Olympiads in Informatics, 2007, Vol. 1, 37–49

Contests in Programming: Quarter Century of Lithuanian Experience

Valentina DAGIENĖ, Jūratė SKŪPIENĖ

Institute of Mathematics and Informatics
Akademijos str. 4, Vilnius, 08663 Lithuania
e-mail: dagiene@ktl.mii.lt, jurate@ktl.mii.lt

Abstract. The paper deals with development of informatics competitions in Lithuanian and in particular of the Lithuanian Olympiads in Informatics over the past 25 years. The role of the Young Programmer's School in both introducing programming to secondary school students, organizing programming competitions and later encouraging and training students to participate in informatics olympiads is emphasized. The evolution of the national olympiad in informatics from the first pen and paper competitions to the current four round contest, held using contest and grading systems, is described. The paper also gives a short overview of other related contests like the Baltic Olympiads in Informatics, and the Beaver contest.

Key words: teaching programming, informatics, algorithms, contests, informatics olympiads.

1. The Early Years

The first talks about teaching programming at schools started in the beginning of the seventies. In those times some large electronic computers were available in Lithuania, however not for teaching. Only a few educators thought that it would be possible to involve kids in the world of computers and programming. Some researchers from the Institute of Mathematics and Informatics supported this idea and worked on it to become a reality. The first few after-school classes in programming were established in several Lithuanian secondary schools in 1970–1975. There also appeared the first ideas to establish a school to teach programming secondary school students (Dagienė, 2006).

A significant role in designing a methodology for teaching programming was played by the scientist of the Institute of Mathematics and Informatics. In 1979 comprehensive materials for teaching programming were prepared, including tasks, texts, and answers. The Ministry of Education of Republic of Lithuania agreed to try this method in ten schools. Scientists from the Institute created a Pascal translator for the electronic computing machines of the unified system (Baliūnaitė, 1979). The Pascal translator was designed specifically for the learning process, putting emphasis on debugging program source so the user (the student) could receive comprehensive output after trying to compile and execute their first program. The translator even corrected some mistakes and afterwards reported what was changed and how.

The Young Programmers' School by Correspondence (JPM – *Jaunųjų programuotojų mokykla*) was founded in 1981 by the Institute of Mathematics and Informatics (Dagys, 1994; Dagys, 2006). The curricula consisted of teaching main concepts of procedural programming and basic algorithms using Pascal. The learning material and assignments used to be published in the national youth daily newspaper *Komjaunimo tiesa* (cur. *Lietuvos Rytas*) twice a month. The students had to sent their algorithms which were evaluated and the solutions were published afterwards. The best students were invited to summer camps where they not only had a chance to see and touch a computer, but also to compete. The students had to solve algorithmic tasks while designed algorithms had to be written on a piece of paper in Pascal.

In 1985 it was decided to organize the first contest of young programmