Students
    - **p.** Fall 2019 –45 Students
    - **q.** Spring 2020 - 36 Students
    - **r.** Fall 2020 –6 Students
    - **s.** Spring 2021 – 11 Students
    - **t.** Fall 2021 –24 Students
    - **u.** Spring 2022 – 13 Students
9. Created a new course: iOS Programming. Is currently being offered as CMP-430.
10. Worked on developing a relationship with IBM/TPF in Poughkeepsie, NY. This relationship should provide Lehman with a partner for placing CMP and, possibly, CIS majors in internships at IBM/TPF.
11. Participated in meeting with IBM/TPF to map out a course on mainframe computing to be offered in Spring 2015.
12. Participating in a committee that is looking into how Lehman College can be involved to help NYC DOE schools be better prepared to teach Computer Science courses. The committee consists of VP Ronald Bergmann, Dean Harriet Fayne, Sandra Lerner, Robert Schneider, Loric Madramoottoo and myself.
13. (AY 2014-2015) Served as Co-Advisor to the Lehman Chapter of The National Society of Black Engineers. Oversaw the planning of the Lehman Hackathon that was held on campus in the Fall 2014 semester.
14. (AY 2015-2016)Served as Advisor to the Lehman Chapter of the National Society of Black Engineers. Oversaw the planning of the Lehman Hackathon that was held on campus in the Fall 2015 Semester.
15. Established a relationship between our department and Workforce Opportunity Services. WOS "is a 501(c)(3) nonprofit charitable organization that helps companies build strong pipelines through unique work-study programs". http://www.wforce.org. WOS will be conducting professional development seminars for our students and they are interested in helping our Computer Information Systems and Computer Science majors find employment within their field of study.
16. Established a relationship between our department and Code To Work. Code To Work "develops partnerships with companies experiencing a shortage of diverse, qualified candidates for entry and mid-level technical jobs. http://www.codetowork.org.

Service To The University

Conferences Attended

U.S. News STEM Solutions – Austin, Texas 6/2013

National Society Of Black Engineers – Boston, Massachusetts 3/2016

Community Service

1. President of Friends of Glenwood Lake Park (FGLP). A 501-C3 that fundraises and advocates for enhancing the neighborhood lake and park.
2. Vice President of Glenwood Lake Neighborhood Association (2015-2016).
3. Member of Glenwood Lake Neighborhood Association and Chair of the Communications Committee in the association.

Membership In Professional Societies (Last Five Years Only)

1. National Council of Teachers of Mathematics
2. Association of Computing Machinery

References (List name, title and affiliation only)

Fong, Liana	Manager. IBM T.J. Watson Research Center, 19 Skyline Drive, Hawthorne, NY 10532. Phone: (914)-784-7269. E-mail: llfong@us.ibm.com
Schnneider, Robert	Chairman, Department of Mathematics & Computer Science, Herbert H. Lehman College, 250 Bedford Park Boulevard West, Bronx, NY 10468. Phone: (718)-960-8870. E-mail: robert.schneider@lehman.cuny.edu
Rothchild, David	Professor, Department of Mathematics & Computer Science, Herbert H. Lehman College, 250 Bedford Park Boulevard West, Bronx, NY 10468. Phone: (718)-960-8784. E-mail: david.rothchild@lehman.cuny.edu.
Taback, Stanley	Professor, Middle and High School Education, Lehman College, 250 Bedford Park Boulevard West CA-B29D, Bronx, NY 10468. Phone: (718)-960-8358. E-mail: stanley.taback@lehman.cuny.edu.

Quiros, Terry School Counselor/College Advisor, South Bronx Preparatory, 360 East 145th Street, Bronx, NY 10454. Phone: (914)-426-0715, E-mail: quirosterry@optonline.net.

LIANG ZHAO

liang.zhao1@lehman.cuny.edu

EDUCATION

Graduate Center, City University of New York *2008 - 2017*

Ph.D. in Mathematics

· Research interest: Computation with structured matrices, randomly-generated matrices, and their application to PDEs and Machine Learning

· Dissertation: *Fast Algorithms on Random matrices and Structured Matrices* · Adviser: Victor Pan

Tsinghua University *2004 - 2008*

B.S. in Mathematics

· Graduation with honors, GPA: 3.70 (8/91 of Class 2004)

· Undergraduate thesis: *Transcendence of hypergeometric functions over function fields* · Advisor: Jia-Yan Yao

EXPERIENCE

Lehman College, City University of New York August 2017 - present

Assistant Professor *New York, NY*

· Research focus: Efficient numerical algorithms in Computational Algebra, Data Mining, and Artificial Intelligence.

· Courses taught: Discrete Structures & Applications in Computer Science, Programming Methods I & II, Artificial Intelligence, Machine Learning, Randomized Algorithms for Matrices, Data Handling and Analysis.

· Advises undergraduate students on the algorithm design and software implementation of image classifier and text classifier based on neural network model.

Graduate Center, City University of New York February 2014 - June 2017

Research Assistant *New York, NY*

· Developed novel state-of-the-art computational algorithms for polynomial root isolation, low-rank matrix approximation, and various optimization problems.

· Implemented algorithms and test their efficiency with application-based data input.

· Organized student seminar on algebraic and numerical computations, inviting Mathametics or Computer Science researchers to present their recent work or studies.

City University of New York September 2010 - December 2016

Adjunct Instructor *New York, NY*

· Courses taught: Pre-Calculus, Calculus, Statistics, Quantitative Reasoning, College Algebra, Mathematics and the Contemporary World.

· Designed writing-intensive syllabus for undergraduate mathematical courses. Students had provided highly positive feedback for being able to gain in-context understanding and conceptual thinking through writing activities.

· Advised students to participate in the Police Reforming Organizing Project by collecting court-monitoring data and obtained statistical evidence on the effectiveness of broken-window policing.

PUBLICATIONS

Qi Luan, Liang Zhao, "Efficient CUR Matrix Decomposition via Relative-Error Double-Sided Least Squares Solving." 32th International Conference on Tools with Artificial Intelligence, 2020.

Victor Y. Pan, Qi Luan, John Svadlenka, Liang Zhao, "Sublinear Cost Low Rank Approximation via Subspace Sampling." International Conference on Mathematical Aspects of Computer and Information Sciences. Springer, Cham, 2019.

Yanzhi Wang, Zheng Zhan, Liang Zhao, Jian Tang, Siyue Wang, Jiayu Li, Bo Yuan, Wujie Wen, Xue Lin, "Universal Approximation Property and Equivalence of Stochastic Computing-based Neural Networks and Binary Neural Networks." Proceedings of AAAI Conference on Artificial Intelligence, 2019.

Siyu Liao, Zhe Li, Liang Zhao, Qinru Qiu, Yanzhi Wang, Bo Yuan, "CircConv: A Structured Convolution with Low Complexity." Proceedings of AAAI Conference on Artificial Intelligence, 2019.

Matthew P. Johnson, Liang Zhao, and Supriyo Chakraborty. "Efficiently Computing Messages that Reveal Selected Inferences While Protecting Others." IEEE Journal of Selected Topics in Signal Processing (2018).

V. Y. Pan, F. Soleymani, and L. Zhao. "An efficient computation of generalized inverse of a matrix." Applied Mathematics and Computation 316 (2018): 89-101.

Matthew P. Johnson, Liang Zhao, and Supriyo Chakraborty. "Computing messages that reveal selected inferences while protecting others." Information Theory Workshop (ITW), 2017 IEEE. IEEE, 2017.

Sammel Bald, Matthew P. Johnson, Levi Rybalov, and Liang Zhao. "Optimizing Regional Power Production Under Thermal Pollution Constraints." *The International Green and Sustainable Computing Conference*, 2017.

Matthew P. Johnson, Liang Zhao, and Supriyo Chakraborty. "Revealing One Thing Without Revealing Another." *The IEEE Information Theory Workshop*, 2017.

Liang Zhao, Siyu Liao, Yanzhi Wang, Jian Tang, and Bo Yuan. "Theoretical Properties for Neural Networks with Weight Matrices of Low Displacement Rank." *The 34th International Conference on Machine Learning*, 2017.

Victor Y. Pan, Liang Zhao. "Real Polynomial Root-finding by Means of Matrix and Polynomial Iterations". *Theoretical Computer Science, Special Issue on Symbolic–Numerical Algorithms (Stephen Watt, Jan Verschelde, and Lihong Zhi, editors)*, in print. DOI: 10.1016/j.tcs.2017.03.032. Also available at arXiv/1501.05390 (2015)

V. Y. Pan, L. Zhao, "Numerically Safe Gaussian Elimination with No Pivoting", *Linear Algebra Appl.*, 2017. http://dx.doi.org/10.1016/j.laa.2017.04.007. Also available at arXiv 1501.05385 CS, (27 pages, 7 figures).

Victor Y. Pan, Liang Zhao. "New Studies of Randomized Augmentation and Additive Preprocessing", *Linear Algebra Appl.*, 512, 252-305 (2017). Available at http://dx.doi.org/10.1016/j.laa.2016.09.035 and arxiv 1412.5864 [math.NA] (38 pages), December 18, 2014.

Victor Y. Pan, Liang Zhao, "Low-rank approximation of a matrix: novel insights, new progress, and extensions", *Proc. of the Eleventh International Computer Science Symposium in Russia* (CSR 2016), (Alexander Kulikov and Gerhard Woeginger, editors), St. Petersburg, Russia, June 2016, Lecture Notes in Computer Science (LNCS), Volume 9691, 352–366, Springer International Publishing, Switzerland.

Victor Y. Pan, Liang Zhao, "Polynomial Real Root Isolation by Means of Root Radii Approximation", *Proceedings of the 17th International Workshop on Computer Algebra in Scientific Computing(CASC 2015)*, (V. P. Gerdt, V. Koepf, and E. V. Vorozhtsov, editors), Lecture Notes in Computer Science, 9301, 347– 358, Springer, Heidelberg, 2015.

Victor Y. Pan, Liang Zhao. "Randomized Circulant and Gaussian Preprocessing", *Proceedings of the 17th International Workshop on Computer Algebra in Scientific Computing(CASC 2015)*, (V. P. Gerdt, V. Koepf, and E. V. Vorozhtsov, editors), Lecture Notes in Computer Science, 9301, 347– 358, Springer, Heidelberg, 2015.

Victor Y. Pan, John Svadlenka, and Liang Zhao. "Estimating the Norms of Circulant and Toeplitz Random Matrices and Their Inverses", *Linear Algebra and Its Applications*, 468, 197–210, 2015.

Dinesh S. Thakur, Zhi-Ying Wen, Jia-Yan Yao, and Liang Zhao. "Transcendence in positive characteristic and special values of hypergeometric functions", *Journal fur die reine und angewandte Mathematik*, 2011.

INVITED TALKS & PRESENTATIONS

City University of New York — May 2018

Computer Science Start-of-the-Year Event — *New York, NY*

· "Neural Networks with Structured Matrices"

City University of New York — April 2017

Joint CUNY Graduate Center-Courant Seminar in Symbolic-Numeric Computing New York, NY ·

"Fast Algorithms on Random Matrices and Structured Matrices"

Pennsylvania State University — March 2017

CCMA Seminar on Deep Learning — *State College, PA*

· "Model Compression of Deep Neural Network via Structured Matrices"

City University of New York - Graduate Center — November 2016

26th Fall Workshop on Computational Geometry — *New York, NY*

· "Numerically Stable Winding Number Algorithm and Its Application to Polynomial Root-Finding"

Purdue University — November 2016

International Workshop on Fast Direct Solver, 2016 — *West Lafayette, IN*

· "Low-rank Approximation of a Matrix: Novel Insights and New Progress"

Aachen University — September 2015

The 18th International Workshop on Computer Algebra in Scientific Computing (CASC 2015)
Aachen, Germany

· "Polynomial Real-Root Isolation by Means of Root-Radii Approximation"

· "Randomized Gaussian Circulant Preprocessing"

RESEARCH GRANTS

PSC-CUNY Award Cycle 51 ($6,000)	2020
PSC-CUNY Award Cycle 50 ($3,500)	2019

FELLOWSHIPS & AWARDS

CUNY Student Research Grant	2013

CUNY Chancellor Research Fellowship	2008-2012
New World Mathematics Award for Undergraduate Thesis	2011
Tsinghua Academic Excellence Scholarship	2005-2008
Honorable Mention Award in ICM2007 (Interdisciplinary Contest in Modeling)	2007

SERVICES

Departmental P&B Committee	2020-present
Ph.D. Program Admissiom Committee	2019-2020
CIS Curriculum Committee	2017-2020
New Faculty Member Search Committee	2017-2018

SKILLS

Languages	Mandarin (native language), English (fluent), French (reading)
Computer Languages	Matlab, Python, C++, Java
Statistical Software	R, SPSS, SAS
Database	MySQL
Deep Learning Tools	PyTorch, Tensorflow

PROFESSIONAL MEMBERSHIP

Institute of Electrical and Electronics Engineers (IEEE)
Association for Computer Machinery (ACM)

Matthew P. Johnson

Department of Mathematics and Computer Science – Lehman College, CUNY
250 Bedford Park Boulevard West – Bronx, NY 10468
Ph.D. Program in Computer Science – The CUNY Graduate Center
250 Bedford Park Boulevard West – New York, NY 10020

T 917-648-9706 • B mpjohnson@gmail.com • Í comet.lehman.cuny.edu/mjohnson

Research Interests

Applications of algorithms, optimization, and mathematical programming, including guaranteed approximations and algorithmic game theory, to real-world and/or beautiful problems.

Application areas include: energy and smart grids, security, networking, sensor networks.

Education

The City University of New York New York, NY
Ph.D. in Computer Science *2005–2010*
Thesis: "Resource Allocation Under Constraints" (advisor: Amotz Bar-Noy)

Columbia University New York, NY *M.S. in Computer Science* *2001–2002*

Columbia University New York, NY *B.S. in Computer Science* *1998–2001*

Lawrence University Appleton, WI
B.A. in Philosophy and Mathematics (double major) *1995–1998*

Positions

Dept. of Math and Computer Science, Lehman College, CUNY Bronx, NY
Assistant Professor *August 2013–present*

Ph.D. Program in Computer Science, The Graduate Center, CUNY New York, NY
Member of the Doctoral Faculty *December 2013–present*
Advanced Science Research Center, CUNY New York, NY
Affiliate Faculty Member *June 2016–present*
EE Dept., University of California, Los Angeles Los Angeles, CA
Postdoctoral Researcher (host: Mani Srivastava) *April 2012–May, 2013*
CS Dept., University of Southern California Los Angeles, CA
Visiting Researcher *October 2011–March 2012*

Weizmann Institute Rehovot, Israel
Visiting Researcher (host: David Peleg) *August 2011*
CSE Dept., Pennsylvania State University Cambridge, MA
Postdoctoral Researcher (host: Tom La Porta) *September 2010–October 2011*
(based at Raytheon BBN Technologies)
Los Alamos National Laboratory Los Alamos, NM
Visiting Researcher *Summer 2010*
US Army Research Laboratory Adelphi, MD
Research Intern *Summers 2008 and 2009*
IBM T.J. Watson Research Center Hawthorne, NY
Research Intern *Summer 2007*
CS Ph.D. Program, City University of New York New York, NY
Research Assistant *Fall 2005–Summer 2010*
City College of New York, CS Department New York, NY
Adjunct Lecturer *2005–2006*
NYU, Stern School of Business, IOMS Department New York, NY
Adjunct Assistant Professor *2004–2008*
CUNY Institute for Software Development & Design New York, NY
Instructor *2004–2005*
Columbia University, CS Department New York, NY
Instructor *2002–2004*

Grants & Honors

Total external funding (personal share) since 2013: $338,100
Total internal funding (personal share) since 2013: $146,000

{ PSC-CUNY Research Award 69778-00 47, $6,000, 2016–2017

{ CUNY Junior Faculty Research Award in Science and Engineering (sponsored by the Sloan Foundation), $50,000, 2016

{ Travel award for my students to attend FWCG, $700, 2015 { NSF grant "INSPIRE: Optimization Algorithms for Regional Thermoelectric Power Generation with Nonlinear Interference" (CCF-1547205), $316,000, 2015-2018

{ Travel award for Mathematics in Data Science workshop, Institute for Computational and Experimental Research in Mathematics, Brown University $600, 2015

{ CUNY ASRC Joint Seed Program Grant, $10,000, 2015-2016

{ PSC-CUNY Research Award 68818-00 46, $6,000, 2015–2016

{ Travel award for NSF CISE CAREER Proposal Writing Workshop, $800, 2015

{ Research in the Classroom Idea Grant, $7,500, 2014–2015 (acceptance rate: 6/73)

- CUNY Collaborative Incentive Research Grant (CIRG 21) 2153 (with co-PI Felisa Vásquez-Abad), $30,000, 2014–2015
- PSC-CUNY Research Award 67665-00 45, $6,000, 2014–2015
- NSF REU supplement to grant DUE-1060598, $20,000, 2013-2014
- Startup funding, Lehman College, $50,000, 2013-2016
- US-Israel BSF Rahamimoff Travel Grant for Young Scientists, $4,000, 2011
- Conference travel grants: IEEE NetSciCom (2011), IEEE MASS (2008), ALGO (2008)
- CUNY Sponsored Dissertation Fellowship (success rate ≈15%), 2009–2010
- Various co-written sub-proposals ("white papers") within the ITA and CTA projects, 2008–2011
- Doctoral Student Research Grant, CUNY, 2007 and 2009
- Zalk Student Travel and Research, CUNY, 2006, 2007, 2008, 2009
- CUNY Science Fellowship, 2005–2010

Score of 99.75/100 (average score: 67/100) on Qualifying Exam (rank: 1 out of 25), 2006

- Master's MS-TA Fellowship, Columbia, 2001-2002

Publications

Under Review / In Preparation

- **[M8]** G. Figueiredo and M.P. Johnson. Separating boxes with boxes. Manuscript, 2015.
- **[M7]** H. Chakraborty and M.P. Johnson. Visualizing sexual violence location data and predicting whether crime reports are filed. Manuscript, 2015.
- **[M6]** I. Feigenbaum and M.P. Johnson. Selfish knapsack. Manuscript, 2015.
- **[M5]** E. Arkin, J. Gao, M.P. Johnson, and J. Zeng. Geometric *r*-gathering. Manuscript, 2015.
- **[M4]** S. Chakraborty, N. Bitouzé, M. Srivastava, L. Dolecek, and M.P. Johnson. Protecting data against unwanted inferences. Manuscript, 2015.
- **[M2]** S. Bald, M.P. Johnson, and O. Liu. Approximating the maximum rectilinear crossing number. Submitted, 2015.
- **[M1]** M.P. Johnson, B. Phelan, A. Bar-Noy, P. Basu, and R. Ramanathan. Minimum-cost broadcast over reliable link-layer multicast. Submitted, 2014.

Journals

- **[J15]** S. Chechik, M.P. Johnson, M. Parter, and D. Peleg. Secluded connectivity problems. Minor revision, *Algorithmica*, 2015.
- **[J14]** M.P. Johnson and A. Bar-Noy. Pan and scan: configuring cameras for coverage. Under revision, *Theoretical Computer Science*, 2015.

{ **[J13]** Y. Alayev, A. Bar-Noy, M.P. Johnson, L. Kaplan, and T. La Porta. You can't get there from here: sensor scheduling with refocusing delays. *Wireless Networks*, 21(5), July 2015.

{ **[J12]** M.P. Johnson, F. Fang, and Y. Gai. Distributing patrols to maximize pristine forest area. Minor revision, *Sustainable Computing*, 2014.

{ **[J11]** M.P. Johnson, A. Gutfraind, and K. Ahmadizadeh. Evader interdiction: algorithms, complexity and collateral damage. *Annals of Operations Research*, November 2014.

{ **[J10]** Y. Alayev, F. Chen, Y. Hou, M.P. Johnson, A. Bar-Noy, T. La Porta, K.K. Leung. Throughput maximization in mobile WSN scheduling with power control and rate selection. *IEEE Transactions on Wireless Communication*, 13(7), July 2014.

{ **[J9]** W. Ren, Q. Zhao, R. Ramanathan, J. Gao, A. Swami, A. Bar-Noy, M.P. Johnson, and P. Basu.

Broadcasting in multi-radio multi-channel wireless networks using simplicial complexes. *Wireless Networks*, 19(6), August 2013

{ **[J8]** F. Chen, M.P. Johnson, Y. Alayev, A. Bar-Noy, and T. La Porta. Who, when, where: timeslot assignment to mobile clients. *IEEE Transactions on Mobile Computing*, 11(1), January 2012.

{ **[J7]** M.P. Johnson, D. Sarıöz, A. Bar-Noy, T. Brown, D. Verma, and C.W. Wu. More is more: robust and efficient coverage in denser sensor deployment. *IEEE Transactions on Sensor Networks*, 8(3), 2012.

{ **[J6]** F. Chen, M.P. Johnson, A. Bar-Noy, and T. La Porta. Proactive data dissemination to mission sites. *Wireless Networks*, 18(7), 2012.

{ **[J5]** M.P. Johnson, A. Bar-Noy, O. Liu, and Y. Feng. Energy peak shaving with local storage. *Sustainable Computing*, 1(3), 2011.

[J4] M.P. Johnson, H. Rowaihy, D. Pizzocaro, A. Bar-Noy, S. Chalmers, T. La Porta and A. Preece. Sensor-mission assignment in constrained environments. *IEEE Transactions on Parallel and Distributed Systems*, 2(11), 2010.

{ **[J3]** H. Rowaihy, M.P. Johnson, O. Liu, A. Bar-Noy, T. Brown, and T. La Porta. Sensor-mission assignment in wireless sensor networks. *ACM Transactions on Sensor Networks*, 6(4), 2010.

∗ *Following convention, publications appearing in TCS venues order author names alphabetically.*

{ **[J2]** S. Eswaran, M.P. Johnson, A. Misra, and T. La Porta. Distributed utility-based rate adaptation protocols for prioritized, quasi-elastic flows. *Mobile Computing and Communications Review*, 13(1), 2009.

{ **[J1]** M.P. Johnson and R. Parikh. Probabilistic conditionals are almost monotonic. *Review of Symbolic Logic*, 1(1), 2008.

Conferences

{ **[C37]** P. Basu, C.-K. Chau, A. Bejan, R. Gibbens, S. Guha, and M.P. Johnson. Efficient multicast in hybrid wireless networks. *MILCOM 2015*, Tampa, FL, October 2015.

{ **[C36]** S. Chakraborty, M.P. Johnson, and M. Srivastava. Generalizing information bottleneck technique for privacy of inferences. *Annual Fall Meeting of the ITA*, College Park, MD, September 2015.

{ **[C35]** A. Bar-Noy, M. P. Johnson, N. Naghibolhosseini, D. Rawitz, and S. Shamoun. The Price of Incorrectly Aggregating Coverage Values in Sensor Selection. *DCOSS 2015*, Fortaleza, Brazil, June 2015.

{ **[C34]** P. Basu, Feng Yu, M.P. Johnson, and A. Bar-Noy. Low Expected Latency Routing in Dynamic Networks. *IEEE MASS 2014*, Philadelphia, PA, October 2014.

{ **[C33]** S. Chakraborty, F. Cerutti, M.P. Johnson, M. Srivastava. Understanding the information flow dynamics in coalition networks: a game-theoretic analysis. *ITA Fall Meeting 2014*, Cardiff, UK, September 2014.

{ **[C32]** M.P. Johnson and D. Sarıöz. Representing a Planar Straight-Line Graph Using Few Obstacles. *CCCG 2014*, Halifax, Nova Scotia, August 2014.

{ **[C31]** M.P. Johnson, O. Liu, and G. Rabanca. Secluded Path via Shortest Path. *SIROCCO 2014*, Hida Takayama, Japan, July 2014.

{ **[C30]** F. Vázquez-Abad, P. Dashevsky, and M.P. Johnson LP-Based Approaches to StationaryConstrained Markov Decision Problems. *WODES 2014*, Paris, France, May 2014.

{ **[C28]** N. Hu, M.P. Johnson, D. Pizzocaro, T. La Porta, and A.D. Preece Resource allocation with non-deterministic demands and profits. *MASS 2013*, Hangzhou, China, October 2013.

{ **[C27]** P. Nain, D. Towsley, M.P. Johnson, F. Yu, P. Basu, and A. Bar-Noy. Traversal times on Markovian paths. *CHANTS 2013*, Miami, FL, September 2013.

{ **[C26]** S. Chechik, M.P. Johnson, M. Parter, and D. Peleg. Secluded connectivity problems. *ESA 2013*, Sophia Antipolis, France, September 2013.

{ **[C29]** S. Chakraborty, K.R. Raghavan, M.P. Johnson, and M.B. Srivastava. A framework for context-aware privacy of sensor data on mobile systems. *HotMobile 2013*, Jekyll Island, GA, February 2013.

{ **[C25]** A. Bar-Noy, Y. Gai, M.P. Johnson, B. Krishnamachar, and G. Rabanca. Funding games: the truth but not the whole truth. *WINE 2012*, Liverpool, UK, December 2012.

{ **[C24]** Zhengyu Yin, Albert Xin Jiang, Matthew P. Johnson, Christopher Kiekintveld, Kevin LeytonBrown, Tuomas Sandholm, Milind Tambe, John P. Sullivan. TRUSTS: Scheduling Randomized Patrols for Fare Inspection in Transit Systems. *IAAI 2002*, Toronto, Canada, July 2012.

[C23] M.P. Johnson, F. Feng, M. Tambe. Patrolling to maximize pristine forest area. *AAAI 2012*, Toronto, Canada, July 2012.

{ **[C22]** A. Xin, Jiang, Z. Yin, M.P. Johnson, M. Tambe, C. Kietkintveld, K. Leyton-Brown, and T. Sandholm. Towards optimal patrol strategies for urban security in transit systems. *AAAI 2012*, Toronto, Canada, July 2012.

{ **[C21]** A. Bar-Noy, F. Chen, M.P. Johnson, and T. La Porta. Convergecast with aggregatable data classes. *IEEE SECON 2012*, Seoul, Korea, June 2012.

{ **[C20]** Y. Alayev, F. Chen, Y. Hou, M.P. Johnson, A. Bar-Noy, T. La Porta, K.K. Leung. Throughput maximization in mobile WSN scheduling with power control and rate selection. *DCOSS 2012*, Hangzhou, China, May 2012.

{ **[C19]** W. Ren, Q. Zhao, R. Ramanathan, J. Gao, A. Swami, A. Bar-Noy, M.P. Johnson, and P. Basu. Broadcasting in multi-radio multi-channel wireless networks using simplicial complexes. *IEEE MASS 2011* (acceptance rate: 19.1%), Valencia, Spain, October 2011.

{ **[C18]** A. Bar-Noy, P. Basu, M.P. Johnson, and R. Ramanathan. Minimum-cost broadcast through varying-size neighborcast. *ALGOSENSORS 2011*, Saarbrüecken, Germany, September 2011.

{ **[C17]** M.P. Johnson and A. Gutfraind. Evader interdiction and collateral damage. *ALGOSENSORS 2011*, Saarbrüecken, Germany, September 2011.

{ **[C16]** M.P. Johnson and A. Bar-Noy. Pan and scan: configuring cameras for coverage. *IEEE INFOCOM 2011* (acceptance rate: 15.96%), Shanghai, China, April 2011.

{ **[C15]** Y. Alayev, A. Bar-Noy, M.P. Johnson, L. Kaplan, and T. La Porta. You can't get there from here: sensor scheduling with refocusing delays. *IEEE MASS 2010* (acceptance rate: 27.6%), San Francisco, CA, November 2010.

{ **[C14]** F. Chen, M.P. Johnson, Y. Alayev, A. Bar-Noy, and T. La Porta. Who, when, where: timeslot assignment to mobile clients. *IEEE MASS 2009* (acceptance rate: 25.4%), Macau SAR, China, October 2009.

{ **[C13]** F. Chen, M.P. Johnson, A. Bar-Noy, I. Fermin, and T. La Porta. Proactive data dissemination to mission sites. *IEEE SECON 2009* (accept rate: 18.8%), Rome, Italy, June 2009.

{ **[C12]** A. Bar-Noy, T. Brown, M.P. Johnson, and O. Liu. Cheap or flexible sensor coverage. *DCOSS 2009* (acceptance rate: 22.4%), Marina Del Rey, CA, June 2009.

{ **[C11]** H. Rowaihy, M.P. Johnson, D. Pizzocaro, A. Bar-Noy, L. Kaplan, T. La Porta, and A. Preece. Detection and localization sensor assignment with exact and fuzzy locations. *DCOSS 2009* (acceptance rate: 22.4%), Marina Del Rey, CA, June 2009.

{ **[C10]** S. Eswaran, M.P. Johnson, A. Misra, and T. La Porta. Adaptive in-network processing for bandwidth and energy constrained mission-oriented multi-hop wireless networks. *DCOSS 2009* (acceptance rate: 22.4%), Marina Del Rey, CA, June 2009.

{ **[C9]** M.P. Johnson, D. Sarıöz, A. Bar-Noy, T. Brown, D. Verma, and C.W. Wu. More is more: robust and efficient coverage in denser sensor deployment. *IEEE INFOCOM 2009* (acceptance rate: 19.7%), Rio de Janeiro, Brazil, April 2009.

{ **[C8]** H. Rowaihy, M.P. Johnson, T. Brown, A. Bar-Noy, and T. La Porta. Assigning sensors to competing missions. *IEEE GLOBECOM 2008* (accept rate: 36.8%), New Orleans, LA, Dec. 2008.

{ **[C7]** M. Gomez, A. Preece, M.P. Johnson, G. de Mel, W. Vasconcelos, C. Gibson, A. Bar-Noy, K. Borowiecki, T. La Porta, D. Pizzocaro, H. Rowaihy, G. Pearson, and T. Pham. An ontology-centric approach to sensor-mission assignment. *EKAW 2008* (acceptance rate for full papers: 16.3%), Sicily, Italy, October 2008.

[C6] A. Bar-Noy, M.P. Johnson, and O. Liu. Peak shaving through resource buffering. *WAOA 2008*, Karlsruhe, Germany, September 2008.

{ **[C5]** M.P. Johnson, H. Rowaihy, D. Pizzocaro, A. Bar-Noy, S. Chalmers, T. La Porta, and A. Preece. Frugal sensor assignment. *DCOSS 2008* (acceptance rate: 25%), Santorini, Greece, June 2008.

{ **[C4]** A. Bar-Noy, M.P. Johnson, Y. Feng, and O. Liu. When to reap and when to sow: lowering peak usage with realistic batteries. *WEA 2008*, Provincetown, MA, May 2008.

{ **[C3]** A. Bar-Noy, T. Brown, M.P. Johnson, T. La Porta, O. Liu, and H. Rowaihy. Assigning sensors to missions with demands. *ALGOSENSORS 2007*, Wroclaw, Poland, July 2007.

{ **[C2]** M.P. Johnson and A.P. Kosoresow. Finding worst-case instances, and lower bounds, for NP-complete problems using genetic algorithms. *SEAL 2002*, Singapore, December 2002.

{ **[C1]** A.P. Kosoresow and M.P. Johnson. Finding worst-case instances of, and lower bounds for, online algorithms using genetic algorithms. *AI 2002*, Canberra, Australia, December 2002.

Book Chapters

{ T. Brown, P. Brass, M.P. Johnson, and S. Shamoun. Region coverage and protection with sensors: a survey. *Network Science for Military Coalition Operations: Information Exchange and Interaction*, D. Verma and D.I. Fotiadis, eds. Information Science Reference, 2010.

{ M.P. Johnson and A.P. Kosoresow. Lower bounds via worst-case instances via genetic algorithms. *Recent Advances in Simulated Evolution and Learning*, K.C. Tan, et al. ed. World Sci., 2004.

Other Publications

{ Various publications (details omitted) at Defense and Security Symposium on Unattended Ground, Sea, and Air Sensor Technologies and Applications.

{ Various publications (details omitted) at the Annual Conference of the ITA (ACITA). { M. Jakob, Z. Moler, A. Komenda, Z. Yin, A.X. Jiang, M.P. Johnson, M. Pechoucek and M. Tambe. AgentPolis: towards a platform for fully agent-based modeling of multi-modal transportation (demonstration) *AAMAS 2012*.

{ H. Rowaihy, S. Eswaran, M.P. Johnson, D. Verma, A. Bar-Noy, T. Brown and T. La Porta. A Survey of Sensor Selection Schemes in Wireless Sensor Networks. *SPIE DDS*, 2007. { A. Preece, D. Pizzocaro, K. Borowiecki, G. de Mel, M. Gomez, W. Vasconcelos, A. Bar-Noy, M.P. Johnson, T. La Porta, H. Rowaihy, G. Pearson and T. Pham. Reasoning and resource allocation for sensor-mission assignment in a coalition context. *MILCOM 2008*, San Diego, CA, Nov. 2008. { H. Rowaihy, M.P. Johnson, S. Eswaran, D. Pizzocaro, A. Bar-Noy, T. La Porta, A. Misra and A. Preece. Utility-based joint sensor selection and congestion control for task-oriented WSNs. *The Asilomar Conference on Signals, Systems, and Computers*, Pacific Grove, CA, October 2008.

Service and Activities

{ Ph.D. Advisor for:

- Haripriya Chakraborty, Fall 2015–present
- Elahe Vahdani, Fall 2015–present
- Samuel Bald, Fall 2014–present { Ph.D. Dissertation Committee Member for:
- Ou Liu (proposal, May 2015)
- Ali Assarpour (proposal, May 2015)
- Ning Xu (second exam, November 2013; proposal, February 2015)
- Feng Yu (second exam, October 2013)
- Valia Mitsou (proposal, November 2013; defense, May 2014)

NSF Review Panelist, June 2016

{ Program Committee Member, DISC 2016

{ NSF GRFP Review Panelist, January 2016

{ Session Chair, Fall Workshop on Computational Geometry, October 2015

{ NSF Review Panelist, July 2015

{ Member, Executive Committee of the CUNY Graduate Center CS PhD Program, Fall 2015–present

{ Member, Faculty Advisory Committee of the Data Science @ CUNY initiative, Spring 2015–present

{ Co-run the weekly Discrete Algorithms Seminar, CUNY Graduate Center, Spring 2014–present

{ Member, Computer Science Five-Year Plan Committee, Lehman College, Spring 2014–present { Reviewer: IEEE/ACM TON, IEEE TWC, IEEE TMC, Discrete Applied Math., Ad Hoc Networks, Sustainable Computing, Journal of Scheduling, Wireless Networks, SECON, SIROCCO, etc.

{ PC Member, AAAI 2012

{ Member, Executive Committee of the CUNY Graduate Center CS Department, 2006–2010

{ Grader, CUNY Math Challenge, 2009–2010

Talks

{ "Crossing Numbers and US Grad School", the Pontificia Universidad Católica del Perú, Lima, Peru, 8/12/2015

{ "Joint Optimization of Thermoelectric Power Plants", DIMACS MPE 2013+ Workshop on Natural Resources, Washington, DC, 6/4/2015

{ “Research in the Classroom: A Tool for Student Success Based on Authentic Research Experiences”, panel at 11th Annual CUE Conference, CCNY, 5/08/2015

{ “Secluded Connectivity Problems”

– CS Theory Seminar, NYU-Poly, Brooklyn, NY, 11/07/2013
– Discrete Algorithms Seminar, CUNY Graduate Center, New York, NY, 8/30/2013 and 8/20/2013 { “Representing a Graph Using Few Obstacles”, 23rd Fall Workshop on Computational Geometry (FWCG), CCNY, New York, NY, 10/25/2013

{ “Clustering with *k*-Means", University of Chicago, Chicago, IL, 5/06/2013

{ “The Knapsack and Ski Rental Problems”, Bard College at Simon’s Rock, Great Barrington, MA, 4/12/2013

{ “Optimizing Despite Handicaps: Networking and Game Theory”

– California State University, Fullerton, CA, 4/10/2013
– Texas State University, San Marcos, TX, 4/04/2013

{ “Optimizing Despite Handicaps: Knapsack Games, Neighborcast and Camera Coverage”, Wichita State University, Wichita, KS, 3/27/2013

{ “Optimizing Despite Handicaps: Knapsack Games, Battery Charging and Neighborcast”, Akamai, Cambridge, MA, 3/25/2013

{ “Optimizing Despite Handicaps: Knapsack Games, Forest Security, Secluded Paths”

– Lehman College, New York, NY, 3/21/2013
– University of Warwick, Warwick, UK, 3/14/2013
– Xerox Research Centre Europe, Genoble, France, 3/12/2013
– Institute for Defense Analyses, Bowie, MD, 2/25/2013

“The Knapsack Problem”

– Lehman College, CSM Seminar, Bronx, NY, 11/14/2013
– University of New Hampshire, Manchester, NH, 3/05/2013
– University of North Carolina, Charlotte, NC, 2/18/2013

{ “Optimizing Despite Handicaps: Knapsack Games and Battery Charging”, University of Southern California, Los Angeles, CA, 7/28/2012

{ “Covering with inexactly placed sensors", 17th Fall Workshop on Computational Geometry (FWCG), IBM Research, Hawthorne, NY, 11/09/2007

Teaching

{ The Graduate Center of CUNY

– Algorithms for Big Data (Spring 2016)
– Combinatorial Algorithms (Spring 2015)
– Approximation Algorithms (Spring 2014)

{ Lehman College

- Intro to Networks / Principles of Communication Networks (Fall 2015)
- LSP honors seminar "Unifying Ideas in Computer Science" (Spring 2015)
- Game Theory and Linear Programming (Fall 2014)
- Discrete Structures (2 times) – Intro

{ NYU, Stern School of Business

- Database Management Systems (4 times)

{ City College of New York

- Data Structures (2 times)
- Intro (2 times) { Columbia University
- Computational Complexity (CVN)
- Data Structures and Algorithms
- Discrete Math
- Programming Languages (C, C++, Java)
- Intro (2 times)

{ CUNY Institute for Software Development & Design

- Oracle 10g: SQL & PL/SQL (4 times)

References

{ Prof. Amotz Bar-Noy, CUNY, amotz@sci.brooklyn.cuny.edu

{ Dr. Prithwish Basu, Raytheon BBN Technologies, pbasu@bbn.com

{ Prof. Bhaskar Krishnamachari, USC, bkrishna@usc.edu

{ Distinguished Prof. Thomas La Porta, PSU, tlp@cse.psu.edu

{ Distinguished Prof. Rohit Parikh, CUNY, rparikh@gc.cuny.edu

{ Prof. Mani Srivastava, UCLA, mbs@ucla.edu

BRIAN MURPHY

brian.murphy@lehman.cuny.edu 718-960-8117

EDUCATION

GRADUATE SCHOOL & UNIVERSITY CENTER, CITY UNIVERSITY OF NEW YORK (May 2007)
Ph.D. Computer Science **3.9**

LEHMAN COLLEGE, CITY UNIVERSITY OF NEW YORK (January 1992)
M.S. Computer Science **4.0**

LEHMAN COLLEGE, CITY UNIVERSITY OF NEW YORK (September 1990)

B. S. Mathematics and Computer Science *Summa Cum Laude*

EMPLOYMENT

LEHMAN COLLEGE, CITY UNIVERSITY OF NEW YORK (September 1993 - Present)

Adjunct Lecturer (September 1993 – August 1998)

Substitute Lecturer (September 1998 – August 1999)

Instructor (September 1999 – August 2004)

Lecturer (September 2004 – August 2007)

Assistant Professor (September 2007 – August 2016)

Associate Professor (September 20016 – Present)

Deputy Chair: Department of Mathematics and Computer Science (January 2013- June 2014)

Chair: Department of Mathematics and Computer Science (July 2014- June 2017)

Chair: Department of Computer Science (July 2017 – Present)

Faculty Advisor: National Society of Black Engineers Lehman College Chapter (August 2014-Present)

Faculty Advisor: Society of Professional Hispanic Engineers Lehman College Chapter (August 2017-Present)

Faculty Advisor: CUNY 2x Tech (August 2018-Present)

Faculty Advisor: Tech Talent Pipeline (August 2017-Present)

Faculty Advisor: NSF Computer Science and Mathematics Scholarship Program – 2 Years (2010's)

Faculty Advisor: Math and Computer Science Club – 2 Years (Early 2000's)

Faculty Advisor: Video Game Programming Club – 4 Years (Early 2000's)

Faculty Advisor: Data Science Club – 1 Year (Early 2000's)

Thesis Advisor: Two Students

Ph.D. Advisor: One Student

Lehman Scholars Mentor: Five Students

Lehman McNair Scholars Mentor: One Student

Personnel and Budget Committee Elected to three year term (2013)

Educational Policy Committee Elected to two three year terms (2010 and 2013)

Computer Science Steering Committee (1998-2017)

Computer Information Systems Steering Committee (1998-2017)

Course Coordinator/Developer: CIS: 228, 249, 349 CMP: 274, 277, 168 Honors, 717, and 788

EON Reality Advisory Committee (Spring 2014 - Fall 2014)

Search Committee: Science Learning Center Coordinator

Search Committee: VP of Continuing Ed and Professional Studies

Search Committee: Interim Dean of Natural and Social Sciences

Search Committee: Dean of School of Education

Search Committee: Two Mathematics, Five Computer Science Faculty, and one Department Advisor

Executive Committee: Graduate Center Computer Science Ph.D. Program (Fall 2012 - Spring2016)
Ad Hock Governance Committee: Graduate Center Computer Science Ph.D. Program (Fall 2012 - Spring 2014)

IBM, T.J. WATSON RESEARCH CENTER, YORKTOWN HEIGHT, NY (September 2011 – August 2012)

Visiting Scientist, Multi-Core Division - Research in Parallel Algorithms

NEW YORK CITY DEPARTMENT OF EDUCATION (August 1996 – August 1997)

Mathematics Teacher, M.S. 95

NEW YORK CITY HEALTH AND HOSPITALS CORPORATION (August 1996 – August 1997)

Computer Aide, North Central Bronx Hospital, M.I.S. Department

COURSES TAUGHT:

CMP 788 GPU Programming
CMP 788 Parallel Programming
CMP 776 Parallel Algorithms and Architecture
CMP 717 Video Game Programming
CMP 485 Independent Study in Computer Science (Various Topics)
CMP 342 Internet Programming
CMP 338 Data Structures & Algorithms I
CMP 334 Computer Organization and Assembly Language
CMP 277 Programming in Java
CMP 274 Programming in C++
CMP 168 Programming Methods II (in Pascal, C++, and Java)
CMP 167 Programming Methods I (in Pascal, C++, Java, Python, and Java again)
CIS 349 Data Communications and Distributed Networks
CIS 249 Introduction to Local Area Networks (LANs)

CIS 228 The Internet
CIS 211 Information Processing
CIS 166 Computer Programming for Information Processing I
MAT 313 Linear Algebra
MAT 104 College Algebra
MAT 090 Algebra and Geometry
MAT 035 Intermediate Algebra

GRANTS & AWARDS

Center for Inclusive Computing - Northeastern Data Grant (2021)

For data collection and analysis to improve student success

WiTNY (2018)

For development of an introductory Computer Science course designed to recruit and retain students (especially female)

PSC-CUNY

Multiple Research Awards totaling $45K

Graduate Research and Technology Initiative (2014)
Allowed for purchase of computer equipment $7K

NVIDIA Equipment Award (2016)
High end NVIDIA CUDA cards were awarded $4K

Tech Talent Pipeline (2017)
A New York City Mayor's Office initiative to build a pipeline between industry and the Computer Science Department to open additional student internships and to provide access to industry specialist to teach at Lehman College.

CUNY 2x Tech (2018)
A New York City Mayor's Office initiative providing resources to help double the number of Computer Science students entering industry with high paying tech jobs. $2M

PUBLICATIONS

“Parallel Solution of Diagonally Dominant Banded Triangular Toeplitz Systems Using Taylor Polynomials”, Michael Kapralos , Aron Wolinetz, Brian J. Murphy, Proceeding of the 2016 IEEE 18th International Conference on High Performance Computing and Communications (HPCC), December 12-14, 2016, Sydney Australia.

“Parallel Butterflies Solve Bidiagonal Toeplitz Systems.”, Brian J. Murphy, Aron Wolinetz, Joshua Rogers, Proceedings of the 2015 International Conference on Parallel and Distributed Processing Techniques and Applications (PDPTA'15), July 27-30, 2015, Las Vegas, USA.

“Embarrassingly Parallel Butterflies Solve Diagonally Dominant Tridiagonal Toeplitz Systems.”, Brian J. Murphy, Proceedings of the 2014 International Conference on Parallel and Distributed Processing Techniques and Applications (PDPTA'14), July 21-24, 2014, Las Vegas, USA.

“Solving Tridiagonal Systems on a GPU”, Brian J. Murphy, Proceedings of the IEEE 20th Annual International Conference on High Performance Computing (HiPC'13) p159-168, December 18-21, 2013, Bangalore, India.

“Acceleration of the Inversion of Triangular Toeplitz Matrices and Polynomial Division”, Brian Murphy, Proceedings of the 13th International Workshop on Computer Algebra in Scientific Computing (CASC 2011) September 5-9, 2011, Kassel, Germany, Lecture Notes in Computer Science Vol. 6885, pg. 321-332 (2011).

“Additive Preconditioning for Matrix Computations”, V.Y. Pan, D. Ivolgin, B. Murphy, R. E. Rosholt, Y. Tang, X. Yan, Linear Algebra and Its Applications, 432, 1070-1089 (2010).

“Unified Nearly Optimal Algorithms for Structured Matrices”, V. Y. Pan, B. Murphy, R. E. Rosholt, Operator Theory: Advances and Applications, 199, 359-375, Birkhauser, Basel (2010) .

“Testing Pivoting Policies in Gaussian Elimination”, Brian Murphy, Guoliang Qian, Rhys Eric Rosholt, Ai-Long Zheng, Severin Ngnosse, Islam Taj-Eddin, Matrix Methods:Theory, Algorithms, and Applications, (Vadim Olshevsky and Eugene Tyrtyshnikov editors), 357-363, World Scientific Publishing, (2010).

“Faster Mulitpoint Polynomial Evaluation via Structured Matrices”, Brian Murphy, Rhys Eric Rosholt, Matrix Methods:Theory, Algorithms, and Applications, (Vadim Olshevsky and Eugene Tyrtyshnikov editors), 354-356, World Scientific Publishing, (2010).

“Nearly Optimal Symbolic-Numerical Algorithms for Structured Integer Matrices and Polynomials”, V.Y. Pan, B. Murphy, R.E. Rosholt, Proc. International Symposium on Symbolic-Numerical Computations (Kyoto, Japan, August 2009), (edited by Hiroshi Kai and Hiroshi Sekigawa), 105-113, ACM Press, New York (2009).

“A New Error-free Floating-Point Summation Algorithm”, V. Y. Pan, B. Murphy, G. Qian, R. E. Rosholt, Computers and Mathematics with Applications, 57, 560-564 (2009).

“Schur Aggregation for Linear Systems and Determinants” (by V. Y. Pan, D. Grady, B. Murphy, G. Qian, R. E. Rosholt, A. Ruslanov), Theoretical Computer Science, Special Issue on Symbolic-Numerical Algorithms (D.A. Bini, V.Y. Pan, and J. Verschelde, editors), 409, pp. 255-268 (2008).

“Additive Preconditioning for Matrix Computations”, V.Y. Pan, D. Ivolgin, B. Murphy, R. E. Rosholt, Y. Tang, X. Yan), Proc. of the Third International Computer Science Symposium in Russia (CSR 2008), Lecture Notes in Computer Science (LNCS), 5010, 372--383 (2008).

“Eigen-solving via Reduction to DPR1 matrices”, V. Y. Pan, B. Murphy, R. Rosholt, Y. Tang, X. Wang, A. Zheng, Computers and Mathematics with Applications, 56, 1, 166-171, (2008).

“Additive Preconditioning and Aggregation in Matrix Computations”, V. Y. Pan, D. Ivolgin, B. Murphy, R. E. Rosholt, I. Taj-Eddin, Y. Tang, X. Yan, Computers and Mathematics with Applications, 55, 8, 1870-1886 (2008).

“Root-finding with Eigen-solving”, V. Y. Pan, D. Ivolgin, B. Murphy, R. E. Rosholt, Y. Tang, X. Wang, X. Yan, Chapter 14, in Symbolic Numerical Computation, (Dongming Wang and Lihong Zhi editors), 185-210, Birkhäuser, Basel/Boston, (2007).

“The Schur Aggregation for Solving Linear Systems of equations”, V. Y. Pan, B. Murphy, R. E. Rosholt, M. Tabanjeh), Proceedings of the Third International Workshop on Symbolic--Numeric Computation (SNC 2007), July 2007, London, Ontario, Canada, (Jan Vershelde and Stephen Watt eds.), 180-188, ACM Press, New York, (2007).

“Real Root-Finding”, Victor Y. Pan, Brian Murphy, Guoliang Qian, Rhys Eric Rosholt, Yuqing Tang, Proceedings of the Third International Workshop on Symbolic--Numeric Computation (SNC'07), July 2007, London, Ontario, Canada, (Jan Vershelde and Stephen Watt eds.), 161-169, ACM Press, New York, (2007).

“Linking the TPR1, DPR1 and Arrow-Head Matrix Structures”, V. Y. Pan, M. Kunin, B. Murphy, R. E. Rosholt, Y. Tang, X. Yan, W. Cao, Computers and Mathematics with Applications, 52, 10-11, 1603-1608, (2006).

Itai Feigenbaum

Contact Information
Department of Computer Science
Lehman College, CUNY
Gillet Hall, Room 137C
250 Bedford Park Blvd W
Bronx, NY 10468
itai.feigenbaum@lehman.cuny.edu
http://www.itaifeigenbaum.com

Education **Department of IEOR, Columbia University**

Ph.D. in Operations Research, 2016

- Dissertation Topic: Optimization in Strategic Environments
- Advisor: Jay Sethuraman

M.Sc. in Operations Research, 2012

Rutgers University–New Brunswick

B.Sc. in Mathematics (Honors Track), 2011

- Minors in Computer Science and Operations Research
- Highest Honors in Mathematics, Summa Cum Laude, Phi Beta Kappa

Current Position

Department of Computer Science, Lehman College and the Graduate Center, CUNY

Assistant Professor, 2016-current

Journal Papers

I. Feigenbaum, J. Sethuraman and C. Ye, *Approximately Optimal Mechanisms for Strategyproof Facility Location: Minimizing L_p Norm of Costs*, Mathematics of Operations Research 42(2) 434-447, 2017.

I. Feigenbaum, Y. Kanoria, I. Lo and J. Sethuraman, *Dynamic Matching in School Choice: Efficient Seat Reallocation After Late Cancellations*, Management Science 66.11 (2020): 5341-5361.

• This paper was the topic of an invited plenary talk in STOC 2018.

I. Feigenbaum, M. Li, J. Sethuraman, F. Wang and S. Zou, *Strategic Facility Location Problems with Linear Single-Dipped and Single-Peaked Preferences*, Autonomous Agents and Multi-Agent Systems 34.2 (2020): 1-47.

Conference Papers

I. Feigenbaum, Y. Kanoria, I. Lo and J. Sethuraman, *The Magicians' Shuffle: Reusing Lottery Numbers for School Seat Redistribution*, Web and Internet Economics: 12th International Conference (WINE 2016), Montreal, Canada, December 11-14, 2016, Proceedings. Vol. 10123. Springer, 2016.

I. Feigenbaum and M. Johnson, *Selfish Knapsack*, The 31st AAAI Conference on Artificial Intelligence (AAAI 17), 2017.

I. Feigenbaum, *Strategyproofness in Kidney Exchange with Cancellations*, Web and Internet Economics (WINE 2022), 2022.

Teaching Experience

At CUNY:

Spring 2017	Instructor, Discrete Structures (two sections)
Fall 2017	Instructor, Algorithmic Game Theory (undergraduate level)
Fall 2017	Instructor, Algorithmic Game Theory (Ph.D. level)
Spring 2018	Instructor, Discrete Structures (two sections)
Fall 2019	Instructor, Algorithmic Game Theory (undergraduate level)
Fall 2019	Instructor, Algorithmic Game Theory (Ph.D. level)
Spring 2020	Instructor, Discrete Structures (two sections)
Fall 2020	Instructor, Discrete Structures (two sections)
Spring 2021	Instructor, Discrete Structures (three sections)

Fall 2021 Instructor, Discrete Structures (two sections) Spring 2022 Instructor, Discrete Structures (two sections)

At Columbia:

Spring 2012 Teaching Assistant, Applications Programming for Financial Engineering

Summer 2012 Teaching Assistant, Foundations of Financial Engineering

Fall 2012 Teaching Assistant, Introduction to OR: Deterministic Models Spring 2013 Teaching Assistant, OR in Public Policy

Fall 2013 Teaching Assistant, Introduction to OR: Deterministic Models Spring 2014 Teaching Assistant, Introduction to OR: Deterministic Models
Fall 2014 Teaching Assistant, Introduction to Mathematical Programming Spring 2015 Teaching Assistant, Introduction to OR: Deterministic Models

Fall 2015 Teaching Assistant, Introduction to Mathematical Programming

Spring 2016 Teaching Assistant, OR in Public Policy

Extended Professional Travel

Visiting Researcher at the Department of Computer Science at the CUNY Graduate Center, New York (Summer 2015)

Service and Activities

Honors advisor for the department of Computer Science at Lehman College (providing guidance to students with exceptional abilities)
Member of the Computer Science Curriculum Revision Committee, Lehman College

Member of the Advisory Committee, Data Science Initiative, Lehman College

Member of Computer Science Search Committee, Lehman College

Research Advisor in the NYC Louis Stokes Alliance for Minority Participation (LSAMP)

Reviewer for Mathematics of Operations Research, Operations Research, Economic

Theory Bulletin, SAGT

Faculty advisor to Tech Talent Pipeline Residency Program, Lehman College

Member of the Algorithms Qualifying Exam Committee, Ph.D. Program in Computer Science, CUNY Graduate Center

Member of the Membership Committee, Ph.D. Program in Computer Science, CUNY Graduate Center

Observer for Lehman College’s Computer Science instructors

Member of Ph.D. Dissertation Committee:

- Matteo Campanelli (2018)

Grants and Honors Six PSC-CUNY Research Awards, $4530 each (2017-2022)

Kenneth and Rosalind Wolfson Award for Academic Excellence in Mathematics (2011)

Lawrence Corwin Memorial Math Prize (2011)

Weill Scholarship Winner (2010 and 2011)

Dean's Excellence Award (2011)

Rutgers Academic Excellence Award (2009)

Miscellaneous U.S. Citizen
Languages: English, Hebrew

Appendix D: Self Evaluation Report from 2011-2012

Lehman College Department of Mathematics and Computer Science

Self Evaluation

The **mission of the department** is threefold: to provide Lehman students with the skills necessary for quantitative and computationally assisted thinking and with the professional training and experience for STEM careers in industry, education, and government; to contribute to the world's body of knowledge through our scholarship; and to serve the community through advocacy for education and career preparation for women and minorities, training of teachers, and partnerships in industry.

The **specific goals of the department** likewise fall into three categories, pertaining to teaching, research, and service to the academic needs of our community, on campus and beyond:

1. Our departmental teaching goals are to provide:

- all students with the skills to obtain, analyze and draw conclusions from data; to develop abstract mathematical models needed to solve everyday problems; to use computers and computational tools effectively; and to think rigorously and reason abstractly
- our MAT (mathematic) majors with a solid grounding in algebra, analysis, geometry and probability/statistics
- our CMP (computer science) and CIS (computer information systems) majors with the ability to design and implement computer systems; to solve computational problems efficiently and correctly; and to integrate information systems with organizations to improve their functioning
- our CGI (computer graphics and imaging) majors with the technical skills and the ability to organize visual elements necessary to communicate concepts and embody experiences across a wide range of media
- students in quantitative majors (e.g. Economics, Biology) with the foundational knowledge to use mathematics and computers effectively in research and coursework

2. As an active member of the Lehman community, the department will:

- educate math and computer science majors in the tools of their field; members of the natural and social sciences with the tools from mathematics and computer science which they will need in order to purse their fields of research
- engage in efforts to include all students, especially women and minorities, in STEM fields
- collaborate with other academic departments at Lehman to promote the effective use of computers and mathematics in everyday life and work, and especially in other fields, such as science and economics, which use mathematics and computer science extensively.
- advocate for improvements in computer usage and environments at Lehman

- work with Bronx high schools and College Now to help prepare students to meet Lehman's rising standards of admission.
- collaborate with local community colleges regarding effective preparation for students transferring in to Lehman

To achieve these goals in a world that is changing technologically and economically, we will

- evaluate syllabi and delivery of specific course offerings and curricula for the department's majors in MAT, CMP, CIS and CGI. Specifically, we will ○ rework our business calculus sequence
    - bring our computing curriculum more in line with ACM curriculum standards, in particular, add a Software Engineering major.
    - request an evaluation of the CMP program by ABET (the computer science accreditation body).
    - develop our honors precalculus and calculus courses to challenge our bestprepared students. ○ continue to develop industrial and research internships, research projects, and other special career-oriented programs
    - modify the M.A. Program for Secondary School Teachers of Mathematics so as to provide appropriate preparation for teaching the recently-revised New York State Common Core Learning Standards for Mathematics

3. Our research goals are as follows:

- **Computer Science;** A fundamental goal for the Computer Science research program is to continue its research in computationally intensive computing, an area of common interest among many current faculty and one that raises interesting issues in all core computer science fields. This requires better infrastructure for the department, for which we will apply to various sources, including governmental agencies like the NSF and corporations like NVIDIA. The current computer science research in the department suggests synergies among computational biology; bioinformatics; image processing; computer graphics and numerical, symbolic, and symbolic-numerical computations. The award of such an infrastructure grant to a department increases its visibility and improves its reputation and ability to hire.
- **Mathematics:** The research of the mathematics faculty is broadly based in all major fields of mathematics and the faculty has won numerous awards. A fundamental goal is to hire a new generation of mathematicians to continue a legacy of outstanding research.

Mathematicians and Computer Scientists at Lehman College (resumes in Appendix A) have always engaged in internationally recognized research. They have been directly involved with the training of the next generation of researchers, professionals and teachers in their fields. This has been in addition to their role in providing all Lehman students with the quantitative background they need to pursue the careers of their choice. To continue at the level that has been maintained for many years, we must continue to hire dynamic faculty.

4. Assessment Plans

- **Mathematics** (syllabi and hard copy of plan in Appendix B)

As we progress through the years, different courses will be assessed in order. Assessments of courses using the new system of outcomes began in Spring 2010. Assessments of the lower level courses, College Algebra-Calculus, will involve a reevaluation of uniform syllabi, which include careful scheduling of every chapter in the textbook. The calculus committee has always reevaluated such courses, but modern methods using outcomes only began in 2010. Assessments of advanced courses will be done on a more individual basis by faculty. Every semester, course outcomes will be tested on a final exam in all math courses. The schedule for thorough assessment of particular courses is on the Math Assessment Schedule Site (googledoc) Reports on the thoroughly assessed courses will become available at this site when they are completed.

Modern assessment of the lower level courses began with the collection of data for College Algebra on our uniform final in Spring 2010. This data consists of the scores of all students on each question of the uniform departmental final. These questions on the final directly measure the various outcomes for the course. In Fall 2010 we proceeded to Precalculus, then Calculus I in Spring 2011. Calculus II will be evaluated in Fall 2011 and then back to College Algebra over a two-year cycle. For each course, the assessment will begin with data collection, followed by analysis, followed by proposals to the Educational Policy Committee and ending with possible adjustments to the syllabi if needed. In addition to assessing for outcomes, we will continue our traditional methods of assessment in which a course is judged not only by progress of the students in the course, but by their success in subsequent courses. For this reason we are assessing the sequence in ascending order.

More advanced courses without uniform syllabi will be assessed every few years by the faculty teaching the course. This assessment began with data collection on Geometry in Spring 2010 (a course designed with NSF funding through the MTTI program to train teachers to teach the new NYS HS curriculum). The Geometry course's outcomes are assessed through various problems on the final exam. Data on course projects has also been collected. In Fall 2010 data was collected on Analysis I (a required course in the math major) and on Discrete Math (another course to train teachers). In 2011, we will assess Linear Algebra and Modern Algebra. Analysis of the data will be completed the semester after the data is collected and faculty will be informed of the results so that they may adapt their courses accordingly. The data in each situation will consist of recording the performance of students on problems on the finals pertaining to each specific course objective.

The major itself will be assessed by directly testing students on their achievement of the math major outcomes in the required courses of the math major. For mathematics, every outcome is directly measured in multiple courses via final exams with an emphasis on *Technology* in the Calculus Laboratory Sequence (MAT155 and MAT156) and an emphasis on *Constructing a Rigorous Mathematical Argument* in Real Analysis (MAT320) and Modern Algebra (MAT314). Other outcomes of the mathematics major appear repeatedly throughout the major. Naturally students may take electives providing further emphasis on any particular outcome which is most relevant to their career path. Future teachers, in particular, are required to take Modern Geometry (MAT345) and Discrete Mathematics (MAT237). Students interested in enhancing their computer programming skills may elect to take Programming Methods I (CMP230) which is assessed as part of the Computer Science major. Each math course objective has been assigned one or more corresponding major objectives (see the syllabi). In this way we may progress through the assessment of the major outcomes as well as the courses' outcomes using the same direct measure: the final exam. The official Lehman College Charted Assessment Plan for the Math Major (googledoc) itemizes the outcomes of the major and how each is assessed.

Assessment Plans and Conclusions for Each Semester:

Spring 2010 Plan and Spring 2010 Plan, Data and Conclusions (completed)
Fall 2010 Plan and Conclusions (in progress)
Spring 2011 Plan

The Math Assessment Ambassador found it easier to create a website as a common repository for data and analysis rather than typing up a single report with appendices. This website is publicly viewable.

- **Computer Curricula.**

The assessment of the computer curricula, began in the Fall of 2010. The necessary beginning step was to establish the learning goals for the three majors (CMP, CIS, and CGI). These are included in Appendices D, E, and F, respectively. Next, a mapping was prepared from program learning objectives to courses for CIS and CMP courses (in appendices D and E). Third, we worked to bring the curricula for CIS and CMP in line with the ACM Curriculum Recommendations and to establish an assessment schedule beginning with the lower-level, multi-section courses.

The CIS curriculum presented a special challenge, since it has been taught primarily by adjuncts. Fortunately, most of the adjuncts have taught the courses long-term, providing some stability to their content. However, there were no standard syllabi and the curriculum had never been reviewed against any standard curricula. To address this, we assigned fulltime professorial faculty to coordinate each CIS course, asking that they observe sections of their courses and develop initial standard syllabi for them. With the help of the adjuncts, syllabi including testable learning objectives were developed in the fall of 2010 and the spring of 2011 for the following nine courses:

○ CIS 106 (Computer Literacy),
○ CIS 166 (Computer Programming for Information Processing I), ○ CIS 211 (Computer Information Systems) ○ CIS 212 (Microcomputer Architecture) ○ CIS 234 (Introduction to Spreadsheet Analysis) ○ CIS 242 (Systems Analysis and Design) ○ CIS 244 (Introduction to Database Management) ○ CIS 331 (Introduction to Network Technologies) ○ CIS 345 (Introduction to Operating Systems) These syllabi are included in Appendix C.

Additionally, in the spring of 2011, we scheduled an assessment of the introductory course for the CIS major, CIS 211. As a result of this assessment, we modified the learning

objectives to work better with our curriculum, and selected a new text that had less overlap with the material in the text for CIS 212, usually the second course for a student majoring in CIS. The two courses had converged over the years in such a way that both were covering information technology only, and we intend the CIS 211 course to focus instead on the use of information systems to support business objectives.
Also in the spring of 2011, we began a series of assessments for courses in the CMP major, starting with CMP 230. A new language, Python, was introduced as the introductory programming language in CMP230 in the Spring of 2011. The three sections of the course were taught collaboratively, using not only the same syllabus[1] but the same assignments and the same exams, graded in common. We will continue this practice in the future. Because of this major change to the course, this was an excellent time to assess it. We were pleased with a 75% pass rate on the course compared to a 50% pass rates in recent years for introductory programming at Lehman. We will try to get the pass rates for other CUNY colleges in the future.

For the fall of 2011, we are repeating the assessments of CIS 211 and CMP 230 and beginning to assess CIS 212 and CMP 326. Also, we are working to enforce the use of the standard syllabi as templates for the syllabi that the adjuncts use. This will be done through the assessment procedure, in which we will assess the learning objectives from the standard syllabus.

Department Overview

1. - Department Overview - Faculty

Faculty teaching mathematics in the department comprise a very prestigious group((resumes in Appendix A). A large percentage of the group are members of the doctoral faculty at the CUNY Graduate Center. Many hold or recently held NSF research grants. For many years Lehman was recognized as having the strongest mathematics faculty in all of CUNY. Indeed, several years ago, one of the national leaders in mathematics claimed that if we were a freestanding school we would be nationally ranked. It is getting harder to maintain that quality. Our teaching load is higher than comparable institutions and the pay is lower. We have had four of our best young faculty resign to go elsewhere. Their quality is attested to by the institutions they went to - Indiana, Minnesota, Wisconsin and London (as an aside, the faculty member who went to Indiana is now being recruited by Cal Tech). Three of the four left while holding NSF research grants and three were members of the doctoral faculty. We have not been allowed to replace the last two faculty members who resigned.

Faculty teaching computer science in the department are an equally prestigious group. Six are members of the doctoral faculty at the CUNY Graduate Center with three of them teaching there every year. Three members currently hold NSF research grants.

The department is very proud of the high quality of teaching by the full-time faculty members with four members of our department being winners of the Lehman College Teacher of the Year Award.

Many departmental faculty are also members of college committees. Some of the committees on which departmental members serve are:
Search Committee for VP for Information Technology, Search Committee for VP for Administration, Research Advisory Board, Technology Fee Committee, Senate Curriculum
Committee, Senate Committee on Academic Standards and Evaluation, Senate Budget Committee, FP&B Committee on Committees, FP&B Budget Committee, FP&B Subcommittee on Tenure and the Executive Committee of the Faculty.

2. - Department Overview - Programs

Our department has five undergraduate majors, four undergraduate minors and three graduate programs.

The undergraduate majors are:

Mathematics B.A. - This is a fairly standard mathematics major targeting both students who wish to go to graduate school and students heading towards a teaching career.

Computer Science B.A. and B.S. - These programs target students heading to graduate school and students wishing to go to work in the computer industry. These majors need reviewing according to ABET accreditation guidelines and ACM curricular guidelines

Computer Information Systems B.S. - This major targets students who wish to become the computer specialist in a small office or work in an IT organization. This is our most popular major. These students tend to be the weakest of our majors and we are slowly raising the standards in this major.

Computer Graphics and Imaging B.S. - This is a joint program with the Art department and targets, as its name implies, students who wish to work in computer graphics.

Our department has four undergraduate minors. In two of the minors, Mathematics and Computer Science, students must take several preliminary courses before getting into the upper level courses. This is due to the sequential nature of these disciplines, In the other two, Computer Applications and Computer Graphics, students can take some upper level courses as part of the minor. As the table in the data section that follows shows, Computer Applications (the minor corresponding to the Computer Information Systems major) is very popular and attracts many of the B.B.A. majors. In recent years, we have been raising the level of requirements for this minor.

Our three graduate programs are:

Computer Science M.S. - This is a program for students who want to go on to a Ph.D. program in Computer Science or for undergraduate Computer Science majors who wish to take more advanced courses before going on to or working in the computer industry.

Program for Secondary School Teachers of Mathematics MA - This program is designed to sharpen the math skills of young mathematics secondary school teachers en route to their professional license.

Mathematics M.A. - In general, this program is designed to prepare students to enter a doctoral program in mathematics. Its courses attract some of the better undergraduate students.

For some time, we were not accepting students into the pure math masters because there weren't enough applicants to justify offering a full range of advanced graduate math courses. Then, some five years ago, 10 teachers who were finishing or had already finished the TMA, had permanent licenses, and were teaching in N.Y.C. schools submitted a petition to our department: If we reopened matriculation in the pure mathematics masters, they would matriculate in it, and use the program to obtain their 30 post masters credits. They wanted the credits to enhance their salaries, of course, but they wanted to do this in a structured high quality program that would also give them a masters degree that could get them into a doctoral program, at some later time.

College Math Requirement - Our department is responsible for the College Math Requirement. Essentially, the requirement is that all students must take a 3 or 4 credit MAT course numbered above 125. Students who are going to major in the sciences generally take MAT 172 and 175 - the standard pre-calculus and calculus courses. Students majoring in Economics or Business Administration take MAT 171 and 174 - pre-calculus and calculus for Business. Most other students (approximately 1000 per year) take MAT 132 - Basic Concepts of Probability and Statistics. Many students (over 1000 per year) do not come to Lehman with sufficient background for these courses and take MAT 104 - College Algebra first. Some other students are not even prepared for MAT 104 and must take the workshops we offer in more elementary Algebra.

Special Programs -

a) Lehman-IBM Internship Program - For the past eleven years we have had an internship program with IBM. The program currently has eight students who spend two days a week on research projects at IBM's Watson Research Lab in Hawthorne, NY. The program has over forty alumni, many of whom have gone on to work at other branches of IBM.

b) NSF Scholarship and Mentoring Program - This nine year old program has provided more than 150 students with substantial mentoring in addition to laptops and stipends of up to $3,000 per student. About forty students are currently in the program that is funded by a series of grants from the NSF totaling $1,100,000. Over 95% of the students in this program graduate.

c) MTTI (Math Teacher and Transformation Institute) - Four departmental faculty, one of whom is the Principal Investigator, are involved (together with faculty from Secondary Ed and the Institute for Literacy Studies) in a $5,000,000 NSF grant to train selected high school teachers. This program, the MTTI program, aims to prepare the teachers to better prepare their high school students for college level mathematics.

3. - Department Overview - Data

Number of Undergraduate Students per Program

Majors:	*Sep 2010*
Comp Sci-B.A.	7
Comp Sci-B.S.	71
CIS – B.S.	124
CGI – B.S.	34
Math – B.A.	86

Minors

Comp Science	13
Computer App	228
CGI	13
Math	15

Number of Graduate Students per Program:

Comp Sci – M.S.	21
Math M.A.	8
Teach Math M.A.	71

Headcount of Students Taking Courses in the Department - 3530.
This is more than any other department meaning that, in terms of student enrollment, we are the largest department in the college. We have 691 students that are in our majors taking our courses (counted multiple times in different courses).

Grant Funding per Year

20010-11	$7,608,192
2009-10	$1,894,402
2008-09	$2,008,341
2007-08	$381,962
2006-07	$920,715

Faculty Profile

Challenges

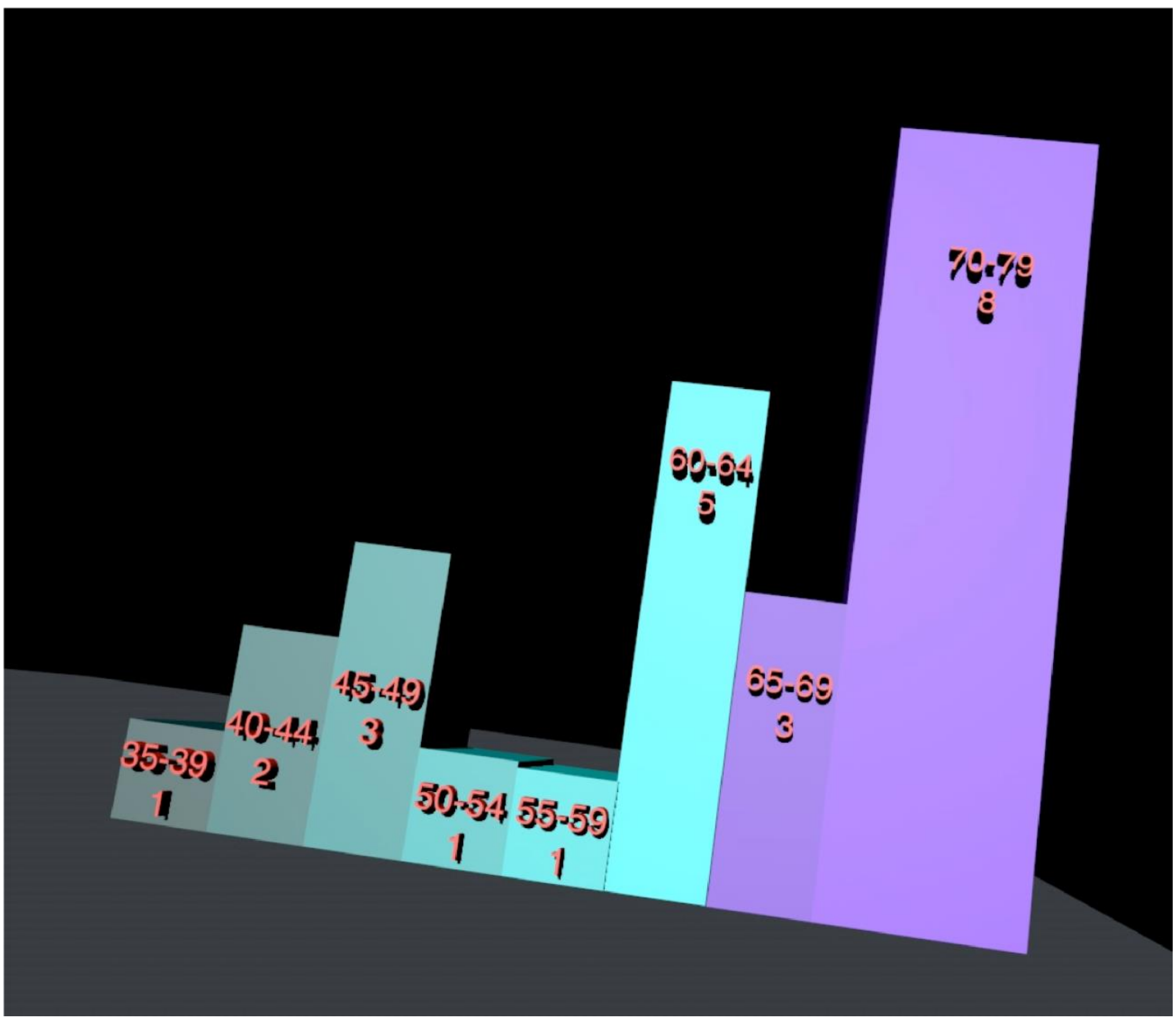
1. - Challenge – Lack of Professorial Contact

Our department is heavily reliant on the use of adjunct faculty and graduate students in the teaching of many of our courses. In Spring 2011:

- 83% of our MAT courses are taught by adjuncts and graduate students.
- 97% of our CIS courses were taught by adjuncts and graduate students and 83% will be taught by adjuncts and graduate students in the fall of 2011
- 19% of our CMP courses were taught by adjuncts and graduate students; in the fall, that will be 25%
- 56% of our CGI courses were taught by adjuncts and graduate students.
- 63% of our sections (118 sections) were taught by adjuncts and graduate students.
- In MAT 104 (college algebra), the course taken by over 1000 students a year to prepare for the College Math Requirement, 100% of the sections were taught by adjuncts and graduate students.
- In MAT 132 (probability and statistics), the course taken by most of the college students to satisfy the College Math Requirement, 100% of the sections were taught by adjuncts and graduate students. It should be noted that when the department agreed to undertake the College Math Requirement, we were promised by Provost Wille that we would have enough full time lines to staff at least half of the courses in the College Math Requirement. Among the goals of Lehman College Strategic Planning Council Report for 2009-2019, is raising the level of the Colleges mathematics requirement.
- In MAT 171-174, the courses taken by Economics and B.B.A. students as part of their major, 92% of the sections were taught by adjuncts and graduate students.
- In MAT 172-175, the math courses taken by science majors in preparation for their major, 89% of the sections were taught by adjuncts and graduate students.. We have amassed adjuncts that excel in teaching courses up to college algebra and a few that excel in teaching higher-level courses. We need fulltime faculty teaching some sections of MAT 132 and MAT 104 (each course services over 1000 students a year). These faculty lend stability to the courses and give us necessary information to continually adjust the syllabi. We need professorial faculty cognizant of advances in probability and statistics closely monitoring MAT 132. It is a concern that students in mathematics often do not consistently meet professorial faculty early in their studies. It is a concern that students regularly do not meet professorial faculty in CIS.

2. - Challenge - Aging of the Faculty

The Lehman College Strategic Planning Council Report for 2009-2019 expresses concern that at present 85 of Lehman's 373 full-time faculty are 65 years of age or older and 143 are 60 or over. In terms of percentages, 22.8% of the college's faculty are 65 or older and 38.3% are 60 or older. These figures of concern over the aging of the Lehman faculty pale when compared to those of our department. In our department, 8 out of 24 (33%) ***are 70 or older*****,** 11 out of 24 (46%) are 65 or older and 16 out of 24 (67%) are 60 or older. The problems of large use of part time faculty will grow as these faculty leave. There are several higher-level courses taught by non-professorial staff. The percentage of higher-level courses inappropriately staffed will grow. We will not be able to offer numbers of higher-level courses. Unless we are given new professorial lines we are headed towards a departmental default. The present faculty is distinguished and attracts new personnel of high quality and has helped the reputation of the school. Resuscitating the department after the professorial faculty has contracted to a small core may be a long and arduous chore.



3. - Challenge - Other areas:

Aside from concerns about the over-reliance on adjuncts and the aging of the faculty, other major concerns of our department are:

- There are 15 multi-section courses of 4 or more sections in the Spring 2010 semester. In 7 of those 15, there are no professorial member teaching and adding stability to the courses.
- Because of the lack of a computer technician, access to the one computing lab is limited to class hours, but students need extensive access including nights and weekends.
- All computing fields require labs and lab support flexible enough to accommodate yearly changes as technology changes, but there are no supporting lab technicians for the Computer Science classroom and faculty research labs. There is no lab space for Computer Science students to do their work and experiment on their own and not have to compete with students doing word processing or surfing the Internet. At the November meeting of the Lehman College Senate, this was expressed by the students as their prime concern.
- The computing infrastructure commonly found in corporate and government offices and other college campuses should be available to Lehman College faculty and students. In particular there should be campus-wide availability of share drives together with login from any location. This is something departmental faculty would work on together with the IT staff.
- The above problems could be alleviated by investigating the use of cloud computing, but students will be required to have Internet access to take full advantage of this.
- For lack of sufficient funding, students do not have sufficient access to tutoring for computer science or CIS courses, in particular in the programming courses.
- Because of the lack of sufficient professorial faculty, we have not been able to offer all the requested graduate mathematics courses for teachers.
- Because of the lack of sufficient professorial faculty, we have not been able to offer an evening section of the basic mathematics course for prospective teachers.
- Because of the lack of sufficient professorial faculty we tried to have an adjunct teach an axiomatic geometry course for prospective teachers with disastrous results.
- Because of the lack of sufficient qualified faculty, we are unable to offer a course in network security in either the CIS or the CMP curriculum, both of which require such a course, and we have difficulty offering the popular and important e-commerce course in the CIS curriculum.
- Because of the lack of sufficient professorial faculty, we have had to turn down a recent request by a local high school to develop with them a Mathematics Partnership grant with the NYS Department of Education

5. Curricular Plans

- Members of the P&B Committee feel that the department should split into separate mathematics and computer science departments.
- Together with faculty from the Department of Biological Sciences, we have instituted a joint Quantitative and Systems Biology minor. The first course in the major will be taught in the fall, although it will be taught as a topics course because the approval wasn't granted until late in the spring. The respective department chairs will ensure that students get appropriate credit toward a minor if they want it. This is of interest to students in both departments and is one of the recommendations of the Lehman
  College Strategic Planning Council Report for 2009-2019
- Less advanced in the planning stage is an Information Security minor. This will require hiring a knowledgeable faculty member.
- We are looking into getting ABET certification for our Computer Science programs (City College and College of Staten Island have this certification). While we expect that our programs will be near the levels required for this certification, it is likely we will have to offer some courses on a more regular basis.
- We would like to develop some calculus-based courses in the Mathematics of Finance. A topics course in this area was taught in Spring 2010 (and was very well received) but the faculty member who taught it resigned.
- In the light of all the talk (in Tweed, Albany and Washington) about getting better trained math teachers, we are planning on strengthening the courses and course requirements in the TMA program. Some courses are currently being tested in the NSF funded MTTI program and we are planning on introducing them to the TMA program.
- As CUNY enters its "Decade of Science" we would like to add Honors sections of our calculus courses and eventually, computer science. It should be noted that the number of students taking Calculus I has increased by 65% from Spring 2009 to Spring 2010. We are part of a grant organized at the CUNY graduate center to create honors sections in mathematics. Given the department's paucity of professorial faculty and its demographics we may have trouble staffing these sections.
- We would like to add a follow-up Computer Literacy course that would emphasize group wise skills such as finding, obtaining, and preparing documents and presentations; using various forms of groupware to contribute to group projects, e.g. using wikis; finding and using publicly available relevant data bases to a specific topic and preparing arguments and presentations based on the data.
- Skilled personnel in computer-related fields are very much in demand: Two of the top 4 (and 4 of the top 30) careers as rated by the US Bureau of Labor statistics are sub-fields of Computer Science for the years 2006-2016. IBM has determined that 50% of its own and its customers' IT staffs are currently eligible to retire. Both these facts indicate that there will be many jobs available for properly prepared students. This should result in an increase in enrollment in Computer Science courses and is already in evidence in the increase in enrollment in the lower level courses for the major.

6. Resources Needed

1. - **Resources Needed - Faculty**

In the previous sections, mainly Adjunct Usage (actually over-usage) and Aging of the Faculty, but also in the Curricular Plans and Other Areas of Concern, we clearly showed the need for a substantial increase in the number of faculty. In particular, in the five areas listed under Adjunct Usage, the need for professorial faculty in our department is a need not only for students in our department but also by many of the students in the college. In a previous section, we indicated that we had had four resignations of young mathematicians in the past 6 years. The last two were clearly not replaced despite the over usage of adjuncts by our department. During 2010-2011 we have had one more resignation, two retirements and one faculty not given tenure. We have been able to hire one lecturer but we have had a net loss of three faculty members. We are definitely headed downhill.

The section Aging of the Faculty indicates very clearly that we will lose many faculty in the next 5 years. Together with the over usage of adjuncts it would be wise to average about 3 to 4 hires in each of the next 5 years. This has to be done in a methodical way, not by just replacing faculty as they retire. First of all, there is a need for additional faculty, Secondly, if we started searching after faculty announced their retirements (and finished their Travia leaves), we would have at least a one years lag in hiring. Thirdly, if we hire 2 in one year and 6 in the next, we would be hiring, at best, our fifth and sixth choices instead of our third and fourth (if we were to hire 4 in each year).

We would like to return to the pre-eminent position we previously had in the national mathematics community and in CUNY. That means hiring top-notch faculty regardless of area. We would like to hire at the level of the four young mathematicians who resigned in the past five years to go to top level schools elsewhere. Our department in previous years has done well by hiring the best available mathematicians and then arranging for them to go into the specific areas of need. For example, a topologist was developing a course in the Mathematics of Finance (before he resigned), a Teichmuller specialist is developing an expertise in Statistics because of departmental needs and a Differential Geometer has just received a $5,000,000 grant in teacher training. Incidentally, all three were recipients of NSF funding in their own areas of research.

We would like at least 3 professorial lines for computational research, with emphasis on scientific computing in collaboration with bio and chemistry, parallel processing, numeric analysis, and complexity theory. The need for these lines has been amply demonstrated in the sections Challenge – Lack of Professorial Contact and Challenge - Other Areas. It has been particularly demonstrated in the section on Curricular Plans. It should be noted that among the tenured faculty teaching computer science there are only two under the age of 60. Because of the economy, many of the more prestigious computer companies are not hiring and are, in fact, laying off people. We should seize the moment to hire the top-notch computer scientists we need now and for the immediate future.

In CIS we should hire at least 3 researchers with interests in health information systems or management information systems or operations research.

2. - **Resources Needed - Support Staff**

The current staff supports the calculus labs, labs for statistics and the computer applications program but has no time to support the unique needs of a dynamic teaching curriculum or active research program including those that involve undergraduate students in Computer Science. Additional technical support, via full-time College Lab Technicians or HEO’s, is needed.

The instructional program needs one such person to support setting up course laboratories with appropriate hardware and software and to help maintain access to labs at night and on weekends. On-line access to CS labs through the Lehman vpn (Virtual Private Network) would be acceptable in some situations, but remote access is a requirement.

The research program needs another such person to support setting up research labs, backing up necessary files, making sure of appropriate remote access to the labs, protecting the Lehman networks from unnecessary interference, helping to find and purchase appropriate hardware and software, and other needs as determined by research faculty.

3. - **Resources Needed - Equipment and Labs**

There is a need for additional labs for our students. The labs need to be well equipped with computers, routers, printers, switches, and other equipment, so that the students can experiment for class assignments and on their own. As indicated earlier, at the November meeting of the Lehman College Senate, the student leadership called this their highest priority

Computer Science faculty require individual labs to do experimental research and such labs are required to attract new faculty. In addition, large computational facilities, including the ability to store and share large amounts of data, are required so that large research problems can be addressed.

Appendix E: External Review Reports from 2011-2012

External Review of the Department of Mathematics and Computer Science at Lehman College, CUNY

Dan Prener (prener@us.ibm.com)

Introduction

After having been provided with the department's self-evaluation, the review committee visited the campus on November 7th and 8th, 2011. Over the course of the two day visit, we met with the Dean, the Provost, the department Personnel and Budget committee, a number of faculty members, and a few students.

History

The Department of Mathematics and Computer Science has a distinguished history. It has long been recognized for the outstanding quality of its mathematics faculty. Approximately four decades ago, the department undertook the expansion of its mission to include computer science. As that trend evolved, the mission was further enlarged to encompass information technology beyond what is covered under the rubric of computer science. As was well documented in the previous external review, much of this greater scope was handled by the existing mathematics faculty, which enthusiastically and conscientiously went about retraining itself.

Current Status

The department continues to be notable for the quality of its mathematics faculty. A glance at their CVs demonstrates this, showing an abundance of publications, fellowships, and grants.

The computer fields are under-represented in the professorial ranks.

Discussion continues about whether the department should be split in two.

As the department pointed out in its self-evaluation, the faculty age distribution is striking, and cause for concern. The size of the full-time faculty has been decreasing, without any corresponding decrease in registration for the department's course offerings. Inevitably, this has led to greater and greater dependence on adjunct faculty. There have been no new professorial lines in the past few years, and no replacements for those who have left such lines.

Analysis

The importance of Mathematics and Computer Science

It is clear that Mathematics and Computer Science have a crucial part to play in the future of Lehman College. The construction of the new science building bespeaks the college's intent to have a strong program in the sciences. The *Strategic Planning Council Report 2009-2019* makes multiple references to the importance of the STEM disciplines. Both mathematics and computer science play a dual role in the STEM disciplines: each is an important field of knowledge on its own, and each is an essential part of the intellectual and practical infrastructure supporting all the STEM disciplines.

The high quality of the mathematics faculty is Lehman's strongest claim to a position in the first tier of CUNY colleges. Conversely, allowing that quality to dissipate will firmly lock Lehman into the second tier.

Although there is a certain logic to devoting resources to the weakest departments, it is a dangerous strategy. If resources are targeted toward areas of strength, the result is peaks of excellence; if they are regularly concentrated on the weakest areas, the result is uniform mediocrity.

The department needs to grow

The department has inexplicably been starved for resources in recent years. This condition must be improved, quickly. The aforementioned shrinking and aging nature of the department lends some urgency to the situation.

Why it is urgent

It takes much longer to build up the reputation of a department than it does to destroy it. The feedback mechanisms accelerate any downward spiral in reputation, and delay any improvements, as prospective hires are likely not only to avoid a department whose quality, or reputation, is on the way down, but to remain skeptical even as it starts to improve. Thus it is essential to deal with the shrinking and aging before it progresses much further. And, of course, there is more substance to the problem than just the somewhat blurred notion of reputation. Assuming that there will be more hiring in the department, it is still important that such hiring start soon, so that there is enough overlap between the current department elders and a new crop of hires to transfer the accumulated institutional wisdom to a new generation. That wisdom comprises a diverse range of experience including such items as the departmental and college cultures, what works and what doesn't in various courses, and how to work with the institution's processes and infrastructure. If there is not sufficient overlap between the current faculty and a new generation to pass along such expertise effectively and smoothly, the consequences will be even worse than the cost to the department's technical reputation.

The presence of some younger faculty in today's department must not be allowed to diminish the urgency of the situation. While it might be tempting to say that they can solve at least the need for overlap of the generations, such complacency would not be prudent. Unless the rejuvenation of the department proceeds quickly, it is likely that several of the current younger faculty will leave. The Provost observed that the department has been successful in bringing in the best people, but not so successful in keeping them. That is hardly surprising. The best people have more possibilities available to them, and when they observe a department on the decline because it is being denied resources, why would they stay? They were hired because they are smart, and, in leaving, they have passed a type of intelligence test and demonstrated some of the qualities for which they were hired.

The unanimity on this matter, of the faculty and students with whom the review committee met, is dramatic. Every one of them mentioned the need for more regular faculty lines, and many mentioned more courses or more sections of those courses that are offered. Here is a sampling of the comments.

> "Math is starved."

> "Computer Science is in a death spiral."

> "We don't offer enough courses."

> "Without new faculty, nothing can be done."

> "We need several more faculty."

> "We can't offer enough courses."

> "[without more faculty] I expect Computer Science to evaporate."

> "Computer Science needs eight or nine more professorial faculty and three or four more lecturers."

> "80% of CMP and CIS courses are taught by adjuncts."

> "If there are no new lines, I'll go on the job market."

> "Everything is taught by adjuncts."

"Too many courses taught by adjuncts."

"Courses are offered too infrequently."

"There are not enough sections of required courses."

"What will happen without additional lines? Kill the CIS minor. Kill the CIS major. Let CMP die by itself."

How big should the department be

There are a number of factors to weigh, in considering what size the department should be. These include such obvious ones as covering the course offerings and registration, and providing the intellectual "critical mass" to sustain a research community. In that connection, it was noted that it is difficult to be a math major at the college, because course offerings are too sparse. This was reported to the review committee both by faculty and by students. One must assume that such comments apply equally to the computer science major. Less obvious might be the fact that even heavy use of adjunct faculty requires professorial rank faculty as well. Today, not enough faculty are involved in planning and supervising courses taught by adjuncts. This is partly a consequence of not having enough professorial rank faculty, and partly of such planning and supervision not being counted realistically as an intrinsic part of the teaching load. It was pointed out that it is quite possible for a student majoring in Computer Information Systems to graduate without ever having any CIS course taught by a regular faculty member. If the bulk of the department's course offerings are taught by adjuncts, then Lehman College is behaving like a community college!

On average, the department's graduates are not adequately prepared for graduate school, either in mathematics or in computer science. While such preparation is certainly not the only function of the department, neither should a student's choosing Lehman College for mathematics or computer science preclude subsequent graduate study. And, on the brighter side, both students and faculty spoke glowingly of the education received by those students who had extensive interactions with professorial faculty. So, arguably, some of the best students are being well served. However, more courses, taught by more regular faculty, with more time, could extend those benefits to a greater portion of the department's majors.

The interdepartmental program in Computer Graphics and Imaging deserves special mention. It is an innovative direction that might serve as a template for developing other programs. One can imagine creative interdepartmental programs in areas such as educational technology, and health informatics. As technology changes society, Lehman College should be on the leading edge. This means constant experimentation with the curriculum, which, in turn, requires enough high-quality faculty, with enough time. Handled skillfully, such programs could become magnets, attracting students by their unique offerings.

What kind of hiring

The Provost expressed some concern to the review committee about the lack of a hiring plan that focussed on specific fields. That concern is misguided. It is the nature of both mathematics and computer science that any professorial faculty member would be thoroughly competent to teach any undergraduate course in the discipline. Hiring should be guided by other considerations. Foremost among those considerations is the quality of the candidate. Excellence breeds more excellence, both among the faculty and in the student body, both by helping to attract the best people, and by influencing the behavior of the existing stakeholders. The best faculty know how to bring in and mentor good young people, who, in turn, will continue the tradition of bringing in grants and creating innovative programs.

It should be further noted that, in the case of computer science, the discipline continues to evolve and change rapidly. This frequently leads to circumstances in which the "hot topic" of today is one that did not exist a decade ago, or sometimes even two years ago. So it would be pointless to try to come up with a long-range plan for faculty specialties by field. Rather, the approach must be to hire the best people, and take advantage of the fact that one aspect of the best people is their flexibility, including their flexibility when it comes to moving their research and their teaching into entirely new areas. Any other approach would doom the department to being perpetually behind the times. The need for flexibility and adaptability in the faculty argues against growing largely by hiring lecturers. Experience has shown that researchers are far more versatile.

Good new hires bring not only their research and teaching, but also their reputation and contacts. Consider the department's intern program with IBM. More such programs should be instituted. But the department needs professors with stature and outside contacts to initiate such programs. One cannot reasonably expect lecturers to fill that need.

One department or two

The short term

It is worth mentioning the issue of splitting the department, if only to postpone it temporarily. On the one hand, it appears inevitable that the department will eventually split. And, on the other hand, nothing is served by considering this in the short run, since there are not enough computer scientists to constitute a credible and effective department right now. Further complicating the issue is the fact that it is easier to hire excellent people in mathematics than in computer science. In the computer disciplines, academia is competing with industry and its significantly higher salaries.

The longer term direction

The President has been quoted as saying that every department for which there is some accreditation available should obtain that accreditation. This means that the department must work toward ABET accreditation in computer science. This will take some time, but the first steps are clear: expand the professorial faculty in computer science, and provide them with the resources needed. One detail that should be dealt with soon has to do with the way hiring decisions in computer science are made. The review committee was told that there are statutory obstacles that prohibit formal involvement in hiring decisions of faculty not on the Personnel and Budget committee. This constrains a minority sub-department, in this case the computer scientists within the Mathematics and Computer Science department, to second-class citizenship. While that is one of several reasons for the department eventually to split, some tactical amelioration should be found, perhaps by explicit modification of the University's rules about such situations.

Equipment

Independent of the question of one department vs. two is the fact that computer courses and research often needs equipment. Although the review committee was told that the College's IT support is very good, there are requirements that are not best met by the central IT organization. The department has its own computer classrooms and labs. Some courses need hardware such as development systems or prototyping boards. These are not infrequent, extraordinary expenses that must be dealt with on a case by case basis. There should be a regular budget to cover such equipment and people to maintain it.

Governance

The department's decision-making powers have been diminishing. The recent abrogation of the department's chairmanship election was an unseemly power struggle between the department and the administration, one that did not serve the College's objectives. And there have been other ways in which authority has become more centralized, to the detriment of institutional goals.

Naively, it may sound sensible to ask that multiple sections of the same course be synchronized with regard to content and schedule. In practice this leads to teaching for the test. If the majority of courses are covered by adjuncts, that might be the best one can do. However, regular faculty should not only be empowered to use their judgment and experience, they should be encouraged to do so. The higher-level objectives are the real target. Micro-management is not the way to reach those goals.

The quantitative literacy program is another such case. The goal of quantitative literacy is, without question, laudable. The College should look to the Mathematics and Computer Science department for leadership in the technical aspects of what should be taught and how it should be taught, regardless of what departments offer the relevant courses. They are the experts. Broader groups, no matter how well meaning, tend to be amateurs.

More generally, the College should examine the issue of academic governance. Is the College to be run by its faculty, in the manner that defines the word *collegial*, or is it to be run from the top down, like a manufacturing operation?

The CUNY environment

There are several issues which, though they affect the department, are not peculiar to the Mathematics and Computer Science department. They should be addressed, but at a higher level than the department.

The College and the Graduate Center

One consequence of CUNY's separate graduate center is that it draws the focus of intellectual communities organizationally away from the colleges, and physically away from their campuses. While this makes for a vibrant graduate center, it does diminish the colleges' sense of community. Faculty turn to the Graduate Center for colloquia. Often, this means that professors are only on the college campus on the days they teach. Several people mentioned to the review committee that such small steps as regular afternoon teas might improve the cohesiveness of the department. The department could schedule teas on its own, but surely the problem is common to many departments throughout CUNY, and so should be examined at the college or university level.

The requirement that all teaching faculty teach at least one course on the campus has removed some flexibility. Situations arise in which a professor has a teaching load of only one course in some semester, perhaps because of released time or to compensate for overloads in previous semesters. Under these circumstances, the professor cannot take advantage of an opportunity to teach at the graduate center. It would be far better to give departments some discretion in these matters than to adhere to a rigid policy.

Working conditions

Salaries and teaching loads are not competitive with the best parts of academia, nor are salaries competitive with industry. It seems unlikely that there will be any fundamental change in this situation. But measures should be taken to improve the faculty conditions, to try to compete for the best faculty. Perhaps some modest steps such as providing teaching assistants and graders might be possible. More support personnel, department administrators and the like, and more and better equipment are also relatively inexpensive ways to improve working conditions.

Alumni

Greater effort should be made to keep track of graduates. Even ignoring the traditional motivation, seeking donations from alumni, departmental programs could benefit in several ways from feedback about what happens to their majors after they graduate. A department could try to do this on its own, but it would be more appropriately handled at the College level. (And, of course, there is always the possibility that alumni who see that their alma mater takes an interest it what they are doing might be more likely to donate.)

Incoming students

The average level of incoming students leaves something to be desired. The much desired quantitative literacy skills should have been acquired before the students reached college. One of the computer science faculty commented that, in many cases, the students would have been better prepared for the college computer courses if they had not taken their high school computer courses. This was not just a cynical line. It was supported with evidence.

CUNY should be using its influence to get the New York City public schools to do a better job. A plausible direction for such an effort would be to point out, in an appropriately diplomatic manner, the nature of the shortcomings, rather than to get involved in how to remedy those shortcomings (beyond the normal involvement of the School of Education). An "appropriately diplomatic" manner would be low key at first, but becoming louder and more public if nothing improves over several years.

Summary of recommendations

[This section adds no new information. It merely summarizes those recommendations that have appeared, together with their rationales, above.]

At the department level

5.1.1 Hire more professorial-rank mathematicians, quickly, to increase course offerings and maintain the high quality of mathematics in the face of an aging and shrinking department.

5.1.2 Hire more professorial-rank computer scientists, quickly, to increase course offerings an reach a critical intellectual mass, lest computer science evaporate away. Find a way to have existing computer science faculty's voices adequately represented in Personnel and Budget committee deliberations about hiring in computer science.

5.1.3 Limit the fraction of course offerings taught by adjuncts. Count the time spent by professorial faculty in planning and supervising courses taught by adjuncts as part of the teaching load.

5.1.4 Lay out a roadmap to ABET accreditation in computer science.

5.1.5 Encourage innovative inter-departmental programs, like the one in Computer Graphics and Imaging, with released time for developing them and publicity to capitalize on them.

5.1.6 Provide regular equipment budgets where needed.

At the College level

5.2.1 Make governance more collegial.

5.2.2 Be guided by the Mathematics and Computer Science department in promoting quantitative literacy, and make sure that it encompasses computer literacy.

5.2.3 Put in place mechanisms to counterbalance the tendency of the Graduate Center to draw the focus of intellectual communities away from the campus.

5.2.4 Provide some flexibility about the requirement that teaching faculty teach at least one course per semester on campus.

5.2.5 Provide faculty with better support, with more department administrators, more teaching assistants and graders, and more and better equipment.

5.2.6 Systematically track alumni, and let departments know what happens to their majors after graduation.

At the University level

5.3.1 Determine the extent to which faculty working conditions, e.g., salaries and teaching loads, are not competitive, and what the consequences are on CUNY's mission.

5.3.2 Apply CUNY's influence to call attention to inadequacies in the NYC public schools' preparation of students for college.

Outside Review
Department of Mathematics and Computer Science Lehman College

January 2, 2012

Summary

To achieve Lehman's strategic Science, Technology, Engineering, and Mathematics (STEM) goals in both research and teaching, the College must move to

1. **immediately hire young research-active faculty in Mathematics.** Research-active faculty in Mathematics are absolutely critical to bring the power of mathematical thinking and research to achieve Lehman's strategic STEM goals in both research and teaching.

   In the area of research, faculty whose research is recognized by external funding, are essential to build relationships with other institutions, government, and industry as well as growing and sustaining a strong, integrated, collaborative research program in STEM at Lehman College.

   In the area of teaching, students need to experience the power of mathematical thinking within mathematics and its use as a framework for advancing research in a diverse range of other disciplines. Hires in mathematics will support collaborative curriculum development to meet the need for building the mathematical capacity and understanding in a range of disciplines at Lehman College including Business, and Health Sciences, as well as Physics, Biology, Chemistry and other sciences.

2. **immediately hire young research-active faculty in Computer Science** with progress towards accreditation as an immediate goal and with a view towards continued growth and adaptability in both the curriculum and research as the discipline and its applications evolve. These hires are critical to cross-department collaborations for advancing the curriculum and research at Lehman to meet the emerging needs of students and researchers in computer science and other disciplines.

3. **sustain successive rounds of hiring** to achieve the goals in the Lehman's Strategic STEM Plan predicated on the department's demonstrated success in hiring that advances these goals for the benefit of the College and its students.

The current **lack of Professorial contact** is unsupportable. In many courses the overwhelming majority of sections are taught by adjuncts and graduate students. For example, in the math courses taken by science majors in preparation for their major, 89% of the sections were taught by adjuncts and graduate students. In probability and statistics, 100% of the sections were taught by adjuncts and graduate students. Hiring young, researchactive faculty in Mathematics and Computer Sciences is essential. Delivering a high-quality, research-informed curriculum requires the active engagement of tenured and tenure track faculty serving in an ever-present leadership role in developing, sustaining, teaching, and continually reviewing the curriculum and curriculum innovations.

The context in the Department of the **aging of the Faculty** presents the opportunity for hires and reinforces the immediacy of the hires. In the Department, 8 out of 24 (33%) are 70 or older, 11 out of 24 (46%) are 65 or older, and 16 out of 24 (67%) are 60 or older.

Department Self Evaluation

The Department prepared an extensive, detailed Self Evaluation that includes department goals in the areas of teaching, research, and service to the academic community along with curriculum assessment plans.

The challenges pointed out in the self evaluation include

1. **Lack of Professorial Contact:** In many courses the overwhelming majority of sections are taught by adjuncts and graduate students. For example, in MAT 172–175, the math courses taken by science majors in preparation for their major, 89% of the sections were taught by adjuncts and graduate students. In MAT 132 (probability and statistics) 100% of the sections were taught by adjuncts and graduate students.
2. **Aging of the Faculty:** In the Department, 8 out of 24 (33%) are 70 or older, 11 out of 24 (46%) are 65 or older, and 16 out of 24 (67%) are 60 or older.

In the area of resources needed, the Self Evaluation report highlights the need for additional faculty and more support staff to meet instructional and research needs along with additional support for labs for students and for experimental research in computer Science. The Department Self Evaluation mentions plans to apply for external funding in support of realizing the potential synergies among computational biology, bioinformatics, image processing, computer graphics, and numerical, symbolic, symbolic-numerical computations.

Topics Discussed During the Site Visit

A major theme raised in conversation with members of the department was the pressing need of the department for hiring young tenure-track faculty to fill positions recently vacated and to meet the increase in needs of the department due to impending retirements. Conversations with the College administrators focused on reaching clarity with the department on the benefits of such hires. As one faculty member noted a "*good department is descending to mediocrity.*"

Another consistent theme was the topic of splitting the department into a Department of Mathematics and a Department of Computer Science. As noted in the Self Evaluation, the P&B Committee supports this split. Based on the conversations during the site visit, the eventual splitting of the Department is widely endorsed by both faculty and administrators. In the discussion of the timing of the split, the consensus was that the computer science program needs to progress further along the path to ABET accreditation before a split would be appropriate. Faculty and Administrators were all supportive of the goal of achieving ABET accreditation of the computer science program, and all agreed that significant further development of the program is required to reach this goal.

A repeated theme in conversation with faculty was the damage to department morale resulting from the recent removal of Robert Feinerman as chair. It should be noted that everyone was most appreciative and supportive of the work of Robert Schneider in being willing to step into the role of chair.

Conversation with both faculty and administrators touched on the current internship programs— the IBM-Lehman College Internship program and the Department Internship Program run by Katherine St. John.

Faculty reinforced the need for support for teaching and research labs.

The two themes that emerged from conversation with the few students that the outside reviewers were able to meet with was the supportive enthusiasm of faculty along with the lack of availability of advanced courses. This shortage of courses, related to an insufficient number research-active faculty, leads students to leave. The opportunity for advanced course work is critical for retention and is a characteristic of a quality school.

Considerations

The need for immediate hires of tenure track faculty

The value of mathematics for students and for researchers in other disciplines comes from the application of mathematical thinking. Bringing this value to students requires researchactive mathematics faculty.

Research-active faculty in Mathematics are absolutely critical to achieving Lehman's strategic Science, Technology, Engineering, and Mathematics (STEM) goals in both research and teaching. In the area of research, faculty whose research is recognized by external funding, are essential in building relationships with other institutions, government, and industry as well as growing and sustaining a strong, integrated, collaborative research program in STEM at Lehman College.

The Department of Mathematics and Computer Science at Lehman has a strong record of recruiting excellent young faculty where the focus is on hiring the best available young research faculty and then working with them to address specific areas of need. For example, as noted in the Self Evaluation, "a topologist was developing a course in the Mathematics of Finance (before he resigned), a Teichmuller specialist is developing an expertise in Statistics because of departmental needs and a Differential Geometer has just received a $5,000,000 grant in teacher training. Incidentally, all three were recipients of NSF funding in their own areas of research."

This record of success in hiring high-quality, young, research-active faculty together with the strength of the current faculty should be leveraged to the fullest in recruiting additional faculty before the current faculty strength is depleted by retirements.

In addition to bringing the power of Mathematical thinking to the Lehman College research enterprise, research-active faculty are required for their leadership role in developing, sustaining, and continually reviewing the curriculum for its delivery of this power to all students taking mathematics courses. This includes examining each of the bullet points in the recently approved criteria for courses in the area of Mathematical and Quantitative Reasoning in the Common Core Structure for their potential to deliver the power of mathematical thinking to students for their subsequent use in other disciplines.

Immediate hires of tenure track faculty in Computer Science are an essential step in moving the program forward to obtain ABET accreditation. Such hires should also serve to grow and sustain collaborative research and contribute to programs for students that cross traditional discipline boundaries such as the current computer graphics and imaging program in collaboration with the Art Department. As noted in the Self Evaluation such hires could be in computational research with emphasis on scientific computing in collaboration with Biology and Chemistry.

Promoting student success

The lack of professorial contact detailed in the Department Self Evaluation is an unsupportable disadvantage to students. The hiring of new tenure track faculty should, as a consequence, dramatically increase the capacity of the Department for curriculum innovation, assessment of current curricula, assessment of and support effective teaching throughout the Department, and the development of new courses and programs both within the Department and in collaboration with other Departments.

Curriculum innovation and assessment. The Department Self Evaluation mentions assessing all courses, reworking the business calculus sequence, and developing the Department's honors precalculus and calculus courses beginning with the assessment of the lower level courses. The Department should also consider the potential for pedagogical innovations and the possible use of technology in curriculum reform to enhance student learning. To cite just one possibility, it could be that for some students mastery of basic skills can be accelerated and strengthened using some form of computer assisted learning and evaluation. The ALEKS program http://www.aleks.com is one such possibility worth considering. For other courses one might consider whether student learning is increased through more extensive use of in-class, small group, exploration and problem solving. The point here is not to propose a particular process, but to emphasize the need for public discussion, experimentation, and assessment of different approaches to enhancing student learning. New tenure track hires will make an important contribution. They will bring new ideas and perspectives to these conversations and experiments.

Internships and undergraduate research in the decade of science. The Department should consider the possibility of expanding the existing opportunities for students for outof-class experiences including internships and undergraduate research.

In addition to raising the level of student learning, such programs can support

(a) intense commitment to learning and a sense by students of taking charge of shapingtheir future
(b) recruitment and retention of students,
(c) student success in either employment or graduate/professional education at the timeof graduation from college, and
(d) increased engagement of alumni ranging from providing opportunities in their ownorganization for research and workplace experiences for current students to supporting students to participate in such opportunities through donations and fellowships.

Since achieving these goals is of strategic value at several levels—the department level, the level of the School of Natural and the Social Sciences, and at the College level—it makes sense to discuss, at all these levels, the balance of strategic benefits of expanding current programs with resources required to be successful in delivering (a) through (d) above.

It is worth noting that undergraduate research can occur in the context of meaningful work for an organization and is often interdisciplinary. See for example, the student slides available at www.math.neu.edu/prism/2010-fall and www.math.neu.edu/prism/2011-fall. Such out-of-class opportunities could be valuable preparation for and/or a reinforcing sequel to activities such as the 10-week summer training and mentoring program in science funded by the recent Alfred P. Sloan Foundation grant supporting CUNY's Decade of Science.

Recommendations

1. The College should move immediately to hire young research-active faculty in Mathematics. Research-active faculty in Mathematics are absolutely critical to bring the power of mathematical thinking and research to achieve Lehman's strategic Science, Technology, Engineering, and Mathematics (STEM) goals in both research and teaching. In the area of research, faculty whose research is recognized by external funding, are essential in building relationships with other institutions, government, and industry as well as growing and sustaining a strong, integrated, collaborative research program in STEM at Lehman College. In the area of teaching students need to experience the power of mathematical thinking within mathematics and through its use as a framework for thinking in a diverse range of other disciplines. Thus, hires in mathematics will support collaborative curriculum development to meet the need for building the mathematical capacity and understanding in a range of disciplines at Lehman College including Business, and Health Sciences, as well as Physics, Biology, Chemistry and other sciences.

2. The College should move immediately to hire young research-active faculty in computerscience with progress towards accreditation as an immediate goal but also with a view towards continued growth and adaptability in both the curriculum and research as the discipline and its applications evolve. These hires are critical to cross-department collaborations for advancing the curriculum and research at Lehman to meet the emerging needs of students and researchers in computer science.
3. Sustain successive rounds of hiring to achieve the goals in the Lehman's Strategic STEMPlan predicated on the department's demonstrated success in hiring that advances these goals for the benefit of the College and its students.

Submitted by

Richard Porter, Interim Chair, Department of Mathematics
Northeastern University, Boston, MA.

e-mail: r.porter@neu.edu

Appendix F: Prioritization Report from 2014-2015

CS REPORT

Program Summary

Throughout this report we make the observation, at times quite forcefully, that the CS program will require a significant number of new fulltime CS faculty hires. Without such hiring it will be difficult if not impossible, particularly in light of expected attrition and increased enrollments, to maintain current standards, return to standards previously met, or to reach the standards necessary to obtain accreditation. Already for our MS program the standard has seen a significant decline and the BS and BA programs see decline in terms of course variety due to the low number of faculty lines. ABET accreditation for CS is mandated in the school vision statement. New hires are necessary to make ABET certification a realistic goal. While we recognize the constraints faced by Lehman/CUNY we present our case for these new lines in the hope that it will be helpful to administration in their efforts to secure badly needed resources.

Criterion: 1. History & Development - Program History

Question: Program History - Describe your program and its history, including both program content and its students/constituents.

The CS program is in its fourth decade. It began with the dedicated efforts of a small number of mathematicians who recognized that CS, then in its infancy, would be an important field of study. Initially CS could be treated as an extension of mathematics with programming. This is hardly true today. CS no longer lends itself to the model that brought it to Lehman College. CS specialists are an absolute necessity. The pace of change in CS is not so much evolutionary but revolutionary. Faculty must continually retrain and build on prior expertise even to teach introductory undergraduate courses. Hiring of CS specific faculty began in 1999. Today we have eight computer scientists on our faculty.

Faculty engage with students both in class and by involving interested students in their research. Students are encouraged to take on internships, often paid at competitive rates, many of which have led to fulltime employment. For 14 years a series of NSF grants provided scholarships with stipends to many of our students. These scholarships required attendance at on campus open colloquium with invited guests as well as our own faculty discussing research and other advanced topics.

The CS program prepares students for both entry into the job market and for further study at the graduate level. Many of our graduates have secured high paying CS related employment with entities such as IBM, Lockheed Martin, Goldman Sachs, JP Morgan Chase, YCharts and the like. Others have obtained less lucrative fulltime professorial positions.

Question: Strategic Plan - Please provide the mission statement and summarize the strategic plan for your program or your department.

1/7/2015 5:07:04 PM

The CS program strives to meet two main objectives:

1. Provide our students with an education in the CS discipline that will enable them to find gainful employment and/or continue on to graduate studies. Many of our students are from the first generation in their families to attend college. Many come from economically disadvantaged homes. Demand for well-trained computer scientists is extremely high and projected to go ever higher. Student interest in this field has been and is still tempered by the reality that the learning curve is steep as the subject builds vertically and depends heavily on honing problem solving abilities, as well as the unfortunate perception that CS is only for “geeks.” The high demand for and low supply of qualified CS workers as well as the large profits generated by businesses in the field has resulted in generous salaries. This makes study of the CS discipline an excellent path to a middle class income for our students.

2. Support and encourage our faculty in their research programs. These research programs should make significant contributions to the discipline of CS. We further encourage faculty to engage interested students, both graduate and undergraduate, in their research programs. The latter can provide students considering graduate school valuable experience.

A third objective has been informally discussed, whereby faculty engage in CS related entrepreneurial ventures with students. Courses have already been offered covering smartphone/tablet programming. A skilled team of programmers can easily produce and market such products to a large audience.

Question: Program Evolution - How have the needs of your constituents evolved over the past 3 years? How has the program adapted to these changes?

CS has not so much experienced evolution as multiple revolutions. We have had to make changes in our program continually over the years. It has been regularly necessary to reconstitute the subject matter of standing courses, create new courses, and guide students toward and away from a selection of specialized topics that have grown and shrunk respectively in importance. A recent overhaul of the major added Networking and Database Systems and removed Programming Languages as requirements of our B.S. in CS starting September 2014. This was due to the fact that the related skills have become highly sought after by employers. For strategic reasons owing to the limited size of our faculty we needed to make these courses requirements of the major in order to be sure we could offer them regularly. We have also offered for the last three semesters courses in smartphone/tablet programming. Additionally, we've offered courses in Graphics Processing Unit (GPU) programming. These courses cover topics that have become very hot recently, but that did not exist a few short years ago. We anticipate many future changes that will steer our curriculum. Cyber Security, Virtual Reality, Big Data, and the Modeling of Scientific Phenomena are just some areas where CS should see tremendous growth in both research and employment opportunities according to agencies such as NSF, NIST, and the Bureau of Labor Statistics to name a few.

Question: Curriculum Map - Please submit your 4- year undergraduate curriculum degree map. (View Instructions)

Criterion: 2: External - External Demand for the Program

Question: External Demand - Describe the external demand for the program from (a) potential students/constituents, and (b) for graduates of this program among employers and graduate schools?

CS is a field of problem solving, critical thinking, quantitative reasoning, and reasoning in general within the STEM disciplines. The courses offered by the CS program help students develop skills in these areas, which are in high demand by employers not only in computer related fields.

A B.S. in CS can be a terminal degree. The employment situation is such that well-trained CS undergraduates can find well-paying jobs that are not dependent on graduate studies. We are mindful of this and try to offer courses such as Software Engineering and Smartphone/Tablet programming that will benefit students who choose to follow this track.

Our program also prepares interested students for graduate studies in CS. In this vein we offer courses in Algorithms, Graph Algorithms, Game Theory, and other theoretical topics.

Demand for double majors including our program has materialized in recent years. Students from other STEM fields, for example Physics, have taken to learning CS, because more and more scientific inquiry involves creation of computer based models. With the advent of cost effective virtual reality this phenomenon is predicted to explode.

Many states including New York State are moving forward with plans to make CS either an option to replace other requirements or an additional requirement for high school graduation. This requires certifying teachers in CS and will create new demand for CS courses. Further, if done right K-12 exposure to CS could remove some of the "geek" stigma associated with CS and hopefully spur interest among students for CS.

Question: Unmet Demand - How does the program (and at what level) meet external demand? Is there evidence of unmet demand? Please provide examples or identify the criteria used to reach your conclusion.

Our CS program is significantly understaffed. For Spring 2015 less than half (8/19) of CS sections offered will be taught by regular fulltime CS faculty. The situation is far worse in our related CIS program where exactly one section (1/19) will be taught by a regular fulltime CS faculty member.

Negative feedback lowers demand for our program. The faculty shortage lowers our course variety below the level certification requires and compared to other institutions. We have heard from students that their friends and family members have chosen City College over Lehman because of this lack of course variety.

Many signs point to a new wave of demand for CS courses. Word is out that there are many highpaying jobs in the field that don't require graduate degrees. Throughout the United States enrollments have risen dramatically. Even considering the upswing in CS enrollments, the Bureau of Labor Statistics predicts an excess of 1 million new CS jobs in the U.S. over the next six years, i.e. 1.4 million new CS jobs, but only 400,000 new CS graduates. Lehman has seen less of an uptick in enrollments than other institutions. We believe that the aforementioned lack of variety plays a role here. In the 1990s, the trend of rising CS enrollments reached Lehman, albeit a few years later than the national average. We believe this pattern is now repeating.

Additionally, although not yet in place, when CS becomes a requirement in high schools we will be hard pressed to meet the demand created for CS courses.

Question: Competition - To what extent is this program competing with programs at other colleges? Which programs at CUNY and in the surrounding region would be considered competitors? Briefly describe the uniqueness of your program compared with those offered by competing institutions?

CS programs are offered at most colleges. Within CUNY we know from firsthand accounts that City
College is drawing away students who would otherwise naturally come to us. We suspect that Hunter College may also play such a role. One of the things that the competition offers that we cannot is more variety in CS elective courses. We believe that City and Hunter may also have an advantage in terms of brand recognition.

Many students who choose our campus over other CUNY campuses for a CS degree do so because of proximity to their homes and/or places of employment, though we do have quite a few who have chosen our CS program at Lehman over the others based on word of mouth reporting from our students. Such students have heard of our other draws. Our department offers smaller class sizes and more personal attention than these other campuses. We had a very prestigious paid internship program with IBM Research for 15 years. IBM has never accepted undergraduates from any other institution for research internships. We have a new paid internship program with IBM's TPF program that should not only offer students experience, but is expected to lead to many job offers from IBM just as the research program has. New contacts with Goldman Sachs have led to paid internship offers and regular employment for some of our students. For 14 years many of our students received department scholarships including stipends from a series of NSF sponsored grants.

Question: Program Recruitment - What domestic and international recruitment strategies and admissions criteria does your program use to attract students or other constituents into the program?

The Department of Mathematics and Computer Science works with FYI and SEEK each semester to provide advisement to incoming freshman. The fundamental purpose of the advising process at FYI registration is to place students in the correct math class. CS advisers have used this opportunity to recognize students who are prepared to begin the CS program and encourage them to do so. Other students who show an interest are advised how best to begin their pursuit of a CS degree. FYI has not over the years offered block programs incorporating CS courses. When a CS advisor attending FYI registration discovers a student interested in a CS major, special accommodations are made by FYI. Unfortunately, not all advisors from our department have been CS faculty and students have slipped through the cracks at times only to start the CS major in their Sophomore year. This makes completing their degree in 4 years difficult even for students who enter Lehman well prepared. To promote the CS program, department representatives have attended major fairs sponsored by Lehman administrative programs. Our representatives have also visited high schools in the Bronx to talk to students about CS careers and have attended tech fairs such as one recently sponsored and hosted by Microsoft at its Time Square Offices for high school students considering CS as a major in college.

Recently, the department hired its first fulltime undergraduate adviser. The adviser has been and will continue to expand upon these efforts.

Criterion: 3: Internal - Internal Demand for the Program

Question: Internal Demand - Explain the trends in the internal demand for your program using the data shown in the following tables: (View Instructions)

Institution Provided Data

Headcount Majors Undergraduate

	DEGREE NAME	FALL 2011	SPRING 2012	FALL 2012	SPRING 2013	FALL 2013	SPRING 2014
Undergraduate	BS	84	88	85	89	103	123

Hunter's CS Chair reports that CS majors have approximately doubled in a steady ramping up since 2011. He and other CS chairs throughout CUNY are concerned about their inability to staff classes even with adjunct faculty because of the sudden massive influx of students into their programs. Similar circumstances exist in CS programs around the nation.

We see an increase in CS majors at Lehman of approximately 40% from Fall 2011 to Spring 2014. The CS program at Lehman College is behind the trend as it was in the late 1990's. The current increase occurs almost exclusively at the tail end of the period and is not as dramatic as elsewhere. It is expected that demand for CS will only grow from here and there is every reason to expect that Lehman College will participate in the growth, if for no other reason than because of spill over from other programs that can't handle the volume. Further, unlike the dot com bubble that led to the previous increase in CS enrollment followed by bust, the tech industry has greatly matured so that many of the premature predictions of computers and therefore CS jobs everywhere have now come to pass. The expectation of many is that there will not be the same sort of bust this time around. Adding fuel to the fire are the already mentioned initiatives in many states to certify teachers to teach CS in high schools.

Question: Services to Other Programs - Describe ways in which your program is: (a) relied upon by other programs on campus; (b) dependent on service courses or other resources offered by other programs? (View Instructions)

The program in CS services other Lehman College programs. Mathematics majors are required to take a single CS course but are allowed to make substitutions of CS courses for mathematics courses and are often advised that it is in their best interest to take more CS courses in the form of a minor. Computer Graphics and Imaging students are required to take several CS courses. Physics and Biology students take CS courses. There is a great and growing need for these and other STEM majors to acquire the skills needed to build computer based models of the systems they investigate. Business and Economics majors can also benefit greatly from CS training.

Our program includes several mathematics courses in the major. Required are Calculus I and II as well as Linear Algebra, while Calculus III can be substituted for a CS course. We also require an elective Mathematics course of 300 or higher level. We encourage students to take additional mathematics if they intend to pursue graduate degrees. Although not a requirement, we also encourage students to take a year or more of Physics, Chemistry, and/or Biology as there are many additional opportunities for computer scientist with interdisciplinary skills within the STEM fields. Business and Economics courses offer similar benefits to CS majors. A handful of our students have wound up doing second majors in one of these fields.

Question: Adaptation to Constituents - What steps has the program taken in the past 3 years to reflect (a) the changing needs of its constituents, (b) technological advances, and (c) to develop innovative and forward-looking curricula?

CS is not only at the forefront of technological advances; it is the driving force behind many technological advancements. Our program is often the first to be aware of and to make use of new technology. As previously stated our program undergoes radical change on a regular basis because it is by its nature deeply invested in technology and we cannot afford to be left behind. There can be no CS program without such rapid change. We regularly reconstitute the subject matter of standing courses, create new courses, and guide students toward and away from a selection of specialized topics that have grown and shrunk respectively in importance. Our faculty need to keep abreast of the rapid changes in the field not only for our research programs and to teach graduate and upper division undergraduate courses on cutting edge topics, but even to teach our introductory courses. We have offered for the last three semesters courses in smartphone/tablet programming. Additionally, we've offered courses in GPU programming. These courses cover topics that have become very hot in recent times, but that did not exist a few short years ago. We anticipate many future changes that will steer our curriculum. Cyber Security, Virtual Reality, Big Data, and the Modeling of Scientific Phenomenon are just some areas where CS should see tremendous growth in both research and employment opportunities according to agencies such as NSF and NIST.

Question:

Unmet Demand - How does the program (and at what level) meet internal demand? Is there evidence of unmet demand? Please provide examples or identify the criteria used to reach your conclusion.

Our CS program is very significantly understaffed. For Spring 2015 less than half (8 of 21) of the CS courses offered will be taught by regular fulltime CS faculty

The lack of variety mentioned in the related response in the External Demand section troubles us here as well. Many students hoping to take a course in for instance Artificial Intelligence in the last few years have been disappointed. Further, even if we offered such a course in the near future, there would be next to no chance that we could afford to offer a follow up course that delves deeper into the subject and/or specializes within the area. This latter concern is tied to both the staffing issue and a curricular issue. Course requirements for CS need to be increased. Students comparing our requirements to those of City College are struck by the immense difference. City's CS program is housed in an engineering school ours in a liberal arts school and so we would not look to close the gap completely, but our CS program once properly staffed does need to increase requirements if we hope to ever get certification and if we want to be able to offer concentrations within the major that take students deeper into subjects they've already explored.

Students who take a double major of Mathematics and Computer Science find that the lack of variety slows their progress toward graduation as it makes it difficult to take enough unique classes in the prescribed timeframe.

Criterion: 4. Support and Staffing - Program Support and Staffing

Question: Program and Staffing Trends - Describe and/or explain the departmental staffing distribution for this program as presented in the data below: (View Instructions)

Institution Provided Data

Number of FT Faculty Tenure / Tenure Track (by Department)

	FALL 2011	FALL 2012	FALL 2013
TOTAL # OF TENURE / TEN. TRACK	25	24	22
DIST. & FULL PROFESSORS	17	16	14
ASSOCIATE PROFESSORS	2	2	3
ASSISTANT PROFESSORS	3	2	2
DIST. LECT. &LECT. CCE BRKDWN	3	4	3

Number of Adjunct and Substitute Professors

	FALL 2011	FALL 2012	FALL 2013
Number of Adjunct	52	33	35
Substitute	6	6	5

Number of Program Staff

	FALL 2011	FALL 2012	FALL 2013
HEO Series	2	2	3
NON-Teaching Adjuncts	0	0	3
College Lab Technitians	0	0	0

The Department of Mathematics and Computer Science has two core groups of faculty, i.e.
Mathematicians and Computer Scientists. Not long ago there were a number of mathematician who leaned towards CS and taught many CS courses. Only one of these mathematician/computer scientists remain. For the Spring 2015 semester the CS program has nine regular fulltime faculty who can teach CS courses. In Fall 2011 we had eleven. Of our current nine, three are eligible to retire. One is a Distinguished Lecturer who has been with us for more than ten years. When a new contract is agreed to it is quite possible that the contract will eliminate this line. Two of these faculty are untenured assistant professors.

Adjuncts and substitutes teaching CS have risen though the number teaching other subjects, i.e. mathematics, CIS, and CGI, within the department has fallen.

The trend for Program Staff numbers on the chart is misleading. Prior to Fall 2011 we had three HEOs and were expecting to add a fourth member to our staff in the form of an executive assistant to the chair. Two of our long time HEOs suffered debilitating illness and then passed. Their positions were filled, but the new position which was supposed to ease the chair’s effort dematerialized.

The three non-teaching adjuncts appearing in the third table is a surprise to us. We are unaware of who these individuals might be and what service they might provide.

Question: Staffing Contributions - What are the roles and contributions of part-time or adjunct faculty and professional staff to this program? How are they supported and integrated into the program?

Adjunct faculty are simply responsible for teaching the courses to which they are assigned and holding office hours to meet with their students if required.

In addition to our excellent administrative assistant we have staff who perform many other task related to running our tutoring services and keeping our computers running. Due to space limitations details can be found by referring to "Support and Staffing:Staffing Contributions" in the reports for the CS BA and CIS BS.

MORE TO BE ADDED. SPREAD OUT MARVIN AND LORIC IN TO OTHER PROGRAMS.

Question: Student Preparedness - Use the tables below to describe the academic preparedness of the constituents who enter the program. (View Instructions)

Institution Provided Data

ACADEMIC PLAN UNDERGRADUATE FRESHMAN

	FALL 2011	SPRING 2012	FALL 2012	SPRING 2013	FALL 2013	SPRING 2014
FULL TIME	30	33	22	24	26	34
PART TIME	2	2	7	10	8	8
TOTAL:	32	35	29	34	34	42

ACADEMIC PLAN UNDERGRDUATE TRANSFER

	FALL 2011	SPRING 2012	FALL 2012	SPRING 2013	FALL 2013	SPRING 2014
FULL - TIME	36	39	38	36	48	56
PART - TIME	16	14	17	18	19	20
TOTAL:	52	53	55	54	67	76

ACADEMIC PLAN UNDERGRADUATE UNKNOWN

	FALL 2012	SPRING 2013	FALL 2013	SPRING 2014
FULL - TIME	1	1	1	3
PART - TIME			1	2
TOTAL:	1	1	2	5

Students seeking a degree in CS should have completed pre-calculus and be well prepared to take
Calculus I. Ideally, students entering a CS program would have excelled at AP Calculus and/or Computer Science. However, the latter group of students form an almost empty set and the former are far too rare at Lehman. Students who enter the CS program with a mathematics deficit face mathematics and CS subject matter that builds vertically. Too many are burdened with College Algebra as their starting point. This pushes Calculus I into their sophomore years. Although it only delays Programming I by a semester, College Algebra is a less than ideal choice of prerequisite for Programming I. Both the mathematical maturity gained and subject matter covered in Pre-calculus make it a much better prerequisite. These issues

conspire to make it very difficult to map out a program of study that can be completed in 4 and even 5 years. Sadly, many very bright but underprepared students who could have done very well in CS are discouraged by this prospect.

Computer Programming is unlike other subjects students have experienced. The degree to which problem solving, critical thinking, and intuition come into play and the mechanisms through which they must be expressed are totally foreign to any but the initiated. This can catch students off guard as they enter this world previously unknown to them. A more gradual or self-paced and fun introduction to programming could benefit some, but this would delay graduation even further.

Question: Program Success - How does the program define student/constituent success? What support structures are in place to facilitate students/constituents?

The program has formal measures of success enumerated in objectives and goals in the syllabi for each course and for the program as a whole.

However, a true measure of success is seen in the career paths of our graduates. There are many Lehman College CS graduates who have gone on to high paying and rewarding careers in CS. An informal email based survey conducted by our advisor recently has found our students employed by entities such as IBM, Lockheed Martin, Goldman Sachs, JP Morgan Chase, YCharts, CUNY, etc. The vast majority of those surveyed graduated within the last 10 years and have salaries that would make many Lehman College faculty members jealous. Our long term internship program with IBM has also provided validation for our efforts on behalf of our students as do other initiatives like the new relationship now being forged with Goldman Sachs.

The truth is that two hour tests can only measure a lower bound on student abilities in CS, but certainly not abilities at levels employers demand. Also, many projects assigned for outside of class are too easily "outsourced." We as well as employers cannot gauge the extent of student abilities based on these criteria from which minima on grades are generally established. Instead to recognize success we must look to see if students are stimulated and excited about the work they are engaged in. We must hear them pose interesting questions and see students work well collaboratively. We are witness to these things.

Criterion: 5. Quality of Program - Quality of Program

Question: Program Assessment - Discuss how program assessment results have been utilized to improve student/constituent learning outcomes.

Informal discussions spanning the curriculum are a regular occurrence. Formal assessment has been limited to our introductory courses.

Four years ago we changed languages from Java (more difficult syntax) to Python (simpler syntax) in Programming I while retaining Java for more advanced classes, instituted a common syllabi, homeworks, quizzes, exams, and schedule for Programming I. Last semester the number of exams was lowered and quizzes raised.

This all in an effort to raise the number of students who achieve success in Programming I.

There is debate among the CS faculty as to just what the result has been. At least one who teaches introductory programming believes there has been success, there are others who believe otherwise or aren't sure. Among those teaching higher level programming courses there seems to be agreement that students, even the best students, are coming to them less prepared than before the changes.

From our 2011 external review which was conducted just after the new policy was put in place: "Naively, it may sound sensible to ask that multiple sections of the same course be synchronized with regard to content and schedule. In practice this leads to teaching for the test. If the majority of courses are covered by adjuncts, that might be the best one can do. However, regular faculty should not only be empowered to use their judgment and experience, they should be encouraged to do so. The higherlevel objectives are the real target. Micro-management is not the way to reach those goals."

Question: Program Outcomes - If your program is subject to external accreditation, describe the results of your most recent accreditation review; If your program is not externally accredited, then please provide the results/recommendations and implementation plan from the most recent self-study.

CS is not currently accredited, but future ABET accreditation is a goal.

Both the most recent self-study and external review concur on many points, but none so strongly as the fact that CS is dramatically understaffed. A quote the external reviewers included in their report on this point: "Computer Science needs eight or nine more professorial faculty and three or four more lecturers." Such a request may not be pragmatic given the universities constraints, but it serves to underscores the dire situation CS finds itself in.

CS is not only understaffed, but lacks necessary support staff and equipment. The external reviewers agreed: "Some courses need hardware such as development systems or prototyping boards. These are not infrequent, extraordinary expenses that must be dealt with on a case by case basis. There should be a regular budget to cover such equipment and people to maintain it" and our internal report states that "The current staff supports the calculus labs, labs for statistics, and the computer applications program but has no time to support the unique needs of a dynamic teaching curriculum or active research program." "Additional technical support, via fulltime College Lab Technicians or HEO's is needed."

Finally the reviewers point out that "Both mathematics and computer science play a dual role in the STEM disciplines: each is an important field of knowledge on its own, and each is an essential part of the intellectual and practical infrastructure supporting all the STEM disciplines."

Question: Teaching Effectiveness - What actions has your program taken to improve teaching effectiveness, student retention and student/constituent success, faculty scholarly and
creative output, and alumni placement? What were the outcomes for each?

As previously noted, CS faculty regularly discuss teaching strategies, teaching resources, as well as subject matter. These have in fact been the topics of a long standing lunch time discussion as well as the most common topics discussed whenever two or more CS faculty are spotted congregating.

Teaching observations are taken very seriously and are often followed up with discussion beyond the required report and conference.

Student evaluations and comments are given their due consideration as well.

In our effort to support our students every effort is made to put not only a technically qualified instructor in each classroom, but someone who is also an effective teacher.

Our internship advisor has had great success placing students and as previously mentioned many of our graduates are successful in the CS field.

A new initiative between the NNS Dean's office, Physics, Chemistry, Mathematics and Computer Science is setting aside space in Gillet Hall for students in the STEM disciplines to work, relax, and interact comfortably with other STEM majors. This should be a valuable space for our commuter students.

Question: Notable Achievements - Please identify and/or describe the most notable achievements of your students, past and present (for example: awards, fellowships, scholarships, publications, etc.)?

A CS student was one of two programmers (the other was from CCNY) on the team that won the 2014 Annual National Hackathon of the National Society of Black Engineers. This was an open event featuring 1200 competitors from universities across the nation that was sponsored by some of the biggest names in Tech and Banking.

A CS student founded the Lehman chapter of the National Society of Black Engineers, hosted a wellattended "Hackathon," and has multiple job offers from companies including Boeing and Goldman Sachs.

Many of our CS graduates work for IBM (so far 19 have indicated such). Many more work for notable corporations of similar stature as already mentioned. Here we add CNN, the NBA, Turner
Broadcasting, IHG, Internap, Jet Blue, Software Group, and for TV personalities including Dr. Oz and Anthony Bourdain. Some have had entrepreneurial success as well.

Numerous CS program graduates have obtained or are pursuing Ph.D. degrees in CS. Several obtained tenure track and/or other professorial positions. Some have obtained tenure and we are aware of one who is now his department's chair =>

Many of our CS graduates have in addition to their day jobs taught as adjuncts within our program.

Unfortunately, we have not kept track of our graduates. Much of the information above and in other comments concerning alumni was collected in an email survey conducted in the last couple of weeks by our advisor using a small number (less than 40) of email addresses scrounged from faculty.

Question: Retention and Persistence - Please describe the strategies adopted by the program to promote student recruitment, retention, persistence, and graduation?

The program is fortunate to have support from the Math and Computer Science Learning Center (MCSLC). The MCSLC does an outstanding job promoting, tutoring, and aiding CS students. Its staff comprises well-trained CS majors committed to helping their peers succeed. Aside from routine one-onone and group tutoring, the MCSLC offers workshops for targeted courses based on student demand.
These courses include College Algebra, Calculus I/II/III, Programming I, and Discrete Mathematics. The center's director is an active member of our department and proud CS Alumnus; he teaches courses, mentors students, and works closely with faculty. Because his finger is always on the pulse of our students, he is an invaluable liaison between students and faculty.

The department recently added another helpful resource: A fulltime adviser. With faculty, the adviser will contribute to student recruitment, retention, and graduation by being a constant and reliable source of information. Because the adviser is available for students each day, the department has already seen a shortening of its still too long advising lines. This, we hope, will encourage students to see an adviser/faculty-mentor more often and ultimately lead to more program graduates. The advisor is a Ph.D. student in Mathematics which makes her an even more valuable asset to our mathematics students. For a department our size, housing two quite different subjects, ideally there would be one advisor dedicated to mathematics and one dedicated to CS. This would shorten the lines further and permit the advisors to attend to neglected tasks such as tracking alumni.

Question: Public Service - Describe the contributions made by your faculty and professional staff to the surrounding community and to the profession.

CSM Scholarship Program (a program to mentor students in Computer Science and Mathematics), Principal Investigator

Bronx High School of Science Intel Science Competition, Faculty Mentor

Chase Small Business Develop Center (Fordham Rd and Jerome Ave) Multiple presentations on how to get a small business website up and how to add content.

Participation in College Now.

Career Day visits to Bronx High Schools and other venues to promote awareness of CS as a career.

Mentors to various clubs.

Criterion: 6.Size and Productivity - Program Size, Scope and Productivity

Question: Faculty Teaching Productivity - Explain the trends in faculty teaching productivity for your program as measured by department work flow summaries provided in the attached PDFs. (View Instructions)

The department workflow summaries indicate that the student head count has grown over the last two years. However the number of regular fulltime faculty capable of teaching CS has decreased.

It is imperative that that faculty be hired for our CS program.

Question: Research and Scholarly Activities - Please describe how active and engaged are your faculty leaders and professional staff in the following areas: Number of refereed publications; Number of books, novels, etc.; Number of edited volumes; Number of chapter contributions to books [monographs]; Number of juried shows/performance;
Number of editorial positions; Number of professional conference presentations;
Number of non-refereed publications. (View Instructions)

The CS faculty, one Distinguished Professor, two Full Professors (one of whom is now on Travia Leave), one Associate Professor, three Assistant Professors (two of whom have not yet completed a second year at Lehman), one Distinguished Lecturer, and one Lecturer as a group have over 450 professional publications in the form of original research articles appearing in internationally known academic journals or proceedings of peer reviewed conferences. (Note: Unlike many scientific fields, CS publishing is atypical in that conference publications are peer reviewed and often accorded at least as much attention and prestige as journal publications.) Other publications in excess of the 450 plus alluded to above include at least 4 books and 22 book chapters, plus10 patents. Over 20 Ph.D. students have been successfully advised/mentored through completion of their degrees.

These researchers have obtained many grants from the NSF and other funding agencies.

One of our faculty has been named a Fellow of the American Mathematical Society.

Question: Student Success - Use the tables below to (a) describe the number of students/constituents who have completed your program and (b) the total number of credit hours taken to graduate. (View Instructions)

Total_no_of_graduates

	2010-2011	2011-2012	2012-2013
Total number of graduates	14	19	19

Graduates_as_percent_of_total_student_headcount

	2010-2011	2011-2012	2012-2013
% graduated of total headcount	0.61	0.79	0.75

Graduation_headcount_for_undergraduates

	2010-2011	2011-2012	2012-2013
First Time Students	1	9	9
Transfers	12	10	10
Other	1	0	0

Credit hours taken to graduate - Undergraduate First Time Students

	2010-2011	2011-2012	2012-2013
Under 140 credit hours	1	8	7
140 credit hours or more	0	1	2

Credit hours taken to graduate - Undergraduate Transfer Students

	2010-2011	2011-2012	2012-2013
Under 140 credit hours	6	8	7
140 credit hourrs and above	6	2	3

For the 3 academic years starting with Fall 2010 and ending with Spring 2013 our CS BS program had 52 graduates. Of these graduates, 37 completed under 140 credits, while 15 completed more than 140 credits.

For the 3 academic years starting with Fall 2010 and ending with Spring 2013 our CS BA program had 9 graduates. Of these graduates, 3 completed under 140 credits, while 6 completed more than 140 credits.

For the 3 academic years starting with Fall 2010 and ending with Spring 2013 our department had 61 CS graduates. Of these graduates, 40 completed under 140 credits, while 21 completed more than 140 credits.

The 21 completing more than 140 credits is not unexpected due to the fact that some of our majors take on double majors.

Criterion: 7a. Revenue - Revenue and Other Resources Generated by the Program

Question: Enrollment Revenue - Use the tables provided to describe the revenue generated through enrollment in your program's courses. (View Instructions)

ENROLLMENT REVENUE BY DEPARTMENT (in $ amount)

	FALL 2011	SPRING 2012	FALL 2012	SPRING 2013	FALL 2013	SPRING 2014
GRADUATE	187933	112907	121874	162660	162855	108185
UNDERGRADUATE	2145814	2084942	1968546	1929377	2065130	2144218
TOTAL	2333747	2197849	2090420	2092037	2227985	2252403

ENROLLMENT REVENUE BY DEPARTMENT FOR SUMMER & WINTER (in $ amount)

	SUMMER 2011	WINTER 2012	SUMMER 2012	WINTER 2013	SUMMER 2013	WINTER 2014
AMOUNT:	491642	12825	430481	22560	435169	11250

Other Revenue (by Department)

	2012	2013	2014
OSRP - Research programs	403711	985504	347393
Lehman Foundation/Fundraising			
Academic Excellence Fees			
Tuition differential (SWK)			
Laboratory & Material Fees			

The table shows that enrollment revenue has increased from 2,090,420 to 2,252,403 during the period starting Fall 2012 and ending Spring 2014. This represents an increase of 7.7%.

Question: Program Revenue - Please describe the activities or strategies used to increase program revenue from enrollment?

Not applicable.

Question: Other Revenue Generated - Please identify and describe the extent to which the program is supported through internal (Lehman or CUNY) and external (local, state, federal, or foundation) funds and grants for research, programs, equipment, and other sources (such as fundraising, excellence fees, laboratory fees, ticket revenue, materials fees, social work tuition fees, etc.)? If your other revenue is not listed below then use the table in question 7a.1. (View Instructions)

In the last 5 years, as a group the CS Faculty have $4.3 million in external grant funding.

Most professorial faculty have intermural funding, but the magnitude of such funding is insignificant compared to the above.

Question: Instituional Advancement - Please list and/or describe any development or advancement funds (or other gifts) received by or for the program. Please provide, if available, the development/advancement strategic plan or case statement(s) for the program. (View Instructions) Not applicable

Criterion: 7b. Expenditures - Expenditures Associated with the Program

Question: Cost Trends - The three-year trend for costs associated with the program is provided below. Please provide your analysis of this trend and likely future trends. (View Instructions)

Program Cost on PS Reguar, PS Adjunct PS Temp Services (by Department)

	2012	2013	2014
PS REGULAR	2842455	2787894	2720070
PS ADJUNCT	996191	693571	738519
PS TEMPORARY	63604	86353	79126
TOTAL	3902250	3567818	3537715

PERSONAL SERVICES:

Program Cost on OTPS (by Department)

	2012	2013	2014
NPS SUPPLIES AND MATER. (80120)	2049	1841	2679
NPS MISC CONTRACT. SERV(80122)			
NPS TRAVEL (80121)			
NPS EQUIPMENT ACQUIP(80123)	193		
FRINGE BENEFITS(80124)			

Program Cost Total on OTPS (by Department) (Sum of OTPS Above)

	2012	2013	2014
TOTAL OTPS	2242	1841	2679

Other

	2012	2013	2014
Academic Excellence Fees			
Tuition differential (SWK)			
Laboratory & Material Fees			

Costs have gone down over this period of time. A major factor in this is the recent large number of retirements of regular faculty. We expect this to continue for the next few years. It is imperative that we hire active, strong research faculty to replace these retirees. This is important for good teaching and also to maintain our high level within the mathematical community and to build our reputation in the CS community.

Question: Program Efficiencies - Please elaborate on the current level of departmental budgeting and expenditures relative to current demands. What mechanisms are in place to request and what policies are used to justify additional funding?

Competition for good mathematics and computer science faculty is intense. Given the PSC-CUNY contract constraints we need to go above scale and give startup money to hires. The mechanisms for deciding salary offers seems at present to be ad hoc. Establishing a rubric before the next round of hiring would be very helpful.

Supply money from the college is barely able to cover necessities. Student fee money is the main source for student equipment. Faculty make requests which are reviewed by the chair and sent to the college committee. Unfortunately this is a slow process and does not allow for rapid acquisition of state of the art tech equiptment.

The department has a history of grants for equipment.

Winter and summer sessions provide some additional revenue.

Question: Professional Development - Please identify and describe the contributions and sources for professional development activities (i.e., teaching expenses, research expenses, conference travel, reassigned time) provided to the program. Describe the level of participation by the program in developing this budget? What proportion of professional development costs are covered by the program? What portion is contributed to by faculty?

The department has accumulated funds that are earmarked for travel to conferences by faculty for professional development. These have alleviated the costs of faculty for both research and pedagogical travel. The faculty is given a max of $1,000 a year and has to use their own or other resources to cover the extra costs.

The faculty of the department has a history of external funding that has given them money for development and released time for research and pedagogical programs. Please see the grant profile of the faculty.

Question: External Mandates - Please identify or describe any/all external mandates that may affect the cost of the program in the next two years?

The department is mandated by the school's vision statement to become accredited in CS. This cannot happen without a large increase in professorial faculty in CS.

The "Strategic Plan for Stem" has a goal to:

"4. Seek accreditation from ABET (Accreditation Board for Engineering and Technology) for Computer Science …".

The mission statement includes:

"Recognized for small classes, close interaction between students and faculty, a successful Teacher Academy and Honors College, and a caring and supportive environment, Lehman College will celebrate its fiftieth anniversary in 2018 as the college of choice in the region, committed to preparing students for graduate studies, professional careers, and lifelong learning."

These mandates external to the department will require that a significant number of new fulltime CS faculty are hired.

Criterion: 8: Opportunity Analysis - Opportunity Analysis and Impact of the Program

Question: Program Aspirations - Please identify your aspirational peer programs, and provide rationale for their selection? Please identify and describe the metrics (measureables) required to bring this program up to that aspirational level, over what length of time, and what departmental and institutional cost/ investment.

CS would like to make collaborative research efforts within the program more common. Currently, only two members work in the same research area. CS would like to obtain ABET accreditation. CS also would like to be able to offer a larger and more appropriate variety of electives and add concentrations within the major as other institutions do, including nearby CUNY peers/competitors CCNY and Hunter College. These important improvements to our program are only possible with the hiring of a significant number of new fulltime CS faculty. CS experts will typically require high salaries because they are in demand by industry, Our current faculty are underpaid, some severely so by CS standards, but make due largely because they believe in the mission of the CS program at Lehman College. It becomes hard to see how that mission is accomplished without additional hiring. While we hope it doesn't come to this, an anonymous faculty quote noted in the recent external review report drives home the point that CS faculty have plenty of more lucrative options if they see that the program will be left to founder and that this is a very serious matter at what looks to be a time of tremendous growth in the field. "If there are no new lines, I'll go on the job market."

Question: Internal and External Opportunities - What internal and external opportunities can be leveraged to strengthen this program or help it develop in new directions?

As mentioned already, the key to all is the hiring of significant numbers of CS faculty.

CS will continue to look for partners to provide students with internships and job opportunities.

Collaborative research opportunities with industry as well as other departments within Lehman College need to be explored as well.

Question: Technology - Please identify or describe any technological innovations that might be used to strengthen this program.

Technology and technological innovations are what our program is all about. The rapidity with which CS changes can in the extreme make technology obsolete to the CS program even before other programs know of its existence. In essence CS needs the flexibility to move fast in many of its technology decisions.

To catch the virtual reality wave, it is time to invest in headsets like Oculus Rift, Project Morpheus, Gear VR, and/or others. Some Kinect sensors and Zvr display as well.

One long standing need has been to have additional labs where experimentation can take place with staff that can maintain and prep the machines with whatever setting/software/hardware the faculty and students need. If not for Lehman's CS program it would be unheard of for a CS program not to have such labs. Classroom labs, locked down so that students can't do anything out of the ordinary are perfect for the Carmen Computer Center and many classes in Gillet, but not for CS faculty and students working on the cutting edge.

A small number of new 1080p projectors mounted in class rooms would be helpful.

This year an experiment was performed with wireless keyboards/mouse pads in class to great effect. The keyboard/mouse pad floats around the class room so that students are in charge of/responsible for writing code that appears on the classroom screen projected from the faculty laptop. Additional keyboards would allow other faculty and their students to benefit from this idea.

Department controlled clickers would allow for easy access.

Question: Final Analysis - Please provide any additional information deemed necessary to demonstrate program success.

CIS REPORT

Program Summary

Our CIS program teaches students a wide array of valuable tech skills employers seek that can be applied to a multitude of employment situations. Our CIS program could be much better. In addition to the above, it could also prepare students much better for careers as IT specialist or to pursue graduate CIS degrees. Unfortunately, this program is taught almost exclusively by adjuncts and although the CS faculty have tried to steer the program there are simply not enough faculty to do this while also maintaining a high quality CS program for our undergraduate students.

Criterion: 1. History & Development - Program History

Question: Program History - Describe your program and its history, including both program content and its students/constituents.

The CIS BS grew as an offshoot of the CS program and an updated version of our former Computing and Management BS which we believe was the first such major of its kind in the world. A CIS BS can be a ticket to a rewarding career in the tech world without pursuing a CS degree which many would agree is the more rigorous discipline. Unfortunately, CIS at Lehman College is largely rudderless. The department can rarely cover a CIS class with a regular fulltime faculty member. In the Spring 2015 semester one section out of 19 will be so covered. This point can be made more dramatic by noting that enrollment decreased since the requirement to take a minor eliminated which has led the department to cut the number of CIS sections offered very significantly in a very short time. Otherwise we might be looking at one section out of 35 taught by a fulltime faculty member. Many if not most of our CIS majors graduate without every taking a class with a fulltime member of the faculty. We have been fortunate that turn over in our CIS adjunct staff has been slow, which gives the students some continuity. This situation is still far from idea.

Question: Strategic Plan - Please provide the mission statement and summarize the strategic plan for your program or your department.

It is hoped that students majoring in CIS gain marketable tech skills. We were unable to do an informal survey of our CIS graduate like the ones we did for CS and Mathematics because faculty did not have contact information.

Question: Program Evolution - How have the needs of your constituents evolved over the past 3 years? How has the program adapted to these changes?

CS faculty have tried to improve the CIS BS. Through discussions with the adjuncts that teach the courses to learn about student needs. How fruitful these discussion have been is unclear. One item that seemed clear was that those students that chose CIS as a minor were generally far outperformed by those that chose it as a major. It was felt that in an effort to teach to the middle or to bring up the bottom that those majoring in the subject were losing out. The precipitous drop in minors now that the requirement for a minor has been removed, may be a blessing in disguise for this program and its students. We hope to raise the standards as we know that this program no longer measures up to competitors/peers

In recent years we have added courses on E-Commerce and changed the content of some of the math courses requires specifically for this major.

Question: Curriculum Map - Please submit your 4- year undergraduate curriculum degree map. (View Instructions)

Criterion: 2: External - External Demand for the Program

Question:

1/6/2015 9:06:31 PM

External Demand - Describe the external demand for the program from (a) potential students/constituents, and (b) for graduates of this program among employers and graduate schools?

Tech skills are in demand by employers. We hope that our offering have given our CIS graduates valuable skills that can be used in many jobs.

Question: Unmet Demand - How does the program (and at what level) meet external demand? Is there evidence of unmet demand? Please provide examples or identify the criteria used to reach your conclusion.

A much more rigorous CIS program would no doubt be a bigger draw for students. A comparison of our program to the CIS program at Baruch College informs us of the great strides we need to make.

Question: Competition - To what extent is this program competing with programs at other colleges? Which programs at CUNY and in the surrounding region would be considered competitors? Briefly describe the uniqueness of your program compared with those offered by competing institutions?

Baruch College is a model for CIS programs. Any student serious about a career in CIS would be well served to chose Baruch over our program. However, we do server a niche. Our students gain tech skills that can be used in any number of jobs.

Question: Program Recruitment - What domestic and international recruitment strategies and admissions criteria does your program use to attract students or other constituents into the program?

None.

Criterion: 3: Internal - Internal Demand for the Program

Question: Internal Demand - Explain the trends in the internal demand for your program using the data shown in the following tables: (View Instructions)

Institution Provided Data

Headcount Majors Undergraduate

DEGREE NAME	FALL 2011	SPRING 2012	FALL 2012	SPRING 2013	FALL 2013	SPRING 2014
Undergraduate BS	128	149	131	124	120	118

The trend is fairly stable.

Question: Services to Other Programs - Describe ways in which your program is: (a) relied upon by other programs on campus; (b) dependent on service courses or other resources offered by other programs? (View Instructions)

Students in just about any other major can benefit from the technical knowledge to be gained from our CIS courses. Whatever field a student enters it pays to be a skilled computer user.

Question: Adaptation to Constituents - What steps has the program taken in the past 3 years to reflect (a) the changing needs of its constituents, (b) technological advances, and (c) to develop innovative and forward-looking curricula?

Changes have been made to the content of the math courses specific to this major to cover material more relevant to th students needs.

Question:

Unmet Demand - How does the program (and at what level) meet internal demand? Is there evidence of unmet demand? Please provide examples or identify the criteria used to reach your conclusion.

Students would benefit from a more rigorous program. Rather than most CIS majors simply assimilating a series of tech skills which can be useful for jobs in general the major could make careers as CIS specialist more likely and prepare students for grad studies in CIS. With minors in CIS all but gone, this is an opportune time to make this improvement for those majoring in CIS.

Criterion: 4. Support and Staffing - Program Support and Staffing

Question: Program and Staffing Trends - Describe and/or explain the departmental staffing distribution for this program as presented in the data below: (View Instructions)

Institution Provided Data

Number of FT Faculty Tenure / Tenure Track (by Department)

	FALL 2011	FALL 2012	FALL 2013
TOTAL # OF TENURE / TEN. TRACK	25	24	22
DIST. & FULL PROFESSORS	17	16	14
ASSOCIATE PROFESSORS	2	2	3
ASSISTANT PROFESSORS	3	2	2
DIST. LECT. &LECT. CCE BRKDWN	3	4	3

Number of Adjunct and Substitute Professors

	FALL 2011	FALL 2012	FALL 2013
Number of Adjunct	52	33	35
Substitute	6	6	5

Number of Program Staff

	FALL 2011	FALL 2012	FALL 2013
HEO Series	2	2	3
NON-Teaching Adjuncts	0	0	3
College Lab Technitians	0	0	0

See CS BS.

Question: Staffing Contributions - What are the roles and contributions of part-time or adjunct faculty and professional staff to this program? How are they supported and integrated into the program?

See CS BS.

Question: Student Preparedness - Use the tables below to describe the academic preparedness of the constituents who enter the program. (View Instructions)

Institution Provided Data

ACADEMIC PLAN UNDERGRADUATE FRESHMAN

	FALL 2011	SPRING 2012	FALL 2012	SPRING 2013	FALL 2013	SPRING 2014
FULL TIME	20	23	18	14	14	9
PART TIME	8	10	3	5	1	4
TOTAL:	28	33	21	19	15	13

ACADEMIC PLAN UNDERGRDUATE TRANSFER

	FALL 2011	SPRING 2012	FALL 2012	SPRING 2013	FALL 2013	SPRING 2014
FULL - TIME	61	62	55	55	69	61
PART - TIME	39	52	54	48	34	42
TOTAL:	100	114	109	103	103	103

ACADEMIC PLAN UNDERGRADUATE UNKNOWN

	SPRING 2012	FALL 2012	SPRING 2013	FALL 2013	SPRING 2014
FULL - TIME				1	1
PART - TIME	2	1	2	1	1
TOTAL:	2	1	2	2	2

Unable to draw conclusion.

Question: Program Success - How does the program define student/constituent success? What support structures are in place to facilitate students/constituents?

We have a difficult time determining success. We do not know the career paths our graduates have pursued. We know that our program is not up to the standards of the Baruch CIS program, but we do believe that we impart a great deal of valuable skills to our students. We wish though to approach the level at which CIS is offered at Baruch.

Criterion: 5. Quality of Program - Quality of Program

Question: Program Assessment - Discuss how program assessment results have been utilized to improve student/constituent learning outcomes.

Assessment has been very difficult becauses course are almost exclusively taught by adjuncts and we do not have enough faculty to monitor the courses.

Question: Program Outcomes - If your program is subject to external accreditation, describe the results of your most recent accreditation review; If your program is not externally accredited, then please provide the results/recommendations and implementation plan from the most recent self-study.

The most recent self-study as well as external review dedicated very little attention to CIS but recognize that having a CIS program is positive for the students even while this program is somewhat rudderless due to the fact that it is taught almost exclusively by adjuncts.

Question: Teaching Effectiveness - What actions has your program taken to improve teaching effectiveness, student retention and student/constituent success, faculty scholarly and creative output, and alumni placement? What were the outcomes for each? Some CIS students have been able to be place into internships.

Question: Notable Achievements - Please identify and/or describe the most notable achievements of your students, past and present (for example: awards, fellowships, scholarships, publications, etc.)?

Unknown.

Question: Retention and Persistence - Please describe the strategies adopted by the program to promote student recruitment, retention, persistence, and graduation?

None.

Question: Public Service - Describe the contributions made by your faculty and professional staff to the surrounding community and to the profession.

See CS BS.

Criterion: 6.Size and Productivity - Program Size, Scope and Productivity

Question: Faculty Teaching Productivity - Explain the trends in faculty teaching productivity for your program as measured by department work flow summaries provided in the attached PDFs. (View Instructions)

Program is taught almost exclusively by adjuncts.

Question: Research and Scholarly Activities - Please describe how active and engaged are your faculty leaders and professional staff in the following areas: Number of refereed publications; Number of books, novels, etc.; Number of edited volumes; Number of chapter contributions to books [monographs]; Number of juried shows/performance; Number of editorial positions; Number of professional conference presentations; Number of non-refereed publications. (View Instructions) See CS BS.

Question: Student Success - Use the tables below to (a) describe the number of students/constituents who have completed your program and (b) the total number of credit hours taken to graduate. (View Instructions)

Institution Provided Data

Total_no_of_graduates

	2010-2011	2011-2012	2012-2013
Total number of graduates	30	33	41

Graduates_as_percent_of_total_student_headcount

	2010-2011	2011-2012	2012-2013
% graduated of total headcount	1.30	1.38	1.61

Graduation_headcount_for_undergraduates

	2010-2011	2011-2012	2012-2013
First Time Students	11	10	13
Transfers	19	23	28
Other	0	0	0

Credit hours taken to graduate - Undergraduate First Time Students

	2010-2011	2011-2012	2012-2013
Under 140 credit hours	11	10	12
140 credit hours or more	0	0	1

Credit hours taken to graduate - Undergraduate Transfer Students

	2010-2011	2011-2012	2012-2013
Under 140 credit hours	18	19	27
140 credit hourrs and above	1	4	1

104 students have graduated with a CIS BS in the last three years.

97 complete in less than 140 credits and 7 required more. Unlike our CS BS students in the CIS BS don't often do minors or double majors.

Criterion: 7a. Revenue - Revenue and Other Resources Generated by the Program

Question: Enrollment Revenue - Use the tables provided to describe the revenue generated through enrollment in your program's courses. (View Instructions)

Institution Provided Data

ENROLLMENT REVENUE BY DEPARTMENT (in $ amount)

FALL SPRING FALL SPRING FALL SPRING

	2011	2012	2012	2013	2013	2014
GRADUATE	187933	112907	121874	162660	162855	108185
UNDERGRADUAT	2145814 1929377	2084942 2065130	1968546 2144218 E			
TOTAL	2333747	2197849	2090420	2092037	2227985	2252403

ENROLLMENT REVENUE BY DEPARTMENT FOR SUMMER & WINTER (in $ amount)

	SUMMER 2011	WINTER 2012	SUMMER 2012	WINTER 2013	SUMMER 2013	WINTER 2014
AMOUNT:	491642	12825	430481	22560	435169	11250

Other Revenue (by Department)

	2012	2013	2014
OSRP - Research programs	403711	985504	347393
Lehman Foundation/Fundraising			
Academic Excellence Fees			
Tuition differential (SWK)			
Laboratory & Material Fees			

See CS BS.

Question: Program Revenue - Please describe the activities or strategies used to increase program revenue from enrollment?

See CS BS.

Question:

Other Revenue Generated - Please identify and describe the extent to which the program is supported through internal (Lehman or CUNY) and external (local, state, federal, or foundation) funds and grants for research, programs, equipment, and other sources (such as fundraising, excellence fees, laboratory fees, ticket revenue, materials fees, social work tuition fees, etc.)? If your other revenue is not listed below then use the table in question 7a.1. (View Instructions) See CS BS.

Question: Instituional Advancement - Please list and/or describe any development or advancement funds (or other gifts) received by or for the program.

Please provide, if available, the development/advancement strategic plan or case statement(s) for the program. (View Instructions) See CS BS.

Criterion: 7b. Expenditures - Expenditures Associated with the Program

Question: Cost Trends - The three-year trend for costs associated with the program is provided below. Please provide your analysis of this trend and likely future trends. (View Instructions)

Institution Provided Data

Program Cost on PS Reguar, PS Adjunct PS Temp Services (by Department)

	2012	2013	2014
PS REGULAR	2842455	2787894	2720070
PS ADJUNCT	996191	693571	738519
PS TEMPORARY	63604	86353	79126
TOTAL PERSONAL SERVICES:	3902250	3567818	3537715

Program Cost on OTPS (by Department)

	2012	2013	2014
NPS SUPPLIES AND MATER.(80120)	2049	1841	2679
NPS MISC CONTRACT. SERV(80122)			
NPS TRAVEL (80121)			
NPS EQUIPMENT ACQUIP(80123)	193		
FRINGE BENEFITS(80124)			

Program Cost Total on OTPS (by Department) (Sum of OTPS Above)

	2012	2013	2014
TOTAL OTPS	2242	1841	2679

Other

	2012	2013	2014
Academic Excellence Fees			
Tuition differential (SWK)			
Laboratory & Material Fees			

See CS BS.

Question:

Program Efficiencies - Please elaborate on the current level of departmental budgeting and expenditures relative to current demands. What mechanisms are in place to request and what policies are used to justify additional funding?
See CS BS.

Question: Professional Development - Please identify and describe the contributions and sources for professional development activities (i.e., teaching expenses, research expenses, conference travel, reassigned time) provided to the program. Describe the level of participation by the program in developing this budget? What proportion of professional development costs are covered by the program? What portion is contributed to by faculty? See CS BS.

Question: External Mandates - Please identify or describe any/all external mandates that may affect the cost of the program in the next two years?
See CS BS.

Criterion: 8: Opportunity Analysis - Opportunity Analysis and Impact of the Program

Question: Program Aspirations - Please identify your aspirational peer programs, and provide rationale for their selection? Please identify and describe the metrics (measureables) required to bring this program up to that

aspirational level, over what length of time, and what departmental and institutional cost/ investment.
Baruch College is the model we would like to emulate.

Only a large number of quality hires would make this possible.

Question: Internal and External Opportunities - What internal and external opportunities can be leveraged to strengthen this program or help it develop in new directions?
Hiring faculty.

Question: Technology - Please identify or describe any technological innovations that might be used to strengthen this program. See CS BS.

Question: Final Analysis - Please provide any additional information deemed necessary to demonstrate program success.

Appendix G: Justification for Faculty Replacement Fall 2022 (Edited)

Justification for Replacement and/or New Position Lines Template[6]

Purpose: This template is a succinct mechanism for each Dean, Associate Provost, and Vice President to use when providing a justification for requesting replacement and/or new position line(s). Given the College's budget constraints, it is necessary to consider support of potential high growth, high impact, and high retention areas in making the request. Each request will be judged on its merits and must adhere to the guidelines below. Type inside the box underneath the justification.

Directions: Your justification document **is to be no longer than five (5) pages**. Should you need to provide additional document(s), i.e., program reviews or accreditation documents, it is preferrable that you provide a link to those document(s). If you are not able to provide a link, please provide executive summaries of the document(s) as an appendix. If there is more than one appendix, label each appendix in numeric order as introduced in the body of the document. Make sure the appendix is labeled in the same order as in this table. **Please limit this entire document to no more than 15 pages with appendices.**

Justification
It is expected that a ***minimum of three programmatic justifications*** from this list be used to support your request for a replacement and/or new position line(s). **Limit your discussion to 300 to 600 words for each of the justifications you use. Bullet points are ideal.** You may use three or more justifications that you deem suitable for your request with evidence to support it.
(1) Program's centrality to College mission and strategic plan https://www.lehman.edu/strategic-planning/ and how the position will support it (e.g., adaptation to regional needs, changing student demography, how position supports the college's efforts to create a more diverse and inclusive workplace and learning environment).
Goal 1: Objective 1.2 of Lehman's strategic plan calls for Lehman to "provide a transformative experience that emphasizes experiential learning and high impact practices for all undergraduate and graduate students, prepares graduates to meet the challenges of the 21st century, enhance digital equity..."

[6] Office of the Provost, October 3, 2022

In recent years the CS department has supported and been supported by initiatives aimed at this goal. For instance:

The New York City Council and New York City Economic Development Corporation supported creation of the Bronx Business Tech Incubator at Lehman. This for tech freelancers, local small businesses, and entrepreneurs to create jobs and promote innovation and technology development in the Bronx. Through this program Lehman College Computer Science majors received training in web development from current CS department faculty and were connected to local small businesses in need of such services. A win-win for the businesses and the students.

The mayor's office, through its Tech Talent Pipeline and CUNY 2x Tech initiatives, allowed the CS department to bring a number of benefits to our students that line up with Lehman's mission. These programs helped bring additional industry practitioner into our classrooms to share their knowledge, experience, and networks with our students. Through these programs many dozens of valuable internships were secured by our students. Further, the Tech Talent Pipeline awarded Lehman College funding to support the hiring of two lecturers who the FP&B and President have recommended for CCEs this year. These two lectures have played an instrumental role in the success of the department in meeting its mission as it lines up Objective 1.2 of Goal 1 of Lehman's strategic plan.

The new CUNY Inclusive Economy program will allow the CS department to continue the type of efforts that were so valuable in the CUNY 2X Tech program to CS majors while Tech Talent Pipeline has moved to CUNY Central and continues the role it has played in collaboration with the CS department.

Through student organizations NSBE, SHPE, and WiCS and faculty run organization GDC Bronx the collaboration of CS faculty has helped many students secure internships and regular employment with companies such as Google, Microsoft, IBM, Facebook, General Motors, Twitter, Goldman Sachs, JP Morgan Chase, Electronic Arts, Lockheed Martin, and many more.

(2) External demand for program (e.g., use local, regional, and national data to demonstrate program need and attractiveness).

BLS data puts the current mean annual wage for software developers in the US at $120,990 with the median at $120,730. The lowest 25% earn $90,870 or less and the highest 25% earn $151,960 or more. The lowest 10% earn $64,470 or less and the highest 10% earn $168,570 or more.
https://www.bls.gov/oes/current/oes151252.htm

BLS projects a 25% growth rate for software developer jobs in 2021-2031 (in comparison with a 5% rate in general), which means 411K more positions in the next ten years (typically with BS as an entry-level education). Source: https://www.bls.gov/ooh/computer-and-information-technology/software-developers.htm#tab-1

Lehman College is an engine of upward mobility recognized with a #3 ranking on *The Chronicle of Higher Education* list of institutions with the highest mobility rates in the nation. Its Department of Computer Science is a big part of this and has been ranked #3 by University HQ in 2021 and 2022 in this area. https://universityhq.org/best-colleges/rankings/most-affordable-computer-science-schools/#rankings

The Provost has shared that Lehman CS students averaged $87,500 starting salaries in 2019.

(3) Internal demand for program (e.g., current enrollment, projected enrollment, enrollment in service/general education courses required by other programs—use internal data to demonstrate current demand impact and potential for internal demand).

Internal demand for both the CIS and the CS majors has grown from Fall 2018 to Fall 2021, even as enrollments at Lehman have dropped. In Fall 2018, there were 4088 stu-cr-hrs taught and in Fall 2021 there were 4903 stu-cr-hrs taught (Provost data), an increase of 20% in three years! The number of majors in CIS dropped in that 4-year period (from 245 in Fall 2018 to 220 in Fall 2021) but the number of majors in CS has increased from 335 to 406. CUNYFirst queries indicate an increase to 463 CS majors and 240 CIS majors for Fall 2022. There is no reason to expect any dramatic reductions over the next few years given that enrollments have climbed in CS while general enrollments have fallen across the college. Given growing demand in industry for computer scientists, data scientists, and those with skills in cybersecurity there is confidence that in fact the CS department will see continued growth. (see response to #2) The student demand for CS remains robust. A soon to be revised CS BS with an additional 20 CS credits required will add to demand for CS courses. Further, CS and CIS courses are currently not included in Pathways. There are 4 to 6 courses that should be in Pathways to benefit CS and CIS majors as well as to expose more students to the increasingly important and far reaching world of computer science.

(4) Quality of program inputs and processes and how position will improve them (e.g., proportion of full-time faculty with terminal degrees and staff appropriate for the discipline, especially in the case of accredited programs, quality of students vis-à-vis admissions criteria, curriculum coherence to meet changing needs of students, and program's adaptation to changes in technology and online learning). Accredited programs should indicate the accreditation standards, compliance for faculty qualifications, or self-study in preparation of site visit.

The CS department has nine dedicated fulltime faculty, six professors with Ph.D.s and three lecturers with masters degrees. With the hiring of an additional lecturer or two the department will be even more "lecturer heavy." The lecturers are needed to cover courses but so are more professorial faculty to both teach advanced electives and to do research. The department is informed that prospective professorial candidates consider the ratio of lecturers to professors when deciding where to apply and that the ratio has already given people pause in the past.

(5) Quality of program outcomes and how the position will improve them (e.g., student learning outcomes in courses with high enrollment and high DWIF rates, and student performance on external certifications; faculty productivity, program efficacy, and program profile/visibility). Accredited programs should provide evidence from self-study report used for the site visit when available. Non-accredited programs should provide evidence from a program review document that outlines program outcomes, evidence of assessment of program at course and program level.
(6) Program's size, scope, productivity, and growth potential with the position (e.g., number of students served, credit hours generated, degrees and certificates awarded annually, number of majors and minors, number and adequacy of faculty and staff, services rendered, research developed).
As indicated in #3, enrollments in CIS and CS have remained stable or have grown. The department has had to make difficult decisions regarding instruction due to the limited size of the full time faculty. Currently there are nine (9) full-time faculty with Travia leave in Fall 2023 followed by retirement for one at the end of 2023. One (1) faculty member is a distinguished professor carrying a large workload balance who fulfills his teaching workload through his work at the Graduate Center. Thus eight (8) full-time faculty, composed of two (2) tenured associate professors, three (3) untenured assistant professors, and three (3) lecturers teach both the large introductory courses as well as the highly specialized advanced courses in the major. In the fall there will be an additional lecturer and thus there will remain 8 full-time faculty responsible for 3600 student-credit-hours and what current CUNYFirst queries indicate are over 800 majors as of Spring 2023. The high number of majors does not tell the complete story of the student experience in Computer Science. Over 300 students enroll each semester in an introductory course hoping to learn coding and advance in the majors. These and other students taking early courses of the major have generally not yet declared their major. The vast majority of these students have no coding experience and are poorly positioned to advance without considerable resources and attention. Thus, the first year experience requires creativity and time from the full-time faculty reducing the faculty available to serve the majors. Replacing the two professoriate lines (as well as adding the new Lecturer line) will bring the department to 11 full-time faculty serving 800 majors. A minimally acceptable student:faculty ratio?
(7) Revenue and other resources generated by program as well as potential for growth.
Continued pressure and demands are levied on the computer science faculty by external factors. The department has focused some attention on computer science education supporting teacher education and it continues to focus heavily on preparing students for the workforce and key technology areas. The department faculty's desire to create a data science program as well as a cybersecurity program have been on hold due to the inappropriately low number of faculty who are already overwhelmed with initiatives. The department could also focus on reviving their graduate program. The demand is high and the potential for increased revenue is also substantial. For example, a fully-online teacher education program that is effective and supports national calls for "computing for all" could be a revenue generator. Workforce demands for software engineers with various levels of skills and

expertise continue to grow and funds are available to support the development of students to enter the technology workforce. At this point, the demands on the department exceed the ability of the department to meet those demands. Additional revenue from grant funding and city funding seem likely as the faculty size increases and faculty have time to pursue the revenue. To better prepare students for employment in the software development industry the department is awaiting approval of a major overhaul of the Computer Science and Computer Information Systems majors. The Computer Science major will increase the credit requirement by 16-20 credits. This will necessitate the offering of additional elective courses. Allowing for this change was a key reason for splitting Computer Science from Mathematics in 2017 so that mathematics requirements of the major would not count against the credit cap. Unfortunately, the Department of Computer Science has never had enough faculty to implement the enhancement to the major. It still does not, but students need this competitive edge. Changes to the CIS major include adding a much-requested programming track that will see students who shy away from the mathematics heavy CS major gain an option to prepare them for low level programming jobs such as web development that still pays very well and can lead a student to choose to pursue future studies in CS. There are no CS or CIS courses in Pathways. There are perhaps 6 but at least 4 courses that should be in the Pathways and are at other CUNYs where the idea that an understanding of computer science at least at a rudimentary level is a requirement of a well-rounded education in the 21st century. Adding these courses to Pathways would lead to more students discovering that they can excel in CS. Over the last decade as the undergraduate program grew and faculty numbers shrunk, the graduate program deteriorated as subjects that once had a significantly more advanced graduate version that assumed an undergraduate prerequisite in the subject were replaced with graduate/undergraduate crosslisted sections. Rejuvenating the graduate program will attract students who currently would not consider the program. Additional faculty are needed for all of these reasons and more that would allow the department to grow and bring in more revenue.
(8) Number and percentage of full-time to part-time faculty ratio by College, Schools, Library, and Programs (2018 – present).
In Fall 2018, there were 10 full-time faculty and 8.4 part-time equivalent <u>lecturers</u> (18 hour teaching load) and 53% of the contact hours were taught by full-time faculty. Note however, that fulltime faculty have taught significantly beyond their designated workloads both adding to courses taught by fulltime faculty and subtracting from courses taught by part-time faculty.
(9) Percentage of annual instructional full-time equivalencies (FTEs) in undergraduate courses delivered by full-time faculty by College, Schools, Library, and Programs (2018 – present).
(10) Contribution to expansion of knowledge through original research and scholarship, and/or creative work.
The CS department is at a critical point concerning its research program. Each of the professors has produced quality research. However, there are currently only five active researchers of the six

professors. The chair is overwhelmed with admistrivia in large part due to the low number of faculty versus the large number of students and initiatives. One of the faculty members is a world-renowned distinguished professor and another is a young associate professor with an outstanding research record who have both been attractors of new talent. Without doubt, three of the four other professor are only at Lehman because of one or both of these faculty. The distinguished professor is in his mid-eighties and still publishing quality work at a remarkable pace. The department needs to bring in new talented researches while the attraction remains. Five or six researchers does not give the department the critical mass to remain an attractive site for researchers.
(11) Any other pertinent information to support request.
1. Growth in Master's degree (requires professoriate level faculty) 2. Additional upper level course requirements for Bachelors (requires professoriate level faculty) 3. CS and CIS courses added to Pathways (requires more faculty) 4. Support of introductory students 5. Enhance programming skills of aspiring CS majors, transfer students, and general population 6. Support of experiential learning/workforce through advising and workforce preparation With the current situation, i.e. 800 majors in the two majors, approximately 12 professors and 4 lecturers are desperately needed. Expected growth will necessitate further faculty additions.

Appendix H: New CS and CIS BS Degree Requirements and Courses

LEHMAN COLLEGE
OF THE
CITY UNIVERSITY OF NEW YORK

DEPARTMENT OF Computer Science

CURRICULUM CHANGE

Name of Program and Degree Award: Computer Information Systems (CIS)
Hegis Number: 0702
Program Code: 83120 / MHC 60201
Effective Term: Fall 2023

1. **Type of Change:** Change in degree requirements.

2. **From:** ~~Strikethrough~~ the changes

~~Computer Information Systems, B.S. (55-57 Credit Major)~~

~~Required Courses (19 credits):~~

~~CIS 166~~	~~Computer Programming for Information Processing I~~	~~3~~
~~CIS 211~~	~~Computer Information Systems~~	~~4~~
CIS 212	Microcomputer Architecture	~~3~~
CIS 244	Introduction to Database Management	3
CIS 331	Network Introduction	3
CIS 344	Database Design and Programming	3

~~Elective Courses (6-8 credits):~~
~~Two additional courses chosen from the 200-level (or higher) CIS courses or from:~~

~~CGI 221~~	~~Applied Imaging and Applications to the World Wide Web I~~	~~3~~
~~CGI 321~~	~~Computer Modeling and Design I~~	~~3~~
~~CGI 421~~	~~Computer Animation I~~	~~3~~
~~CMP 168~~	~~Programming Methods II~~	~~4~~

~~One of the courses must be a 300- (or 400-) level CIS course.~~

In Mathematics: Required Courses (15 credits):

MAT 132	Introduction to Statistics	4
MAT 174	Elements of Calculus	4
MAT 301	Applied Statistics and Computer Analysis for Social Scientists	3
MAT 348	Mathematical Methods for Management	4

~~MAT 174:~~ Students considering graduate work should take MAT 175, MAT 176 instead of MAT 174.

~~**In Economics: Required Courses (9 credits):**~~

~~ECO 166~~	~~Introduction to Macroeconomics~~	~~3~~
~~ECO 167~~	~~Introduction to Microeconomics~~	~~3~~
~~ACC 171~~	~~Introduction to Accounting for Non-Accounting Majors~~	~~3~~
	~~or~~	
~~ACC 185~~	~~Principles of Accounting~~	~~3~~

~~**Further Electives (6 credits):**~~
~~**Students must choose two courses from the following:**~~
~~*One 200-level CIS course: Or higher, for 3 credits.*~~
~~AND~~
~~*At least one of PHI 221 and POL 299 must be chosen.*~~

~~PHI 221~~	~~Ethical Issues in Computing and Technology~~	~~3~~
~~POL 299~~	~~Law, Computers, and the Internet: The Politics of Information Technology~~	~~3~~

3. **To:** Underline the changes
Computer Information Systems, B.S. (60 - 68 Credit Major)

Courses Required For All CIS Majors:

CMP 128 - Programming Through Web Development	3
CMP 157 - Programming Methods I Lab	1
CMP 167 - Programming Methods I	4
CIS 212 - Microcomputer Architecture	4
CIS 213 - Microcomputer Architecture Lab	1
CIS 234 - Introduction to Spreadsheet Analysis	3
CIS 244 - Introduction to Database Management	3
CIS 247 – Practical UNIX:Programming & System Administration	4
CIS 331 - Network Introduction	3
CIS 344 - Database Design and Programming	3
CIS 345 - Introduction to Operating Systems	4
PHI 221 or POL 299	3
MAT 132 - Introduction to Statistics	4
MAT 174 - Elements of Calculus	4
MAT 301 - Applied Statistics and Computer Analysis for Social Scientists	3
MAT 348 - Mathematical Methods for Management	4

Students considering graduate work should take MAT 175, MAT 176 instead of MAT 174.

At least 1 Elective Track is Required for All CIS Majors: REQUIRED Track 9-17 credits

Elective Tracks for CIS Majors:

Web Development Track

CMP 158 - Programming Methods II Lab	1
CMP 168 - Programming Methods II	4
CMP 343 - Web Development	4
CIS 346 - E-Commerce	4
	Total 13 credits

Programming Track

CMP 158 - Programming Methods II Lab	1
CMP 168 - Programming Methods II	4
CMP 232 - Discrete Mathematics	4
CMP 269 - Programming Methods III	4
CMP 338 - Data Structures	4
	Total 17 credits

Hardware Track

CIS 341 - Computer System Fundamentals (can be substituted by A+ Certificate)	3
CMP 232 - Discrete Mathematics	4
CMP 334 - Computer Organization	4
	Total 11 credits

Business Track

ECO 166 Microeconomics	3
ECO 167 Macroeconomics	3
ACC 171 or 185 Intro to Accounting	3
	Total 9 credits

CGI Track

ART 112 Introduction to Digital Imaging	3
CGI 221 Applied Imaging and Applications to the World Wide Web I	3
CGI 321 Computer Modeling and Design I	3
CGI 421 Computer Animation I	3
	Total 12 credits

4. **Rationale (Explain how this change will impact learning outcomes of the department and Major/Program):**
In the rapidly changing field of Computer Information Systems, the current major is no longer serving our students' needs.

The new major is now divided into core requirements that all CIS majors must complete. These requirements provide our students with the foundation they will need to complete their undergraduate studies and will prepare them for possible graduate studies going forward.

In addition, the major now includes multiple specialization tracks which students can choose from. Students need only complete one track in order to complete the major. However, they may choose to add additional track(s) if they wish.

5. **Date of departmental approval:** March 01, 2022

LEHMAN COLLEGE
OF THE
CITY UNIVERSITY OF NEW YORK

DEPARTMENT OF Computer Science

CURRICULUM CHANGE

Name of Program and Degree Award: Computer Science, B.S.
Hegis Number: 0701
Program Code: 60202 / MHC 60203
Effective Term: Fall 2023

1. **Type of Change:** Change in degree requirements

2. **From:** ~~Strikethrough~~ the changes
~~Computer Science, B.S. (57-61 Credit Major)~~
Credits

MAT 175	Calculus I	4
MAT 176	Calculus II	4
MAT 313	Elements of Linear Algebra	4
CMP 167	Programming Methods I	~~3~~
CMP 168	Programming Methods II	4
~~CMP 232~~	~~Elementary Discrete Structures & Applications to Computer Science~~	~~4~~
CMP 334	Computer Organization	4
~~CMP 338~~	~~Data Structures and Algorithms I~~	~~4~~
~~CMP 405~~	~~Introduction to Networks~~	~~3~~
~~CMP 420~~	~~Database Systems~~	~~4~~
~~CMP 426~~	~~Operating Systems~~	~~4~~

Electives:

Four Advanced CMP courses (300 or 400 Level)
12-16
MAT 226 or PHY-305 can be substituted for one of these courses
~~One Advanced MAT course (300 or 400 Level) not including~~
~~3-4~~
~~MAT 300, MAT 301, MAT 348, CMP 332 or CMP 416~~

3. **To:** Underline the changes
Computer Science, B.S. (78-80 Credit Major)

MAT 175 Calculus I
4
MAT 176 Calculus II
4
MAT 226 Vector Calculus
4
MAT 313 Elements of Linear Algebra
4

Total Required MAT
16

CMP 157	Programming Methods I Lab	1
CMP 158	Programming Methods II Lab	1
CMP 167	Programming Methods I	4
CMP 168	Programming Methods II	4
CMP 232	Discrete Mathematics	4
CMP 269	Programming Methods III	4
CMP 334	Computer Organization	4
CMP 338	Data Structures	4
CMP 340	Introduction to Probability for Computer Science	4
CMP 410	Design and Analysis of Algorithms	4

Total Required CMP 34

Electives: Four Advanced CMP courses (300 or 400 Level) 16

PHY-305 can be substituted for one of these courses

Tracks: Three Advanced CMP courses in a single track (see below) 12

Tracks should be selected in consultation with the department advisor.

Systems Track

CMP 426	Operating Systems	4
CMP 405	Introduction to Networks	4
CMP 420	Database Systems	4

Total For Systems Track 12

Web Development Track

CMP 405	Introduction to Networks	4
CMP 342	Internet Programming	4
CMP 343	Full Stack Web Development	4
Total For Web Development Track		12

Software Engineering Track

CMP 346	Object Oriented Techniques	4
CMP 428	Video Game Programming	4
CMP 408	Software Engineering	4
Total For Software Engineering Track		12

Mobile Programming Track

CMP 346	Object Oriented Techniques	4
CMP 431	Mobile Programming for Android	4
CMP 432	Mobile Programming for iOS	4
Total For Mobile Programming Track		12

Artificial Intelligence Track

CMP 414	Artificial Intelligence	4
CMP 333	Data Management and Analysis	4
CMP 415	Machine Learning	4
Total For Artificial Intelligence Track		12

4. **<u>Rationale (Explain how this change will impact learning outcomes of the department and Major/Program)</u>:**
In the rapidly changing field of Computer Science, the current major is no longer serving our students' needs.

The new major is now divided into core requirements that all CMP majors must complete. These requirements provide our students with the foundation they will need to complete their undergraduate studies and will prepare them for possible graduate studies going forward.

In addition, the major now includes multiple specialization tracks which students can choose from. Students need only complete one track in order to complete the major. However, they may choose to add additional track(s) if they wish.

5. **<u>Date of departmental approval</u>:** March 01, 2022

LEHMAN COLLEGE
OF THE
CITY UNIVERSITY OF NEW YORK

DEPARTMENT OF Computer Science

CURRICULUM CHANGE

1. **Type of change:** New course

2.

Department(s)	Computer Science
Career	[X] Undergraduate [ ] Graduate
Academic Level	[X] Regular [ ] Compensatory [ ] Developmental [ ] Remedial
Subject Area	Computer Information Systems
Course Prefix & Number	CIS 213
Course Title	Microcomputer Architecture Lab
Description	Practical application of concepts learned in CIS212 - Architecture of microcomputer systems.
Pre/ Co Requisites	Corequisite: CIS 212.
Credits	1
Hours	1
Liberal Arts	[ ] Yes [X] No
Course Attribute (e.g. Writing Intensive, WAC, etc)	
General Education Component	X Not Applicable ____ Required ____ English Composition ____ Mathematics ____ Science ____ Flexible ____ World Cultures ____ US Experience in its Diversity ____ Creative Expression

		____ Individual and Society ____ Scientific World

3. **Rationale:**
This course will serve as a one hour lab per week, where practical application of Microcomputer Architecture concepts learned in the co-requisite course CIS 212 will occur.

4. **Learning Outcomes (By the end of the course students will be expected to):**
By the end of the course students will be expected to demonstrate their understanding of concepts learned in CIS212 - Architecture of microcomputer systems, by applying their learned knowledge on a variety of physical and virtual devices.

1. Define systems architecture and related terms
2. Define and identify the components and functions of computer networks
3. Describe numbering systems and their use in data representation
4. Describe CPU instruction and execution cycles
5. Describe the distinguishing characteristics of primary and secondary storage
6. Demonstrate how the CPU and bus interact with peripheral devices
7. Describe basic concepts of text and image representation and display
8. Explain communication protocols
9. Describe logical and physical network topologies
10. Describe the application development process and the role of methodologies, models, and tools

5. **Date of Departmental Approval:** March 24, 2021

LEHMAN COLLEGE
OF THE
CITY UNIVERSITY OF NEW YORK

DEPARTMENT OF_COMPUTER SCIENCE

CURRICULUM CHANGE

1. **Type of change:** *New Course*

2.

Department(s)	Computer Science
Career	[X] Undergraduate [ ] Graduate
Academic Level	[X] Regular [ ] Compensatory [ ] Developmental [ ] Remedial
Subject Area	Computer Science
Course Prefix & Number	CMP 157
Course Title	Programming Methods I Lab
Description	Application of concepts learned in CMP 167 to develop programming solutions to problems as lab assignments. Code will be written in an Integrated Developer Environment.
Pre/ Co Requisites	Prerequisite: MAT 104 or higher Corequisite: CMP 167
Credits	1
Hours	2
Liberal Arts	[X] Yes [ ] No
Course Attribute (e.g. Writing Intensive, WAC, etc)	
General Education Component	__X__ Not Applicable ____ Required ____ English Composition ____ Mathematics ____ Science

	____ Flexible ____ World Cultures ____ US Experience in its Diversity ____ Creative Expression ____ Individual and Society ____ Scientific World

3. **Rationale:**

This course will serve as the co-requisite for the first programming course CMP 167. Students will apply the concepts learned to develop their programming skills.

4. **Learning Outcomes (By the end of the course students will be expected to):**

1. Independently design, create, debug simple applications
2. Define and use variables of various data types
3. Define and use methods
4. Demonstrate the use of parameters and information passing in programs
5. Use existing libraries and their methods
6. Manipulate strings
7. Manipulate 1 dimensional arrays
8. Use control structures such as decision branching & iteration
9. Create classes to represent objects

5. **Date of Departmental Approval:** March 01, 2022

LEHMAN COLLEGE
OF THE
CITY UNIVERSITY OF NEW YORK

DEPARTMENT OF Computer Science

CURRICULUM CHANGE

1. **Type of change**: New Course.

2.

Department(s)	Computer Science
Career	[x] Undergraduate [] Graduate
Academic Level	[x] Regular [ ] Compensatory [ ] Developmental [ ] Remedial
Subject Area	Computer Science
Course Prefix & Number	CMP 269
Course Title	Programming Methods III
Description	An In-depth exploration of Object Oriented programming with emphasis on inheritance, interfaces, multi-threading, I/O, GUI, recursion and unit testing. Programming projects to be implemented in different languages.
Pre / Co Requisites	Prerequisites: CMP 158 and CMP 168
Credits	4
Hours	4
Liberal Arts	[X] Yes [ ] No
Course Attribute (e.g. Writing Intensive, WAC, etc)	None
General Education Component	_X___ Not Applicable _______ Required _______ English Composition _______ Mathematics _______ Science _______ Flexible _______ World Cultures _______ US Experience in its Diversity _______ Creative Expression _______ Individual and Society _______ Scientific World

3. **Rationale:**

The existing introductory programming sequence consists of two programming courses and covers the programming concepts required to develop basic functional applications in Java. The addition of this third course as part of the introductory programming sequence will offer more in-depth exploration of Object Oriented programming as well as exposure to different programming languages needed to succeed in subsequent courses. The addition of this third course as part of the introductory programming sequence will enable students to solidify their programming skills and broaden their knowledge of techniques and languages before moving on to more advanced courses.

4. **Learning Outcomes (By the end of the course students will be expected to):**

By the end of the course students should be able to read and write code in multiple programming languages and do the following:

1. Demonstrate OOP through proper use of encapsulation, polymorphism and multi-level inheritance hierarchies.
2. Implement multiple Interfaces as well as inheriting from classes that have implemented Interfaces
3. Implement generics and their use in object declarations
4. Demonstrate understanding and usage of Collections in each of the assigned programming languages
5. Demonstrate understanding and usage of Enumerations
6. Synchronize Threads using Semaphores in applications
7. Design and Develop fully functional applications in the assigned programming languages
8. Perform Serialization and Streaming of Objects
9. Use File I/O for both text and object storage/retrieval in applications
10. Development of GUI
11. Make use of Debugging Techniques and Tools
12. Make use of Testing Techniques and Tools (Unit testing and Integration testing)

5. Date of Departmental Approval: March 24, 2021

LEHMAN COLLEGE
OF THE
CITY UNIVERSITY OF NEW YORK

DEPARTMENT OF Computer Science

CURRICULUM CHANGE

1. **Type of change**: New Course.

2.

Department(s)	Computer Science
Career	[x] Undergraduate [x] Graduate
Academic Level	[x] Regular [ ] Compensatory [ ] Developmental [ ] Remedial
Subject Area	Computer Science
Course Prefix & Number	CMP 333
Course Title	Data Management and Analysis
Description	Introduction of Data handling tools and techniques, extracting and presenting information about data, and computational thinking processes.
Pre / Co Requisites	Prerequisites: CMP 158, CMP 168, CMP 232
Credits	4
Hours	4
Liberal Arts	[X] Yes [ ] No
Course Attribute (e.g. Writing Intensive, WAC, etc)	None
General Education Component	__X__ Not Applicable ______ Required ______ English Composition ______ Mathematics ______ Science ______ Flexible ______ World Cultures ______ US Experience in its Diversity ______ Creative Expression ______ Individual and Society ______ Scientific World

3. **Rationale:**

Data is the foundation of the digital age. Many industries generate massive datasets on a daily basis, and thus are in urgent need of professionals who can explore the dataset using data processing software and present relevant characteristics of the datasets in an explicit and understandable way. Mastering modern data analysis software and techniques is crucial for Computer Science students to meet the requirements of the fast-growing data science and analytics job market.

In a data science project, the data usually needs to be extracted from multiple files, databases, webpages, or PDFs. They can be in the form of numbers, measurements, words, labels, etc. Very often researchers need to convert the data into a cleaner and tidier form through several steps of the data wrangling process such as string processing, HTML parsing, working with dates and times, and imputing missing values. It is a crucial skill to communicate information about datasets clearly and efficiently through data description and visualization tools, such as statistical graphs, charts, plots, and information graphics. Effective description of data makes complex data more accessible and understandable, and helps researchers reason about data more easily. Adding this course to the Computer Science curriculum would help our graduates remain competitive in highly sought-after data science positions.

4. **Learning Outcomes (By the end of the course students will be expected to):**

1. Import data into Python from files in different formats.
2. Scrape data from websites and databases.
3. Process text, images, and date and times.
4. Create data aggregation using pivot tables.
5. Apply formulas to extract statistical information.
6. Visualize the dataset and make inferences.

5. Date of Departmental Approval: March 24, 2021

LEHMAN COLLEGE
OF THE
CITY UNIVERSITY OF NEW YORK

DEPARTMENT OF Computer Science

CURRICULUM CHANGE

1. **Type of change**: New Course.

2.

Department(s)	Computer Science
Career	[x] Undergraduate [ ] Graduate
Academic Level	[x] Regular [ ] Compensatory [ ] Developmental [ ] Remedial
Subject Area	Computer Science
Course Prefix & Number	CMP 340
Course Title	Introduction to Probability for Computer Science
Description	Introductory probability theory with applications to computer science. Axioms of probability, conditional probability, discrete and continuous random variables, expectation, variance and covariance, law of large numbers, central limit theorem.
Pre / Co Requisites	Prerequisites: MAT 226 and CMP 338
Credits	4
Hours	4
Liberal Arts	[X] Yes [ ] No
Course Attribute (e.g. Writing Intensive, WAC, etc)	None
General Education Component	__X___ Not Applicable _______ Required _______ English Composition _______ Mathematics _______ Science _______ Flexible _______ World Cultures _______ US Experience in its Diversity _______ Creative Expression _______ Individual and Society

	______ Scientific World

3. **Rationale:**

Probability theory is of fundamental importance in computer science. Its role in computer science is rapidly growing in recent years, partially due to the rise in interest in machine learning and, more generally, in data science. Knowledge of basic probability theory is important for students who would like to develop careers in machine learning/data science. It also serves them well in almost any other area of computer science, including data structures and algorithms, optimization, cryptography, simulation, and more. It is therefore our belief that a working knowledge of basic probability theory is necessary for every computer science student. The course will have many similarities to MAT 330, but will have more examples of applications of probability theory to computer science. We should also point out that the mathematics department has indicated that it is unable to offer MAT 330 to all computer science major, and so it becomes necessary for us to offer a local CS variant of the course.

4. **Learning Outcomes (By the end of the course students will be expected to):**

1. Understand and apply basic probability theory, including random variables, distributions, and expectation.
2. Understand and apply the law of large numbers and central limit theorem.
3. Be prepared for learning advanced topics in Computer Science that rely on probability theory.

5. Date of Departmental Approval: March 24, 2021

LEHMAN COLLEGE
OF THE
CITY UNIVERSITY OF NEW YORK

DEPARTMENT OF Computer Science

CURRICULUM CHANGE

1. **Type of change:** New Course

2.

Department(s)	Computer Science
Career	[X] Undergraduate [ ] Graduate
Academic Level	[X] Regular [ ] Compensatory [ ] Developmental [ ] Remedial
Subject Area	Computer Science
Course Prefix & Number	CMP 343
Course Title	Full Stack Web Development
Description	Full stack web development through experiential project-based learning using apis, protocols, and popular frameworks.
Pre/ Co Requisites	Prerequisites: CMP 158, CMP 168
Credits	4 Credits
Hours	4 Hours
Liberal Arts	[ ] Yes [X] No
Course Attribute (e.g. Writing Intensive, WAC, etc)	
General Education Component	__X__ Not Applicable ____ Required ____ English Composition ____ Mathematics ____ Science ____ Flexible ____ World Cultures ____ US Experience in its Diversity ____ Creative Expression

		____ Individual and Society ____ Scientific World

3. **Rationale:**
In a world increasingly connected by the Internet, the World Wide Web has emerged as a nearly ubiquitous platform for personal communication, business operations, and global information sharing.
Students of computer science will benefit from understanding the web and creating applications.

4. **Learning Outcomes (By the end of the course students will be expected to):**
   1. Explain the concepts, implement the syntax, and know the benefits of using JavaScript frameworks on the client-side as well as server-side.
   2. Develop aesthetically pleasing web applications that include back-end constructs such as databases and application servers.
   3. Describe the benefits of:
      a. various architectures
      b. design patterns
      c. elements of production-readiness
      d. historical revisions and version adoption by browsers

5. **Date of Departmental Approval:** March 24, 2021

LEHMAN COLLEGE
OF THE
CITY UNIVERSITY OF NEW YORK

DEPARTMENT OF Computer Science

CURRICULUM CHANGE

1. **Type of change**: New Course.

2.

Department(s)	Computer Science
Career	[x] Undergraduate [ ] Graduate
Academic Level	[x] Regular [ ] Compensatory [ ] Developmental [ ] Remedial
Subject Area	Computer Science
Course Prefix & Number	CMP 415
Course Title	Machine Learning
Description	Introduction to applied machine learning models and algorithms using a high level programming language and relevant scientific libraries.
Pre / Co Requisites	Prerequisites: CMP 158, CMP 168, MAT 313
Credits	4
Hours	4
Liberal Arts	[ ] Yes [X] No
Course Attribute (e.g. Writing Intensive, WAC, etc)	None
General Education Component	__X___ Not Applicable ______ Required ______ English Composition ______ Mathematics ______ Science ______ Flexible ______ World Cultures ______ US Experience in its Diversity ______ Creative Expression ______ Individual and Society ______ Scientific World

3. **Rationale:**
Machine learning is a method of data analysis that automates analytical model building. Such systems can learn from data, identify patterns, and make decisions with minimal human intervention. With the growing volumes and varieties of datasets in recent decades, there is a large job market for experts who can analyze the massive datasets and make data-driven decisions by utilizing machine learning models.

This course will prepare Computer Science students with the essential programming tools and machine learning techniques applicable to various data analysis tasks. Moreover, it will help students practice the application of quantitative analysis and interpretation skills to draw conclusions based on real-world information. Both skills are crucial for students who want to succeed in a data science career.

4. **Learning Outcomes (By the end of the course students will be expected to):**
1. Prepare data sets for training machine learning models.
2. Apply the concepts and procedures for popular machine learning algorithms.
3. Select proper methods to build and train machine learning models.
4. Tune a machine learning model to improve its performance.
5. Evaluate the performance of the machine learning model using the test dataset.

5. Date of Departmental Approval: March 24, 2021

LEHMAN COLLEGE
OF THE
CITY UNIVERSITY OF NEW YORK

DEPARTMENT OF Computer Science

CURRICULUM CHANGE

1. **Type of change**: New Course.

2.

Department(s)	Computer Science
Career	[x] Undergraduate [ ] Graduate
Academic Level	[x] Regular [ ] Compensatory [ ] Developmental [ ] Remedial
Subject Area	Computer Science
Course Prefix & Number	CMP 431
Course Title	Mobile Programming for Android Devices
Description	Introduction to Android Mobile Programming. Developing applications for devices running the Android Operating System using the Android Studio IDE to develop their Android Applications.
Pre / Co Requisites	Prerequisite: CMP 338
Credits	4
Hours	4
Liberal Arts	[ ] Yes [X] No
Course Attribute (e.g. Writing Intensive, WAC, etc)	None
General Education Component	__X___ Not Applicable _______ Required _______ English Composition _______ Mathematics ______ Science _______ Flexible _______ World Cultures _______ US Experience in its Diversity _______ Creative Expression

	______ Individual and Society ______ Scientific World

3. **Rationale:**

The pre-existing course code CMP 430 has been used for Mobile Development courses with students permitted to take CMP 430 up to 2 times to earn a total of 8 credits (4 for Android Mobile Development, and 4 for iOS Mobile Development). The UCC requested we keep CMP430 without offering it or requiring it for the major, but create 2 new course codes CMP 431 and CMP 432 that will be used to distinguish which Operating System the Mobile Development course focused on thus eliminating the need/ability to take the same course code of CMP430 for credit twice.

4. **Learning Outcomes (By the end of the course students will be expected to):**

1. Develop Android mobile applications.
2. Develop applications that properly handle the Android application lifecycle.
3. Create user interfaces, activities and handle events.
4. Pass data between activities.
5. Use themes, layouts and styles.
6. Work with menus, tabs, preferences, and settings.
7. Work with threads and files on Android mobile devices.
8. Use intents, services, notifications, alarms, and broadcast receivers.
9. Store data using SQLite database
10. (If time permits) Work with Content Providers

5. Date of Departmental Approval:

March 01, 2022

LEHMAN COLLEGE

OF THE
CITY UNIVERSITY OF NEW YORK

DEPARTMENT OF Computer Science

CURRICULUM CHANGE

1. **Type of change**: New Course.

2.

Department(s)	Computer Science
Career	[x] Undergraduate [ ] Graduate
Academic Level	[x] Regular [ ] Compensatory [ ] Developmental [ ] Remedial
Subject Area	Computer Science
Course Prefix & Number	CMP 432
Course Title	Mobile Programming for iOS
Description	Introduction to IOS programming using Apple's new language Swift. Developing applications for Apple's iPhones and iPads using the SwiftUI framework and Xcode
Pre / Co Requisites	Prerequisite: CMP 338
Credits	4
Hours	4
Liberal Arts	[ ] Yes [X] No
Course Attribute (e.g. Writing Intensive, WAC, etc)	None
General Education Component	__X___ Not Applicable ______ Required ______ English Composition ______ Mathematics _____ Science ______ Flexible ______ World Cultures ______ US Experience in its Diversity

	______ Creative Expression ______ Individual and Society ______ Scientific World

3. **Rationale:**
The pre-existing course code CMP 430 has been used for Mobile Development courses with students permitted to take CMP 430 up to 2 times to earn a total of 8 credits (4 for Android Mobile Development, and 4 for iOS Mobile Development). The UCC requested we keep CMP430 without offering it or requiring it for the major, but create 2 new course codes CMP 431 and CMP 432 that will be used to distinguish which Operating System the Mobile Development course focused on thus eliminating the need/ability to take the same course code of CMP430 for credit twice.

4. **Learning Outcomes (By the end of the course students will be expected to):**

At the end of the course students will be expected to

1. Demonstrate understanding of the essentials of iOS mobile application development.
2. Use Apple's Integrated Development Environment (IDE) XCode.
3. Develop applications that demonstrate proper use of the iOS application lifecycle.
4. Create user interfaces, activities and handle events using SwiftUI.
5. Work with Queues (threads) and files on iOS mobile devices.
6. Use the Foundation library of iOS tools.

5. Date of Departmental Approval:

March 01, 2022

LEHMAN COLLEGE
OF THE
CITY UNIVERSITY OF NEW YORK

DEPARTMENT OF Computer Science

CURRICULUM CHANGE

1. **Type of change**: New Course.

2.

Department(s)	Computer Science
Career	[x] Undergraduate [ ] Graduate
Academic Level	[x] Regular [ ] Compensatory [ ] Developmental [ ] Remedial
Subject Area	Computer Science
Course Prefix & Number	CMP 447
Course Title	Linear Programming and Operations Research
Description	Introduction to linear programming and other optimization techniques in Operations Research with applications of practical problems and theoretical computer science.
Pre / Co Requisites	Prerequisites: CMP 338, MAT 175, MAT 313
Credits	4
Hours	4
Liberal Arts	[ ] Yes [X] No
Course Attribute (e.g. Writing Intensive, WAC, etc)	None
General Education Component	__X___ Not Applicable _______ Required _______ English Composition _______ Mathematics ______ Science _______ Flexible _______ World Cultures _______ US Experience in its Diversity _______ Creative Expression _______ Individual and Society _______ Scientific World

3. **Rationale:**
Linear programming, as well as other optimization techniques in operations research, is the method to achieve the best outcome within limited resources and has a variety of applications in both practical and theoretical computer science. With the evolution of computer science in recent years, these techniques have been increasingly used across multiple areas of computer science, especially in Machine Learning and Artificial Intelligence. Therefore, we believe that it is important to provide our undergraduate students the basic knowledge of linear programming and operations research and propose this new course.

4. **Learning Outcomes (By the end of the course students will be expected to):**

1. Articulate and apply the basic methods of linear and nonlinear programming problems, including duality.
2. Formulate linear programming models.
3. Solve linear programming problems using simplex method.
4. Conduct sensitivity analysis of linear programming problems.

5. Date of Departmental Approval: March 24, 2021

LEHMAN COLLEGE
OF THE
CITY UNIVERSITY OF NEW YORK

DEPARTMENT OF Computer Science

CURRICULUM CHANGE

1. **Type of change**: New Course.

2.

Department(s)	Computer Science
Career	[x] Undergraduate [ ] Graduate
Academic Level	[x] Regular [ ] Compensatory [ ] Developmental [ ] Remedial
Subject Area	Computer Science
Course Prefix & Number	CMP 475
Course Title	Combinatorial & Graph Algorithms
Description	Introduction to graph and combinatorial algorithms with applications.
Pre / Co Requisites	Prerequisite: CMP 338
Credits	4
Hours	4
Liberal Arts	[x] Yes [] No
Course Attribute (e.g. Writing Intensive, WAC, etc)	None
General Education Component	__X___ Not Applicable ______ Required ______ English Composition ______ Mathematics ______ Science ______ Flexible ______ World Cultures ______ US Experience in its Diversity ______ Creative Expression ______ Individual and Society ______ Scientific World

3. **Rationale:**

Graphs arise in a variety of real-world situations, such as road map, supply network and social network, and hundreds of interesting computational problems are couched in terms of graphs. As such, algorithms for working with graphs are fundamental to the field of computer science. In view of its importance, and the evolution of computer science in recent years, where graph algorithms have seen more and more usage both in practical and in theoretical computer science, we believe that it is important to provide our undergraduate students the basic knowledge of graph algorithms and propose this new course.

4. **Learning Outcomes (By the end of the course students will be expected to):**

1. Demonstrate understanding of basic concepts in Graph Theory.
2. Represent a given graph as adjacency lists or an adjacency matrix.
3. Describe and implement the breadth first search and the depth first search algorithm.
4. Apply Prim and Kruskal algorithm to solve minimal spanning tree problems
5. Apply algorithms to solve shortest path problems.
6. Apply algorithms to solve maximum problems

5. Date of Departmental Approval: March 24, 2021

LEHMAN COLLEGE
OF THE
CITY UNIVERSITY OF NEW YORK

DEPARTMENT OF Computer Science

CURRICULUM CHANGE

1. **Type of change**: New Course.

2.

Department(s)	Computer Science
Career	[x] Undergraduate [ ] Graduate
Academic Level	[x] Regular [ ] Compensatory [ ] Developmental [ ] Remedial
Subject Area	Computer Science
Course Prefix & Number	CMP 476
Course Title	Parallel Algorithms & Architecture
Description	Survey of parallel computer architecture and models of parallel computation with examples of specific algorithms for searching, sorting, numerical algebraic, and combinatorial computations.
Pre / Co Requisites	Prerequisites: CMP 334, CMP 338, MAT 313
Credits	4
Hours	4
Liberal Arts	[X] Yes [ ] No
Course Attribute (e.g. Writing Intensive, WAC, etc)	None
General Education Component	___X___ Not Applicable _______ Required _______ English Composition _______ Mathematics ______ Science _______ Flexible _______ World Cultures _______ US Experience in its Diversity _______ Creative Expression _______ Individual and Society

	______ Scientific World

3. **Rationale:**

For many decades parallel algorithms and architectures were deployed almost exclusively in supercomputers. These were applied to only the most demanding high-performance workloads in the sciences, engineering, and digital signal processing where a serial processor was entirely inadequate to the task. Another niche for parallelism was where a very low power solution was needed for an embarrassingly parallel task such as those found in the use of sensor networks. For most in the field of Computer Science the arena of parallelism has been largely just a very complex academic curiosity examined at the graduate level. In the past decade, however, digital circuit designers/manufacturers have fast approached an immovable physical barrier to one avenue of continued progress. Further miniaturization, which is a major contributor to speed increases and power conservation, will come to a halt without an entirely new foundation for building digital circuits, because circuit sizes are approaching the size of a single atom. This realization has led to new thinking in processor design. An explosion of parallel processing has occurred in commodity CPUs and GPUs. Unfortunately, most computer programmers are ill-prepared to take advantage of these facilities. It is fair to say that parallel algorithm design and implementation is orders of magnitude more difficult than programming serially. This course will introduce students to the arena of parallel algorithms and architecture to better prepare them for changes that have come and are coming to the field.

4. **Learning Outcomes (By the end of the course students will be expected to):**

1. Know the advantages and disadvantages of using parallel versus serial hardware.
2. Know use cases of various parallel hardware configurations.
3. Apply the concept of Thread Level Parallelism.
4. Apply the concept of Instruction Level Parallelism.
5. Be cognizant of the inherent difficulties in devising parallel algorithms and their implementations and know how to overcome these issues.
6. Understand how to navigate memory hierarchies of parallel hardware to efficiently move data to where it is needed.
7. Perform intelligent subdivisions and agglomerations.
8. Recognize components of implementations of parallel primitives such as Prefix Scan and Fast Fourier Transform.
9. Explain how parallel primitives are used in devising parallel algorithms.
10. Be prepared for further study of parallel algorithms and architecture so as to facilitate targeted independent study in the area.

5. **Date of Departmental Approval:** March 24, 2021

LEHMAN COLLEGE
OF THE
CITY UNIVERSITY OF NEW YORK

DEPARTMENT OF Computer Science

CURRICULUM CHANGE

1. **Type of Change:** Credits, Prerequisite/Corequisite

2. **From:** ~~Strikethrough~~ the changes

Department(s)	Computer Science
Career	[X] Undergraduate [ ] Graduate
Academic Level	[X] Regular [ ] Compensatory [ ] Developmental [ ] Remedial
Subject Area	Computer Information System
Course Prefix & Number	CIS 212
Course Title	Micro Computer Architecture
Description	Architecture of microcomputer systems and its supporting system software. Various microprocessor systems, expansion bus design, memory design and management, secondary storage technologies and management, peripherals, and telecommunication technologies.
Pre/ Co Requisites	~~CIS 211~~
Credits	~~3 Credits~~
Hours	4 Hours
Liberal Arts	[ ] Yes [X] No
Course Attribute (e.g. Writing Intensive, WAC, etc)	
General Education Component	__X__ Not Applicable ____ Required ____ English Composition ____ Mathematics ___ Science ____ Flexible ____ World Cultures

	____ US Experience in its Diversity ____ Creative Expression ____ Individual and Society ____ Scientific World

3. **To:** Underline the changes

Department(s)	Computer Science
Career	[X] Undergraduate [ ] Graduate
Academic Level	[X] Regular [ ] Compensatory [ ] Developmental [ ] Remedial
Subject Area	Computer Information Systems
Course Prefix & Number	CIS 212
Course Title	Micro Computer Architecture
Description	Architecture of microcomputer systems and its supporting system software. Various microprocessor systems, expansion bus design, memory design and management, secondary storage technologies and management, peripherals, and telecommunication technologies.
Pre/ Co Requisites	Prerequisite: MAT 104 or higher Co-Requisite: CIS 213
Credits	4 Credits
Hours	4 Hours
Liberal Arts	[ ] Yes [X] No
Course Attribute (e.g. Writing Intensive, WAC, etc)	
General Education Component	__X__ Not Applicable ____ Required ____ English Composition ____ Mathematics ___ Science ____ Flexible ____ World Cultures ____ US Experience in its Diversity ____ Creative Expression ____ Individual and Society ____ Scientific World

4. **<u>Rationale (Explain how this change will impact the learning outcomes of the department and Major/Program)</u>:**
We remove the prerequisite of CIS 211 because CIS 211 is no longer a required course for the CIS undergraduate program. No previous knowledge of computer architecture will be assumed for students entering CIS 212.

We increased one credit for this course because the extensive in-depth study of Microarchitecture Components dictates that CIS 212 should be a 4-hour course. Firstly, we plan to remove the prerequisite CIS 211, which means that this course will also include material previously assumed to be known. Secondly, microarchitecture, and computer architecture in general has changed at an unimaganeably rapid pace since the inception of CIS 212. As a result, the course content has evolved and expanded since this course's creation to consist of modern changes and extensive in-depth exploration of the technological advancements. The current allotted credits do not reflect the commitment required from the students.
Application of concepts learned will be applied in the Microcomputer Architecture Lab CIS 213. As a result, CIS 213 will be listed as a co-requisite.

5. **<u>Date of departmental approval</u>:** March 24, 2021

LEHMAN COLLEGE
OF THE
CITY UNIVERSITY OF NEW YORK

DEPARTMENT OF Computer Science

CURRICULUM CHANGE

1. **Type of Change**: Prerequisite/Corequisite, description

2. **From:** ~~Strikethrough~~ the changes

Department(s)	Computer Science
Career	[X] Undergraduate [ ] Graduate
Academic Level	[X] Regular [ ] Compensatory [ ] Developmental [ ] Remedial
Subject Area	Computer Information System
Course Prefix & Number	CIS 234
Course Title	Introduction to Spreadsheet Analysis
Description	Use of spreadsheet software for elementary data analysis, simple modeling and forecasting. ~~Worksheets, files, graphs, and formatted output and screen presentation. M~~acro programming.
Pre/ Co Requisites	~~MAT 171 and a grade of B- or better in CIS 166~~
Credits	3 Credits
Hours	4 Hours
Liberal Arts	[ ] Yes [X] No
Course Attribute (e.g. Writing Intensive, WAC, etc)	
General Education Component	__X__ Not Applicable ____ Required ____ English Composition ____ Mathematics ___ Science ____ Flexible ____ World Cultures

	____ US Experience in its Diversity ____ Creative Expression ____ Individual and Society ____ Scientific World

3. **To:** Underline the changes

Department(s)	Computer Science
Career	[X] Undergraduate [ ] Graduate
Academic Level	[X] Regular [ ] Compensatory [ ] Developmental [ ] Remedial
Subject Area	Computer Information Systems
Course Prefix & Number	CIS 234
Course Title	Computer Group Productivity Tools
Description	Use of spreadsheet software for elementary data analysis, simple <u>modeling, forecasting and macro programming.</u>
Pre/ Co Requisites	<u>Prerequisite: CMP 157, CMP 167, MAT 171/MAT 172</u>
Credits	3 Credits
Hours	4 Hours
Liberal Arts	[ ] Yes [X] No
Course Attribute (e.g. Writing Intensive, WAC, etc)	
General Education Component	__X__ Not Applicable ____ Required ____ English Composition ____ Mathematics ___ Science ____ Flexible ____ World Cultures ____ US Experience in its Diversity ____ Creative Expression ____ Individual and Society ____ Scientific World

4. **<u>Rationale (Explain how this change will impact the learning outcomes of the department and Major/Program)</u>:**
The prerequisites of this course are being changed due to the withdrawal of two courses (CIS 166 and CIS 211) from the Computer Information Systems Major Requirements and the addition of two courses (CMP 157 and CMP 167) to the Computer Information Systems Major Requirements.

5. **<u>Date of departmental approval</u>:** March 24, 2021

LEHMAN COLLEGE
OF THE
CITY UNIVERSITY OF NEW YORK

DEPARTMENT OF Computer Science

CURRICULUM CHANGE

1. **Type of Change:** Prerequisite

2. **From:** ~~Strikethrough~~ the changes

Department(s)	Computer Science
Career	[X] Undergraduate [ ] Graduate
Academic Level	[X] Regular [ ] Compensatory [ ] Developmental [ ] Remedial
Subject Area	Computer Information System
Course Prefix & Number	CIS 242
Course Title	Introduction to Systems Analysis and Design
Description	Study of a computer system life cycle hardware and software organization.
Pre/ Co Requisites	~~CIS 211~~
Credits	4 Credits
Hours	4 Hours
Liberal Arts	[ ] Yes [X] No
Course Attribute (e.g. Writing Intensive, WAC, etc)	
General Education Component	__X__ Not Applicable ____ Required ____ English Composition ____ Mathematics ____ Science ____ Flexible ____ World Cultures ____ US Experience in its Diversity

	____ Creative Expression ____ Individual and Society ____ Scientific World

3. **To:** Underline the changes

Department(s)	Computer Science
Career	[X] Undergraduate [ ] Graduate
Academic Level	[X] Regular [ ] Compensatory [ ] Developmental [ ] Remedial
Subject Area	Computer Information Systems
Course Prefix & Number	CIS 242
Course Title	Introduction to Systems Analysis and Design
Description	Study of a computer system life cycle hardware and software organization.
Pre/ Co Requisites	<u>Prerequisite: CMP 157, CMP 167</u>
Credits	4 Credits
Hours	4 Hours
Liberal Arts	[ ] Yes [X] No
Course Attribute (e.g. Writing Intensive, WAC, etc)	
General Education Component	__X__ Not Applicable ____ Required ____ English Composition ____ Mathematics ___ Science ____ Flexible ____ World Cultures ____ US Experience in its Diversity ____ Creative Expression ____ Individual and Society ____ Scientific World

4. **Rationale (Explain how this change will impact the learning outcomes of the department and Major/Program):**
The prerequisites of this course are being changed due to the withdrawal of two courses (CIS 166 and CIS 211) from the Computer Information Systems Major Requirements and the addition of two courses (CMP 157 and CMP 167) to the Computer Information Systems Major Requirements in lieu of the removed courses.

5. **Date of departmental approval:** March 24, 2021

LEHMAN COLLEGE
OF THE
CITY UNIVERSITY OF NEW YORK

DEPARTMENT OF Computer Science

CURRICULUM CHANGE

1. **Type of Change:** Prerequisite

2. **From:** ~~Strikethrough~~ the changes

Department(s)	Computer Science
Career	[X] Undergraduate [ ] Graduate
Academic Level	[X] Regular [ ] Compensatory [ ] Developmental [ ] Remedial
Subject Area	Computer Information System
Course Prefix & Number	CIS 244
Course Title	Introduction to Database Management
Description	Fundamental concepts of database organization: fields, records, tables, indexes, queries, forms, reports. Use of a relational database system for loading, modifying and querying a database. Programming in a database language.
Pre/ Co Requisites	~~CIS 166 with a minimum grade of B-~~
Credits	3 Credits
Hours	4 Hours
Liberal Arts	[ ] Yes [X] No
Course Attribute (e.g. Writing Intensive, WAC, etc)	
General Education Component	__X__ Not Applicable ____ Required ____ English Composition ____ Mathematics ___ Science ____ Flexible

	____ World Cultures ____ US Experience in its Diversity ____ Creative Expression ____ Individual and Society ____ Scientific World

3. **<u>To</u>:** <u>Underline</u> the changes

Department(s)	Computer Science
Career	[X] Undergraduate [ ] Graduate
Academic Level	[X] Regular [ ] Compensatory [ ] Developmental [ ] Remedial
Subject Area	Computer Information Systems
Course Prefix & Number	CIS 244
Course Title	Introduction to Database Management
Description	Fundamental concepts of database organization: fields, records, tables, indexes, queries, forms, reports. Use of a relational database system for loading, modifying and querying a database. Programming in a database language.
Pre/ Co Requisites	<u>Prerequisite: CMP 157, CMP 167</u>
Credits	3 Credits
Hours	4 Hours
Liberal Arts	[ ] Yes [X] No
Course Attribute (e.g. Writing Intensive, WAC, etc)	
General Education Component	__X__ Not Applicable ____ Required ____ English Composition ____ Mathematics ___ Science ____ Flexible ____ World Cultures ____ US Experience in its Diversity ____ Creative Expression ____ Individual and Society

	____ Scientific World

4. **<u>Rationale (Explain how this change will impact the learning outcomes of the department and Major/Program)</u>:**
The prerequisites of this course are being changed due to the withdrawal of two courses (CIS 166 and CIS 211) from the Computer Information Systems Major Requirements and the addition of two courses (CMP 157 and CMP 167) to the Computer Information Systems Major Requirements in lieu of the removed courses

5. **<u>Date of departmental approval</u>:** March 24, 2021

LEHMAN COLLEGE
OF THE
CITY UNIVERSITY OF NEW YORK

DEPARTMENT OF Computer Science

CURRICULUM CHANGE

1. **Type of Change:** Credits, Prerequisite/Corequisite

2. **From:** ~~Strikethrough~~ the changes

Department(s)	Computer Science
Career	[X] Undergraduate [ ] Graduate
Academic Level	[X] Regular [ ] Compensatory [ ] Developmental [ ] Remedial
Subject Area	Computer Information System
Course Prefix & Number	CIS 247
Course Title	Practical Unix Programming and System Administration
Description	Topics chosen from the following: text editors, file system, utility programs, pipe and filter paradigms, shell language programming; tools for maintenance of normal system operation, security, hardware and software configuration management and network connections.
Pre/ Co Requisites	~~CIS-166 and CIS-211~~.
Credits	~~3 Credit~~s
Hours	4 Hours
Liberal Arts	[ ] Yes [X] No
Course Attribute (e.g. Writing Intensive, WAC, etc)	
General Education Component	__X__ Not Applicable ____ Required ____ English Composition ____ Mathematics ____ Science

	____ Flexible
	    ____ World Cultures
	    ____ US Experience in its Diversity
	    ____ Creative Expression
	    ____ Individual and Society
	    ____ Scientific World

3. **<u>To:</u>** <u>Underline</u> the changes

Department(s)	Computer Science
Career	[X] Undergraduate [ ] Graduate
Academic Level	[X] Regular [ ] Compensatory [ ] Developmental [ ] Remedial
Subject Area	Computer Information Systems
Course Prefix & Number	CIS 247
Course Title	Practical Unix Programming and System Administration
Description	Topics chosen from the following: text editors, file system, utility programs, pipe and filter paradigms, shell language programming; tools for maintenance of normal system operation, security, hardware and software configuration management and network connections.
Pre/ Co Requisites	<u>CMP157, CMP 167</u>
Credits	<u>4 Credits</u>
Hours	4 Hours
Liberal Arts	[ ] Yes [X] No
Course Attribute (e.g. Writing Intensive, WAC, etc)	

General Education Component	__X__ Not Applicable
	____ Required
	    ____ English Composition
	    ____ Mathematics
	    ____ Science
	
	____ Flexible
	    ____ World Cultures
	    ____ US Experience in its Diversity

	____ Creative Expression ____ Individual and Society ____ Scientific World

4. **Rationale (Explain how this change will impact the learning outcomes of the department and Major/Program):**
The current allotted credits do not reflect the commitment required from the students. This is a 4 hour course consisting of Practical Unix Programming and System Administration, where students will be involved in experiential learning assignments and projects. Since the creation of the course, the content has evolved a lot, to consist of both modern changes and extensive exploration and study. Moreover, the prerequisites of this course are being changed due to the withdrawal of two courses (CIS 166 and CIS 211) from the Computer Information Systems Major Requirements and the addition of two courses (CMP 157 and CMP 167) to the Computer Information Systems Major Requirements.The prerequisites of this course are being changed due to the withdrawal of two courses (CIS 166 and CIS 211) from the Computer Information Systems Major Requirements in lieu of the removed courses.

5. **Date of departmental approval:** March 24, 2021

LEHMAN COLLEGE
OF THE
CITY UNIVERSITY OF NEW YORK

DEPARTMENT OF Computer Science

CURRICULUM CHANGE

1. **Type of Change:** Prerequisite/Corequisite

2. **From:** ~~Strikethrough~~ the changes

Department(s)	Computer Science
Career	[X] Undergraduate [ ] Graduate
Academic Level	[X] Regular [ ] Compensatory [ ] Developmental [ ] Remedial
Subject Area	Computer Information System

Course Prefix & Number	CIS 266
Course Title	Computer Programming for Information Processing II
Description	Techniques of business information processing using object-oriented programming. Random access files, data management and control, variable arrays, object variables. Introduction to advanced techniques.
Pre/ Co Requisites	~~A grade of C or better in CIS 166~~
Credits	4 Credits
Hours	4 Hours
Liberal Arts	[ ] Yes [X] No
Course Attribute (e.g. Writing Intensive, WAC, etc)	
General Education Component	__X__ Not Applicable ____ Required ____ English Composition ____ Mathematics ___ Science ____ Flexible ____ World Cultures ____ US Experience in its Diversity ____ Creative Expression ____ Individual and Society ____ Scientific World

3. **To:** Underline the changes

Department(s)	Computer Science
Career	[X] Undergraduate [ ] Graduate
Academic Level	[X] Regular [ ] Compensatory [ ] Developmental [ ] Remedial
Subject Area	Computer Information Systems
Course Prefix & Number	CIS 266
Course Title	Computer Programming for Information Processing II

Description	Techniques of business information processing using object-oriented programming. Random access files, data management and control, variable arrays, object variables. Introduction to advanced techniques.
Pre/ Co Requisites	CMP 157, CMP 167
Credits	4 Credits
Hours	4 Hours
Liberal Arts	[ ] Yes [X] No
Course Attribute (e.g. Writing Intensive, WAC, etc)	
General Education Component	_X___ Not Applicable ____ Required ____ English Composition ____ Mathematics ____ Science ____ Flexible ____ World Cultures ____ US Experience in its Diversity ____ Creative Expression ____ Individual and Society ____ Scientific World

4. **Rationale (Explain how this change will impact the learning outcomes of the department and Major/Program):**
The prerequisites of this course are being changed due to the withdrawal of two courses (CIS 166 and CIS 211) from the Computer Information Systems Major Requirements and the addition of two courses (CMP 157 and CMP 167) to the Computer Information Systems Major Requirements.
Therefore the grade requirement in the prerequisite course CIS 166 is not needed and in fact the CIS 166 course is not needed as a prerequisite since it is being removed.

5. **Date of departmental approval:** March 24, 2021

LEHMAN COLLEGE
OF THE
CITY UNIVERSITY OF NEW YORK

DEPARTMENT OF Computer Science

CURRICULUM CHANGE

1. **Type of Change:** Prerequisite/Corequisite

2. **From:** ~~Strikethrough~~ the changes

Department(s)	Computer Science
Career	[X] Undergraduate [ ] Graduate
Academic Level	[X] Regular [ ] Compensatory [ ] Developmental [ ] Remedial
Subject Area	Computer Information System
Course Prefix & Number	CIS 329
Course Title	Local Area Networks (LAN's)
Description	An overview of LANs as well as hands-on introduction to a popular network operating system. General topics will include LAN media, topologies, protocols, multiplatform connectivity, remote access, and rudimentary internet working.
Pre/ Co Requisites	~~CIS 211~~, CIS 212
Credits	3 Credits
Hours	4 Hours
Liberal Arts	[X] Yes [ ] No
Course Attribute (e.g. Writing Intensive, WAC, etc)	
General Education Component	_X___ Not Applicable ____ Required ____ English Composition ____ Mathematics ___ Science

	____ Flexible ____ World Cultures ____ US Experience in its Diversity ____ Creative Expression ____ Individual and Society ____ Scientific World

3. **To:** Underline the changes

Department(s)	Computer Science
Career	[X] Undergraduate [ ] Graduate
Academic Level	[X] Regular [ ] Compensatory [ ] Developmental [ ] Remedial
Subject Area	Computer Information Systems
Course Prefix & Number	CIS 329
Course Title	Local Area Networks (LAN's)
Description	An overview of LANs as well as hands-on introduction to a popular network operating system. General topics will include LAN media, topologies, protocols, multiplatform connectivity, remote access, and rudimentary internet working.
Pre/ Co Requisites	Prerequisites: CMP157, CMP167 Pre or Corequisite CIS 212
Credits	3 Credits
Hours	4 Hours
Liberal Arts	[X] Yes [ ] No
Course Attribute (e.g. Writing Intensive, WAC, etc)	
General Education Component	__X__ Not Applicable ____ Required ____ English Composition ____ Mathematics ____ Science ____ Flexible ____ World Cultures ____ US Experience in its Diversity ____ Creative Expression

	____ Individual and Society ____ Scientific World

4. **<u>Rationale (Explain how this change will impact the learning outcomes of the department and Major/Program)</u>:**
The prerequisites of this course are being changed due to the withdrawal of two courses (CIS 166 and CIS 211) from the Computer Information Systems Major Requirements and the addition of two courses (CMP 157 and CMP 167) to the Computer Information Systems Major Requirements.
CIS 212 remains as a corequisite for the current course.

5. **<u>Date of departmental approval</u>:** March 24, 2021

LEHMAN COLLEGE
OF THE
CITY UNIVERSITY OF NEW YORK

DEPARTMENT OF Computer Science

CURRICULUM CHANGE

1. **Type of Change:** Prerequisite/Corequisite

2. **From:** ~~Strikethrough~~ the changes

Department(s)	Computer Science
Career	[X] Undergraduate [ ] Graduate
Academic Level	[X] Regular [ ] Compensatory [ ] Developmental [ ] Remedial
Subject Area	Computer Information System
Course Prefix & Number	CIS 331
Course Title	Network Introduction
Description	Introduction to network technologies (Ethernet, ATM, WiFi, Bluetooth, ZigBee), network architectures (telephone, OSI, and Internet), and standard tools for administering and monitoring networks. Evaluation of network technologies and designs for supporting some services; design and configuring networks for those services.
Pre/ Co Requisites	~~CIS 211,~~ CIS 212
Credits	3 Credits
Hours	4 Hours
Liberal Arts	[ ] Yes [X] No
Course Attribute (e.g. Writing Intensive, WAC, etc)	
General Education Component	__X__ Not Applicable ____ Required ____ English Composition ____ Mathematics ____ Science

	____ Flexible ____ World Cultures ____ US Experience in its Diversity ____ Creative Expression ____ Individual and Society ____ Scientific World

3. **To:** Underline the changes

Department(s)	Computer Science
Career	[X] Undergraduate [ ] Graduate
Academic Level	[X] Regular [ ] Compensatory [ ] Developmental [ ] Remedial
Subject Area	Computer Information Systems
Course Prefix & Number	CIS 331
Course Title	Network Introduction
Description	Introduction to network technologies (Ethernet, ATM, WiFi, Bluetooth, ZigBee), network architectures (telephone, OSI, and Internet), and standard tools for administering and monitoring networks. Evaluation of network technologies and designs for supporting some services; design and configuring networks for those services.
Pre/ Co Requisites	<u>Prerequisites: CMP157, CMP167, CIS 212</u>
Credits	3 Credits
Hours	4 Hours
Liberal Arts	[ ] Yes [X] No
Course Attribute (e.g. Writing Intensive, WAC, etc)	
General Education Component	__X__ Not Applicable ____ Required ____ English Composition ____ Mathematics ____ Science

	____ Flexible ____ World Cultures ____ US Experience in its Diversity ____ Creative Expression ____ Individual and Society ____ Scientific World

4. **Rationale (Explain how this change will impact the learning outcomes of the department and Major/Program):**
The prerequisites of this course are being changed due to the withdrawal of two courses (CIS 166 and CIS 211) from the Computer Information Systems Major Requirements and the addition of two courses (CMP 157 and CMP 167) to the Computer Information Systems Major Requirements.
The existing prerequisite of CIS 212 is still a prerequisite for the current course.

5. **Date of departmental approval:** March 24, 2021

LEHMAN COLLEGE
OF THE
CITY UNIVERSITY OF NEW YORK

DEPARTMENT OF Computer Science

CURRICULUM CHANGE

1. **Type of Change:** Prerequisite/Corequisite

2. **From:** ~~Strikethrough~~ the changes

Department(s)	Computer Science
Career	[X] Undergraduate [ ] Graduate
Academic Level	[X] Regular [ ] Compensatory [ ] Developmental [ ] Remedial
Subject Area	Computer Information Systems
Course Prefix & Number	CIS 341
Course Title	Computer System Fundamentals
Description	Examination, removal, and reassembly of computer hardware components, such as processors, disks, memory, and buses. Installing and operating the following computer system components: operating systems, user interfaces, subsystems (such as Web servers), development environments, communications, and distributed file systems. Performance characteristics also will be discussed and measured.
Pre/ Co Requisites	~~CIS 211,~~ CIS 212
Credits	3 Credits
Hours	4 Hours
Liberal Arts	[X] Yes [ ] No
Course Attribute (e.g. Writing Intensive, WAC, etc)	

General Education Component	__X__ Not Applicable _____ Required _____ English Composition _____ Mathematics ___ Science _____ Flexible _____ World Cultures _____ US Experience in its Diversity _____ Creative Expression _____ Individual and Society _____ Scientific World

3. **To:** Underline the changes

Department(s)	Computer Science
Career	[X] Undergraduate [ ] Graduate
Academic Level	[X] Regular [ ] Compensatory [ ] Developmental [ ] Remedial
Subject Area	Computer Information Systems
Course Prefix & Number	CIS 341
Course Title	Computer System Fundamentals
Description	Examination, removal, and reassembly of computer hardware components, such as processors, disks, memory, and buses. Installing and operating the following computer system components: operating systems, user interfaces, subsystems (such as Web servers), development environments, communications, and distributed file systems. Performance characteristics also will be discussed and measured.
Pre/ Co Requisites	<u>Prerequisites: CMP157, CMP167</u> <u>Pre or Corequisite: CIS 212</u>
Credits	3 Credits
Hours	4 Hours
Liberal Arts	[X] Yes [ ] No
Course Attribute (e.g.	

Writing Intensive, WAC, etc)	
General Education Component	__X__ Not Applicable ____ Required ____ English Composition ____ Mathematics ___ Science ____ Flexible ____ World Cultures ____ US Experience in its Diversity ____ Creative Expression ____ Individual and Society ____ Scientific World

4. **Rationale (Explain how this change will impact the learning outcomes of the department and Major/Program):**
The prerequisites of this course are being changed due to the withdrawal of two courses (CIS 166 and CIS 211) from the Computer Information Systems Major Requirements and the addition of two courses (CMP 157 and CMP 167) to the Computer Information Systems Major Requirements.
The existing corequisite of CIS 212 is still a corequisite for the current course.

5. **Date of departmental approval:** March 24, 2021

LEHMAN COLLEGE
OF THE
CITY UNIVERSITY OF NEW YORK

DEPARTMENT OF Computer Science

CURRICULUM CHANGE

1. **Type of Change:** Prerequisite/Corequisite

2. **From:** ~~Strikethrough~~ the changes

Department(s)	Computer Science
Career	[X] Undergraduate [ ] Graduate
Academic Level	[X] Regular [ ] Compensatory [ ] Developmental [ ] Remedial
Subject Area	Computer Information System
Course Prefix & Number	CIS 344
Course Title	Database Design and Programming
Description	Programming in a database language. Emphasis on programming ideas and techniques and user interfaces in a modern database system. Review of elementary relational database concepts, with emphasis on programming rather than theory.
Pre/ Co Requisites	~~CIS 166~~, CIS 244
Credits	3 Credits
Hours	4 Hours
Liberal Arts	[X] Yes [ ] No
Course Attribute (e.g. Writing Intensive, WAC, etc)	
General Education Component	__X__ Not Applicable ____ Required ____ English Composition ____ Mathematics ____ Science ____ Flexible

	____ World Cultures ____ US Experience in its Diversity ____ Creative Expression ____ Individual and Society ____ Scientific World

3. **<u>To</u>:** <u>Underline</u> the changes

Department(s)	Computer Science
Career	[X] Undergraduate [ ] Graduate
Academic Level	[X] Regular [ ] Compensatory [ ] Developmental [ ] Remedial
Subject Area	Computer Information Systems
Course Prefix & Number	CIS 344
Course Title	Database Design and Programming
Description	Programming in a database language. Emphasis on programming ideas and techniques and user interfaces in a modern database system. Review of elementary relational database concepts, with emphasis on programming rather than theory.
Pre/ Co Requisites	<u>Prerequisites: CMP157, CMP 167</u>, CIS 244
Credits	3 Credits
Hours	4 Hours
Liberal Arts	[X] Yes [ ] No
Course Attribute (e.g. Writing Intensive, WAC, etc)	
General Education Component	__X__ Not Applicable ____ Required ____ English Composition ____ Mathematics ____ Science ____ Flexible ____ World Cultures ____ US Experience in its Diversity ____ Creative Expression ____ Individual and Society ____ Scientific World

4. **<u>Rationale (Explain how this change will impact the learning outcomes of the department and Major/Program)</u>:**
The prerequisites of this course are being changed due to the withdrawal of two courses (CIS 166 and CIS 211) from the Computer Information Systems Major Requirements and the addition of two courses (CMP 157 and CMP 167) to the Computer Information Systems Major Requirements.
The previously stated prerequisite of CIS 244 remains as a prerequisite for the current course.

5. **<u>Date of departmental approval</u>:** March 24, 2021

LEHMAN COLLEGE
OF THE
CITY UNIVERSITY OF NEW YORK

DEPARTMENT OF Computer Science

CURRICULUM CHANGE

1. **Type of Change:** Credits, Prerequisite/Corequisite

2. **From:** ~~Strikethrough~~ the changes

Department(s)	Computer Science
Career	[X] Undergraduate [ ] Graduate
Academic Level	[X] Regular [ ] Compensatory [ ] Developmental [ ] Remedial
Subject Area	Computer Information System
Course Prefix & Number	CIS 345
Course Title	Operating Systems
Description	Overview of operating systems (O.S.) from both a theoretical and a systems manager point of view. Process management and multitasking, memory management, resource management, file management, I/O management, command interpreter/shell, and shell scripts will be covered. Large networked systems will be discussed from a systems point of view.
Pre/ Co Requisites	~~CHE 166, CIS 211,~~ CIS 212
Credits	~~3~~ credit
Hours	4 hours
Liberal Arts	[ ] Yes [X] No
Course Attribute (e.g. Writing Intensive, WAC, etc)	
General Education Component	__X_ Not Applicable ____ Required _____ English Composition _____ Mathematics _____ Science

	____ Flexible ____ World Cultures ____ US Experience in its Diversity ____ Creative Expression ____ Individual and Society ____ Scientific World

3. **<u>To</u>:** <u>Underline</u> the changes

Department(s)	Computer Science
Career	[X] Undergraduate [ ] Graduate
Academic Level	[X] Regular [ ] Compensatory [ ] Developmental [ ] Remedial
Subject Area	Computer Information Systems
Course Prefix & Number	CIS 345
Course Title	Operating Systems
Description	Overview of operating systems (O.S.) from both a theoretical and a systems manager point of view. Process management and multitasking, memory management, resource management, file management, I/O management, command interpreter/shell, and shell scripts will be covered. Large networked systems will be discussed from a systems point of view.
Pre/ Co Requisites	<u>Prerequisites: CMP 157, CMP 167, CIS 212</u>
Credits	<u>4 credit</u>
Hours	4 hours
Liberal Arts	[ ] Yes [x] No
Course Attribute (e.g. Writing Intensive, WAC, etc)	
General Education Component	__X__ Not Applicable ____ Required ____ English Composition ____ Mathematics ____ Science ____ Flexible ____ World Cultures

	____ US Experience in its Diversity ____ Creative Expression ____ Individual and Society ____ Scientific World

4. **Rationale (Explain how this change will impact the learning outcomes of the department and Major/Program):**
The current allotted credits do not reflect the commitment required from the students. This is a 4 hour course with in depth study of Operating Systems, where the content has evolved to consist of extensive exploration and study. Furthermore, there was an error in the previous bulletin listing where CHE166 was stated to be the prerequisite instead of CIS166. The prerequisites of this course are being changed due to the withdrawal of two courses (CIS 166 and CIS 211) from the Computer Information Systems Major Requirements and the addition of two courses (CMP 157 and CMP 167) to the Computer Information Systems Major Requirements in lieu of the removed courses. CIS166 has been replaced by CMP157 and CMP 167, therefore the erroneous listing of CHE166 is being corrected by replacing it with CMP 157 and CMP167.

CIS 212 remains as a prerequisite for the current course.

5. **Date of departmental approval:** March 24, 2021

LEHMAN COLLEGE
OF THE
CITY UNIVERSITY OF NEW YORK

DEPARTMENT OF Computer Science

CURRICULUM CHANGE

1. **Type of Change:** Prerequisite and Credits

2. **From:** ~~Strikethrough~~ the changes

Department(s)	Computer Science
Career	[X] Undergraduate [ ] Graduate
Academic Level	[X] Regular [ ] Compensatory [ ] Developmental [ ] Remedial
Subject Area	Computer Information System
Course Prefix & Number	CIS 346
Course Title	E-Commerce
Description	Introduction to electronic commerce on the Internet: Designing an e-commerce site including web Server installation, configuration, and tuning; web page content and development, site marketing and advertisement, legal and security considerations, shopping cart management, credit card and other debit transactions.
Pre/ Co Requisites	~~CIS 166, CIS 211~~
Credits	~~3 Credits~~
Hours	4 Hours
Liberal Arts	[ ] Yes [X] No
Course Attribute (e.g. Writing Intensive, WAC, etc)	
General Education Component	__X__ Not Applicable ____ Required ____ English Composition ____ Mathematics ____ Science

	____ Flexible ____ World Cultures ____ US Experience in its Diversity ____ Creative Expression ____ Individual and Society ____ Scientific World

3. **To:** Underline the changes

Department(s)	Computer Science
Career	[X] Undergraduate [ ] Graduate
Academic Level	[X] Regular [ ] Compensatory [ ] Developmental [ ] Remedial
Subject Area	Computer Information Systems
Course Prefix & Number	CIS 346
Course Title	E-Commerce
Description	Introduction to electronic commerce on the Internet: Designing an e-commerce site including web Server installation, configuration, and tuning; web page content and development, site marketing and advertisement, legal and security considerations, shopping cart management, credit card and other debit transactions.
Pre/ Co Requisites	<u>CMP157, CMP 167</u>
Credits	<u>4 Credits</u>
Hours	4 Hours
Liberal Arts	[ ] Yes [X] No
Course Attribute (e.g. Writing Intensive, WAC, etc)	
General Education Component	__X__ Not Applicable ____ Required ____ English Composition ____ Mathematics ____ Science ____ Flexible ____ World Cultures ____ US Experience in its Diversity

	____ Creative Expression ____ Individual and Society ____ Scientific World

4. **Rationale (Explain how this change will impact the learning outcomes of the department and Major/Program):**
The prerequisites of this course are being changed due to the withdrawal of two courses (CIS 166 and CIS 211) from the Computer Information Systems Major Requirements. The replacement of the prerequisite courses are CMP 157, CMP 167 which are the two programming courses that have been brought into the Computer Information Systems Major Requirements.
The change in credits is due to the required expansion of curriculum and the study of e-commerce since the inception of the course. The realm of e-commerce has undergone many changes in recent years, requiring technologists to vastly expand their knowledge and understanding of content development, web server installation and configuration, user interface creation, marketing, advertisement, privacy, security, financial transactions, shopping cart management and legal considerations. The immense changes in technology, the internet, and e-commerce in general dictate the requirement to expand the content of the course curriculum and thus the number of credits associated with the CIS 346 E-Commerce course.

5. **Date of departmental approval:** March 24, 2021

LEHMAN COLLEGE
OF THE
CITY UNIVERSITY OF NEW YORK

DEPARTMENT OF Computer Science

CURRICULUM CHANGE

1. **Type of Change:** Prerequisite/Corequisite

2. **From:** ~~Strikethrough~~ the changes

Department(s)	Computer Science
Career	[X] Undergraduate [ ] Graduate
Academic Level	[X] Regular [ ] Compensatory [ ] Developmental [ ] Remedial
Subject Area	Computer Information System
Course Prefix & Number	CIS 349
Course Title	Data Communications and Distributed Networks
Description	Data communications: standard models, system operations, major components, digital transmission (including some current schemes such as SONET, ISDN and ATM). Hands-on introduction to local area network architectures, link-layer protocols and their design and analysis.
Pre/ Co Requisites	~~CIS 211~~, CIS 212
Credits	4 Credits
Hours	4 Hours
Liberal Arts	[X] Yes [ ] No
Course Attribute (e.g. Writing Intensive, WAC, etc)	
General Education Component	__X__ Not Applicable ____ Required ____ English Composition ____ Mathematics ____ Science

	____ Flexible ____ World Cultures ____ US Experience in its Diversity ____ Creative Expression ____ Individual and Society ____ Scientific World

3. **To:** Underline the changes

Department(s)	Computer Science
Career	[X] Undergraduate [ ] Graduate
Academic Level	[X] Regular [ ] Compensatory [ ] Developmental [ ] Remedial
Subject Area	Computer Information Systems
Course Prefix & Number	CIS 349
Course Title	Data Communications and Distributed Networks
Description	Data communications: standard models, system operations, major components, digital transmission (including some current schemes such as SONET, ISDN and ATM). Hands-on introduction to local area network architectures, link-layer protocols and their design and analysis.
Pre/ Co Requisites	Prerequisites: CMP157, CMP167, CIS 212
Credits	4 Credits
Hours	4 Hours
Liberal Arts	[X] Yes [ ] No
Course Attribute (e.g. Writing Intensive, WAC, etc)	
General Education Component	__X__ Not Applicable ____ Required ____ English Composition ____ Mathematics ___ Science ____ Flexible ____ World Cultures

	____ US Experience in its Diversity ____ Creative Expression ____ Individual and Society ____ Scientific World

4. **Rationale (Explain how this change will impact the learning outcomes of the department and Major/Program):**
The prerequisites of this course are being changed due to the withdrawal of two courses (CIS 166 and CIS 211) from the Computer Information Systems Major Requirements and the addition of two courses (CMP 157 and CMP 167) to the Computer Information Systems Major Requirements in lieu of the removed courses.

5. **Date of departmental approval:** March 24, 2021

LEHMAN COLLEGE
OF THE
CITY UNIVERSITY OF NEW YORK

DEPARTMENT OF Computer Science

CURRICULUM CHANGE

1. **Type of Change:** Credits, Prerequisite/Corequisite, Description

2. **From:** ~~Strikethrough~~ the changes

Department(s)	Computer Science
Career	[X] Undergraduate [ ] Graduate
Academic Level	[X] Regular [ ] Compensatory [ ] Developmental [ ] Remedial
Subject Area	Computer Science
Course Prefix & Number	CMP 167
Course Title	Programming Methods I
Description	Structured computer programming using a modern high-level programming language. Includes console I/O, data types, variables, control structures, including iteration, arrays, function definitions and

	calls, parameter passing, functional decomposition, and an introduction to objects. Debugging techniques. Note: For students who intend to major in Computer Science, Mathematics, Computer Graphics and Imaging, or the sciences. Some previous computer programming experience is recommended. Not intended for students in Accounting or Computer Information Systems~~; the technical content is the same as CIS 166 but the emphasis is different.~~
Pre/ Co Requisites	Prerequisite MAT 104 ~~or department placement~~
Credits	~~3 Credits~~
Hours	4 Hours
Liberal Arts	[X] Yes [ ] No
Course Attribute (e.g. Writing Intensive, WAC, etc)	
General Education Component	__X__ Not Applicable ____ Required ____ English Composition ____ Mathematics ___ Science ____ Flexible ____ World Cultures ____ US Experience in its Diversity ____ Creative Expression ____ Individual and Society ____ Scientific World

3. **To:** Underline the changes

Department(s)	Computer Science
Career	[X] Undergraduate [ ] Graduate
Academic Level	[X] Regular [ ] Compensatory [ ] Developmental [ ] Remedial
Subject Area	Computer Science

Course Prefix & Number	CMP 167
Course Title	Programming Methods I
Description	Structured computer programming using a modern high-level programming language. Includes console I/O, data types, variables, control structures, including iteration, arrays, function definitions and calls, parameter passing, functional decomposition, and an introduction to objects. Debugging techniques. Note: For students who intend to major in Computer Science, Mathematics, Computer Graphics and Imaging, or the sciences. Some previous computer programming experience is recommended. Not intended for students in Accounting or Computer Information Systems.
Pre/ Co Requisites	Prerequisite: MAT 104 or higher Corequisite: CMP157
Credits	4 Credits
Hours	4 Hours
Liberal Arts	[X] Yes [ ] No
Course Attribute (e.g. Writing Intensive, WAC, etc)	
General Education Component	____ Not Applicable ____Required ____ English Composition ____ Mathematics ____ Science ____ Flexible ____ World Cultures ____ US Experience in its Diversity ____ Creative Expression ____ Individual and Society ____ Scientific World

4. **Rationale (Explain how this change will impact the learning outcomes of the department and Major/Program):**

The current allotted credits do not reflect the commitment required from the students. This is a 4 hour course for students who intend to major in Computer Science, Mathematics, Computer Graphics and Imaging, or the sciences. Students will be involved in experiential learning assignments and projects.

The CMP 167 course was initially 4 credits 4 hours and was converted to 3 credits 4 hours approximately a decade ago as part of changes made to the curriculum. The reversion to the initial 4 credits is overdue for many years now for multiple reasons. The CMP 167 course has undergone changes to the curriculum and requires extensive time and dedication from the students so they can learn the concepts of programming and be able to apply those concepts. This course is a 4 hour 4 credit lecture that has an associated lab component.
The prerequisite for CMP 167 has changed from MAT 104 to MAT 108 due to the removal of MAT 104 from the catalog.

5. **<u>Date of departmental approval</u>:** March 01, 2022

LEHMAN COLLEGE
OF THE
CITY UNIVERSITY OF NEW YORK

DEPARTMENT OF COMPUTER SCIENCE

CURRICULUM CHANGE

1. **Type of Change:** *Corequisite, Description, General Education Component*

2. **From:** ~~Strikethrough~~ the changes

Department(s)	Computer Science
Career	[x] Undergraduate [ ] Graduate
Academic Level	[x] Regular [ ] Compensatory [ ] Developmental [ ] Remedial
Subject Area	Computer Science
Course Prefix & Number	CMP 168
Course Title	Programming Methods II
Description	Continuation of parameter passing with a focus on devising function definitions and tracing recursive calls. Sorting and searching algorithms and a comparison of their performance. GUI programming. ~~Threads,~~ Exceptions and Exception Handling. Object Oriented Programming techniques. Lab exercises include designing, writing and debugging programs using commercial IDEs.
Pre/ Co Requisites	Prerequisite CMP 167 Grade of B- or better
Credits	4
Hours	4
Liberal Arts	[x] Yes [ ] No
Course Attribute (e.g. Writing Intensive, WAC, etc)	
General Education Component	__X_ Not Applicable ____ Required ____ English Composition ____ Mathematical and Quantitative Reasoning ____ Science

	____ Flexible ____ World Cultures ____ US Experience in its Diversity ____ Creative Expression ____ Individual and Society ____ Scientific World

3. **To:** Underline the changes

Department(s)	Computer Science
Career	[x] Undergraduate [ ] Graduate
Academic Level	[x] Regular [ ] Compensatory [ ] Developmental [ ] Remedial
Subject Area	Computer Science
Course Prefix & Number	CMP 168
Course Title	Programming Methods II
Description	Continuation of parameter passing with a focus on devising function definitions and tracing recursive calls. Sorting and searching algorithms and a comparison of their performance. GUI programming. Exceptions and Exception Handling. Object Oriented Programming techniques. Designing, developing, and debugging programs using commercial IDEs.
Pre/ Co Requisites	<u>Prerequisites: CMP 157, CMP 167 Grade of B- or better</u> <u>Corequisite: CMP 158</u>
Credits	4
Hours	4
Liberal Arts	[X] Yes [ ] No
Course Attribute (e.g. Writing Intensive, WAC, etc)	
General Education Component	____ Not Applicable __X__ Required ____ English Composition

	____ Mathematics and Quantitative Reasoning __X__ Science _X___ Flexible ____ World Cultures ____ US Experience in its Diversity ____ Creative Expression ____ Individual and Society _X___ Scientific World

4. **<u>Rationale (Explain how this change will impact the learning outcomes of the department and Major/Program)</u>:**

The co-requisite will enable application of the concepts learned and better coverage of the material. This is a 4 hour course for students who intend to major in Computer Science, Mathematics, Computer Graphics and Imaging, or the sciences. Students will be involved in experiential learning assignments and projects.

5. **<u>Date of departmental approval</u>:**

March 01, 2022

DEPARTMENT OF Computer Science

CURRICULUM CHANGE

1. **Type of Change:** Title, Prerequisite, Corequisite

2. **From:** ~~Strikethrough~~ the changes

Department(s)	Computer Science
Career	[x] Undergraduate [ ] Graduate
Academic Level	[x] Regular [ ] Compensatory [ ] Developmental [ ] Remedial
Subject Area	Computer Science
Course Prefix & Number	CMP 232
Course Title	~~Elementary~~ Discrete ~~Structures and Applications to Computer Science~~
Description	Sets, relations, and functions; propositional calculus, Boolean algebras, and combinatorial circuits, counting methods; proof techniques; analysis of algorithms; graphs and trees, puzzles; finite machines, sequential circuits, and recognizers.
Pre/ Co Requisites	~~Mat 172 or departmental placement or permission~~
Credits	4
Hours	4
Liberal Arts	[x] Yes [ ] No
Course Attribute (e.g. Writing Intensive, WAC, etc)	
General Education Component	____ Not Applicable ____ Required ____ English Composition ____ Mathematical and Quantitative Reasoning ____ Science ____ Flexible ____ World Cultures ____ US Experience in its Diversity ____ Creative Expression ____ Individual and Society

	____ Scientific World

3. **<u>To</u>:** <u>Underline</u> the changes

Department(s)	Computer Science
Career	[x] Undergraduate [ ] Graduate
Academic Level	[x] Regular [ ] Compensatory [ ] Developmental [ ] Remedial
Subject Area	Computer Science
Course Prefix & Number	CMP 232
Course Title	Discrete <u>Mathematics</u>
Description	Sets, relations, and functions; propositional calculus, Boolean algebras, and combinatorial circuits, counting methods; proof techniques; analysis of algorithms; graphs and trees, puzzles; finite machines, sequential circuits, and recognizers. <u>Note: Students who earned a grade below B- in MAT 172 are strongly urged to speak to an adviser in the Department of Mathematics and Computer Science prior to registering fort CMP 232</u>
Pre/ Co Requisites	<u>Prerequisite: MAT 172</u> <u>Pre or Co-Requisites: MAT 175, CMP 157, CMP 167</u>
Credits	4
Hours	4
Liberal Arts	[x] Yes [ ] No
Course Attribute (e.g. Writing Intensive, WAC, etc)	
General Education Component	__X__ Not Applicable ____ Required ____ English Composition ____ Mathematics and Quantitative Reasoning ____ Science ____ Flexible ____ World Cultures

	____ US Experience in its Diversity ____ Creative Expression ____ Individual and Society ____ Scientific World

4. **Rationale (Explain how this change will impact the learning outcomes of the department and Major/Program):**

- **a.** Course title was changed to more accurately represent the contents of the course
- **b.** New co-requisites will enable better coverage of the material: knowledge of limits from Mat 175 will allow to better cover the topic of Orders of Growth in CMP 232, while basic knowledge of algorithmic concepts from CMP 167 will enable application of the theoretical material to algorithms without the need to teach the topic from scratch.

5. **Date of departmental approval:** 1/26/2021

LEHMAN COLLEGE
OF THE
CITY UNIVERSITY OF NEW YORK

DEPARTMENT OF Computer Science

CURRICULUM CHANGE

1. **Type of Change:** Title, description

2. **From:** ~~Strikethrough~~ the changes

Department(s)	Computer Science
Career	[x] Undergraduate [ ] Graduate
Academic Level	[x] Regular [ ] Compensatory [ ] Developmental [ ] Remedial
Subject Area	Computer Science
Course Prefix & Number	CMP 338
Course Title	Data Structures ~~and Algorithms I~~
Description	Abstract characterizations of data structures ~~such as arrays, stacks, queues, trees and graphs will be studied along with~~ algorithms ~~that make use of such structures, including algorithms~~ for sorting, searching, and memory management. ~~Implementation issues will be~~

	~~considered, and students will write programs that embody these structures and algorithms.~~
Pre/ Co Requisites	Prerequisites: CMP 232, and ~~CMP 326 or CMP 168~~
Credits	4
Hours	4
Liberal Arts	[x] Yes [ ] No
Course Attribute (e.g. Writing Intensive, WAC, etc)	
General Education Component	__X__ Not Applicable ____ Required ____ English Composition ____ Mathematics ____ Science ____ Flexible ____ World Cultures ____ US Experience in its Diversity ____ Creative Expression ____ Individual and Society ____ Scientific World

3. **To:** Underline the changes

Department(s)	Computer Science
Career	[x] Undergraduate [ ] Graduate
Academic Level	[x] Regular [ ] Compensatory [ ] Developmental [ ] Remedial
Subject Area	Computer Science
Course Prefix & Number	CMP 338
Course Title	Data Structures
Description	Abstract characterizations of data structures, <u>analysis and implementation of</u> algorithms for sorting, searching, and memory management.
Pre/ Co Requisites	Prerequisites: CMP 232, <u>and CMP 157 and CMP 168</u> <u>Pre or Corequisite CMP 269</u>

Credits	4
Hours	4
Liberal Arts	[x] Yes [ ] No
Course Attribute (e.g. Writing Intensive, WAC, etc)	
General Education Component	__X__ Not Applicable ____ Required ____ English Composition ____ Mathematics ____ Science ____ Flexible ____ World Cultures ____ US Experience in its Diversity ____ Creative Expression ____ Individual and Society ____ Scientific World

4. **Rationale (Explain how this change will impact the learning outcomes of the department and Major/Program):**
Course name has been changed to better reflect the contents of the course. The course description has been modified to better represent the course curriculum. A greater emphasis will be given to the data structures and less to the analysis of algorithms in CMP 338.
CMP 326 is the new course code for CMP 168 which explains why either course can be counted as a prerequisite.

5. **Date of departmental approval:** 1/26/2021

LEHMAN COLLEGE
OF THE
CITY UNIVERSITY OF NEW YORK

DEPARTMENT OF Computer Science

CURRICULUM CHANGE

1. **Type of Change:** Credits

2. **From:** ~~Strikethrough~~ the changes

Department(s)	Computer Science
Career	[X] Undergraduate [ ] Graduate
Academic Level	[X] Regular [ ] Compensatory [ ] Developmental [ ] Remedial
Subject Area	Computer Science
Course Prefix & Number	CMP 405
Course Title	Introduction to Networks
Description	Introduction to network protocols and algorithms. Intensive study of the most important protocols at each layer. Examination of their strengths and weaknesses. Basic algorithms for identifying primary servers, constructing forwarding and broadcasting trees, and determining routing tables. Writing a simple networking service at the I.P. layer or higher. Lab exercises include building and testing small networks.
Pre/ Co Requisites	Prerequisite CMP 334 and CMP 338
Credits	~~3 Credits~~
Hours	4 Hours
Liberal Arts	[X] Yes [ ] No
Course Attribute (e.g. Writing Intensive, WAC, etc)	
General Education Component	__X__ Not Applicable ____ Required ____ English Composition ____ Mathematics ___ Science

	____ Flexible
	    ____ World Cultures
	    ____ US Experience in its Diversity
	    ____ Creative Expression
	    ____ Individual and Society
	    ____ Scientific World

3. **<u>To</u>:** <u>Underline</u> the changes

Department(s)	Computer Science
Career	[X] Undergraduate [ ] Graduate
Academic Level	[X] Regular [ ] Compensatory [ ] Developmental [ ] Remedial
Subject Area	Computer Science
Course Prefix & Number	CMP 405
Course Title	Introduction to Networks
Description	Introduction to network protocols and algorithms. Intensive study of the most important protocols at each layer. Examination of their strengths and weaknesses. Basic algorithms for identifying primary servers, constructing forwarding and broadcasting trees, and determining routing tables. Writing a simple networking service at the I.P. layer or higher. Lab exercises include building and testing small networks.
Pre/ Co Requisites	Prerequisites CMP 334 and CMP 338
Credits	<u>4 Credits</u>
Hours	4 Hours
Liberal Arts	[X] Yes [ ] No
Course Attribute (e.g. Writing Intensive, WAC, etc)	

General Education Component	__X__ Not Applicable
	____ Required
	    ____ English Composition
	    ____ Mathematics
	    ____ Science

	____ Flexible ____ World Cultures ____ US Experience in its Diversity ____ Creative Expression ____ Individual and Society ____ Scientific World

4. **Rationale (Explain how this change will impact the learning outcomes of the department and Major/Program):**
The current allotted credits do not reflect the commitment required from the students. This is a 4 hour course for students who intend to major in Computer Science, or the sciences. Students will be involved in experiential learning assignments and projects. The immense changes and vast expansion in technology, networks, data transfer protocols and algorithms used since the inception of the course have forced the expansion of the course's curriculum and thus require an increase in the number of credits associated with the course. Introduction to network protocols and algorithms. Intensive study of the most important protocols at each layer. Examination of their strengths and weaknesses. Basic algorithms for identifying primary servers, constructing forwarding and broadcasting trees, and determining routing tables. Writing a simple networking service at the I.P. layer or higher. Lab exercises include building and testing small networks.

5. **Date of departmental approval:** March 24, 2021

LEHMAN COLLEGE
OF THE
CITY UNIVERSITY OF NEW YORK

DEPARTMENT OF Computer Science

CURRICULUM CHANGE

1. **Type of Change:** Title, description

2. **From:** ~~Strikethrough~~ the changes

Department(s)	Computer Science
Career	[x] Undergraduate [ ] Graduate
Academic Level	[x] Regular [ ] Compensatory [ ] Developmental [ ] Remedial
Subject Area	Computer Science
Course Prefix & Number	CMP 410
Course Title	~~Data Structures and Algorithms II~~
Description	Design and analysis of algorithms~~: Worst and average case behavior. Design strategies, such as divide and conquer, the greedy principle,~~ dynamic programming, and backtracking, ~~are illustrated by examples chosen from sorting and searching, applications of graph theory,~~ scheduling, pattern matching, matrix multiplication, and other topics. NP-complete problems. ~~Parallel processing and algorithms.~~
Pre/ Co Requisites	Prerequisites: CMP 338 and MAT 313
Credits	4
Hours	4
Liberal Arts	[x] Yes [ ] No
Course Attribute (e.g. Writing Intensive, WAC, etc)	
General Education Component	_X__ Not Applicable ____ Required ____ English Composition ____ Mathematics ____ Science

	____ Flexible ____ World Cultures ____ US Experience in its Diversity ____ Creative Expression ____ Individual and Society ____ Scientific World

3. **To:** Underline the changes

Department(s)	Computer Science
Career	[x] Undergraduate [ ] Graduate
Academic Level	[x] Regular [ ] Compensatory [ ] Developmental [ ] Remedial
Subject Area	Computer Science
Course Prefix & Number	CMP 410
Course Title	Design and Analysis of Algorithms
Description	Design strategies and analysis of algorithms: dynamic programming, scheduling, pattern matching, matrix multiplication, and other topics. NP-complete problems.
Pre/ Co Requisites	Prerequisites: CMP 338 and MAT 313
Credits	4
Hours	4
Liberal Arts	[x] Yes [ ] No
Course Attribute (e.g. Writing Intensive, WAC, etc)	
General Education Component	__X_ Not Applicable ____ Required ____ English Composition ____ Mathematics ____ Science ____ Flexible ____ World Cultures ____ US Experience in its Diversity ____ Creative Expression

	____ Individual and Society ____ Scientific World

4. **<u>Rationale (Explain how this change will impact the learning outcomes of the department and Major/Program)</u>:**
Course name has been changed to better reflect the contents of the course. The course description has been modified to better represent the course curriculum. A greater emphasis will be given to the analysis of algorithms in CMP 410 and less to data structures themselves.

5. **<u>Date of departmental approval</u>:** 1/26/2021

LEHMAN COLLEGE
OF THE CITY UNIVERSITY OF NEW YORK

DEPARTMENT OF Computer Science

CURRICULUM CHANGE

1. **Type of Change:** Prerequisite, Corequisite

2. **From:** ~~Strikethrough~~ the changes

Department(s)	Computer Science
Career	[x] Undergraduate [ ] Graduate
Academic Level	[x] Regular [ ] Compensatory [ ] Developmental [ ] Remedial
Subject Area	Computer Science
Course Prefix & Number	CMP 428
Course Title	Video Game Programming
Description	General game architecture, asynchronous input, animated sprites, action oriented A.I., collision detection, scrolling, sound clips, 3D graphics. Student projects involving development of several video games, both individually and in teams. Note: Students should expect to devote a great deal of time working both individually and in teams to produce several video games written in Java. This is a Programming Intensive course.
Pre/ Co Requisites	~~Prerequisite CMP 338~~ ~~Corequisite MAT 226~~
Credits	4
Hours	4
Liberal Arts	[ ] Yes [X] No
Course Attribute (e.g. Writing Intensive, WAC, etc)	
General Education Component	__X__ Not Applicable ____ Required ____ English Composition ____ Mathematical and Quantitative Reasoning ____ Science

	____ Flexible
	    ____ World Cultures
	    ____ US Experience in its Diversity
	    ____ Creative Expression
	    ____ Individual and Society
	    ____ Scientific World

3. **To:** Underline the changes

Department(s)	Computer Science
Career	[x] Undergraduate [ ] Graduate
Academic Level	[x] Regular [ ] Compensatory [ ] Developmental [ ] Remedial
Subject Area	Computer Science
Course Prefix & Number	CMP 428
Course Title	Video Game Programming
Description	General game architecture, asynchronous input, animated sprites, action oriented AI, collision detection, scrolling, sound clips, and 3D graphics. Student projects involving development of several video games, both individually and in teams. Note: Students should expect to devote a great deal of time working both individually and in teams to produce several video games. This is a Programming Intensive course.
Pre/ Co Requisites	<u>Prerequisite CMP 338</u> <u>Prerequisite or Corequisite MAT 313</u>
Credits	4
Hours	4
Liberal Arts	[X] Yes [ ] No
Course Attribute (e.g. Writing Intensive, WAC, etc)	
General Education Component	__X__ Not Applicable ____ Required ____ English Composition

	___ Mathematics and Quantitative Reasoning ____ Science ____ Flexible ____ World Cultures ____ US Experience in its Diversity ____ Creative Expression ____ Individual and Society ____ Scientific World

4. **<u>Rationale (Explain how this change will impact the learning outcomes of the department and Major/Program)</u>:**
MAT 313: Elements of Linear Algebra is now a better corequisite than MAT 226 Vector Calculus due to the nature of the material covered in CMP 428. Since the development of CMP 428 Video Game Programming, the curriculum for both MAT 313 and MAT 226 have changed. MAT 226 is no longer a prerequisite for MAT 313 and MAT 313 contains more topics relevant to the concepts addressed in CMP 428. These topics include matrix/vector operations and linear transformations, which are not discussed in MAT 226.

5. **<u>Date of departmental approval</u>:** March 24, 2021

LEHMAN COLLEGE
OF THE
CITY UNIVERSITY OF NEW YORK

DEPARTMENT OF Computer Science

CURRICULUM CHANGE

1. **Type of Change:** Courses Withdrawn

2. **Description:**
Withdrawn Courses:

CIS 166 - Computer Programming for Information Processing I
CIS 211 - Computer Information Systems
CIS 228 - The Internet
CIS 246 - E-Commerce

3. **Rationale (Explain why this course/program is no longer needed in the Department):**
In the rapidly changing field of Computer Information Systems, the current major is no longer serving our students' needs.

The new major is now divided into core requirements that all CIS majors must complete. These requirements provide our students with the foundation they will need to complete their undergraduate studies and will prepare them for possible graduate studies going forward.

In addition, the major now includes multiple specialization tracks which students can choose from. Students need only complete one track in order to complete the major. However, they may choose to add additional track(s) if they wish.

Lastly, it was discovered that one course (CIS 246) was incorrectly listed on website presenting the Lehman College Course Catalog.

CIS 166 - is very similar to CMP 167. CIS majors will now be required to take CMP 167 instead of CIS-166.

CIS 211 - The content of this course has become obsolete and is not contributing to the students' education.

CIS 228 - The content of this course is very similar to the newly introduced course CMP 128 which will be its replacement. CIS majors will now be required to take CMP 128 as their first course in the CIS Major in an attempt to ease them into programming. CMP 128 will be the prerequisite for CMP 167.

CIS 246 - The course is a duplicate of the correct course CIS 346 on the course catalog.

4. **Date of departmental approval:** March 24, 2021

Appendix I: Who has taught CS through the years?

9 CS Faculty Currently:

1 Distinguished
2 Associates
3 Assistants
3 Lecturers

Distinguished:
Victor Pan (1988-Present)

Associate:
Brian Murphy (1998-Present)
Mathew Johnson (2012-Present)

Assistant:
Itai Feigenbaum (2016-Present)
Liang Zhou (2017-Present)
Mingxian Zoung (2018-Present)

Lecturer:
Sameh Fakhouri (2011-Present) – retiring this year
Eva Sofianos (2018-Present)
Steven Fulenkeza (2018-Present)

Note: It has been five (5) years without a search for new CS faculty.

14+4 CS Faculty 2006:

1 Distinguished
6 Fulls
4 Associates
1 Assistant
2 Lecturers
+4 Fulls

Distinguished:
Victor Pan (1988-Present)

Full:
Esther Phillips (1983-2006)
Charles Berger (1983-2012)
Melvyn Fitting (1983-2012)
Robert Schneider (1983-2018)
Katherine St. John (1999-2017)
Nancy Grifeth (2003-2015)
+ Joseph Lewittes (1983-2007)
+ Keen (1983-2015)
+ Feinerman (1983-2016)
+ Handel (1983-2017)

Associate:
Julian Laderman (1983-2007)
Gwang Jung (1999-Present)
Boris Yamrom (2002-2006)
Yves Jean (2004-2011)

Assistant:
Rhys Rosholt (1998-2013)

Lecturer:
Brian Murphy (1998-Present)
Anthony Cocchi (2002-2015)

11+3 CS Faculty 2008:

1 Distinguished
5 Fulls
2 Associates
1 Assistants
2 Lecturers
+3 Fulls

Distinguished:
Victor Pan (1988-Present)

Full:
Charles Berger (1983-2012)
Melvyn Fitting (1983-2012)
Robert Schneider (1983-2018)
Katherine St. John (1999-2017)
Nancy Grifeth (2003-2015)
+ Keen (1983-2015)
+ Feinerman (1983-2016)
+ Handel (1983-2017)

Associate:
Gwang Jung (1999-Present)
Yves Jean (2004-2011)

Assistant:
Brian Murphy (1998-Present)

Lecturer:
Rhys Rosholt (1998-2013)
Anthony Cocchi (2002-2015)

<u>12+3 CS Faculty 2011</u>

1 Distinguished
5 Fulls
2 Associates
1 Assistant
3 Lecturers
+3 Fulls

Distinguished:
Victor Pan (1988-Present)

Full:
Charles Berger (1983-2012)
Melvyn Fitting (1983-2012)
Robert Schneider (1983-2018)
Katherine St. John (1999-2017)
Nancy Grifeth (2003-2015)
+ Keen (1983-2015)
+ Feinerman (1983-2016)
+ Handel (1983-2017)

Associate:
Gwang Jung (1999-Present)
Yves Jean (2004-2011)

Assistant:
Brian Murphy (1998-Present)

Lecturer:
Rhys Rosholt (1998-2013)
Anthony Cocchi (2002-2015)
Sameh Fakhouri (2011-Present)

<u>13+3 CS Faculty 2012</u>

1 Distinguished
5 Fulls
1 Associate
3 Assistants
3 Lecturers
+3 Fulls

Distinguished:
Victor Pan (1988-Present)

Full:
Charles Berger (1983-2012)
Melvyn Fitting (1983-2012)
Robert Schneider (1983-2018)
Katherine St. John (1999-2017)
Nancy Grifeth (2003-2015)
+ Keen (1983-2015)
+ Feinerman (1983-2016)
+ Handel (1983-2017)

Associate:
Gwang Jung (1999-Present)

Assistant:
Brian Murphy (1998-Present)
Mathew Johnson (2012-Present)
Megan Owen (2012-2017)

Lecturer:
Rhys Rosholt (1998-2013)
Anthony Cocchi (2002-2015)
Sameh Fakhouri (2011-Present)

11+2 CS Faculty 2013

1 Distinguished
3 Fulls
1 Associate
3 Assistants
3 Lecturers
+2 Fulls

Distinguished:
Victor Pan (1988-Present)

Full:
Robert Schneider (1983-2018)
Katherine St. John (1999-2017)
Nancy Grifeth (2003-2015)
+ Feinerman (1983-2016)
+ Handel (1983-2017)

Associate:
Gwang Jung (1999-Present)

Assistant:
Brian Murphy (1998-Present)
Mathew Johnson (2012-Present)
Megan Owen (2012-2017)

Lecturer:
Rhys Rosholt (1998-2013)
Anthony Cocchi (2002-2015)
Sameh Fakhouri (2011-Present)

CS Faculty All-time

Distinguished:
Victor Pan (1988-Present)

Full:
* Richard Mozak (1983-199x) * - Not represented above
* Connor Lazarov (1983-2001) * - Not represented above
Esther Phillips (1983-2006)
Charles Berger (1983-2012)
Melvyn Fitting (1983-2012)
Robert Schneider (1983-2018)
Katherine St. John (1999-2017)
Nancy Grifeth (2003-2015)
+ Joseph Lewittes (1983-2007) + - often taught for CS
+ Keen (1983-2015) + - often taught for CS
+ Feinerman (1983-2016) + - often taught for CS
+ Handel (1983-2017) + - often taught for CS

Associate:
Julian Laderman (1983-2007)
Brian Murphy (1998-Present)
Gwang Jung (1999-Present)
Boris Yamrom (2002-2006)
Yves Jean (2004-2011)

Assistant:
Mathew Johnson (2012-Present)
Megan Owen (2012-2017)
* Javier Lopez (2014-2015) * - Not represented above
Itai Feigenbaum (2016-Present)
Liang Zhou (2017-Present)
Mingxian Zoung (2018-Present)

Lecturer:
* David Bethelmey (199x-1998) * - Not represented above
Rhys Rosholt (1998-2013)
Anthony Cocchi (2002-2015)
Sameh Fakhouri (2011-Present)
Eva Sofianos (2018-Present)
Steven Fulenkeza (2018-Present)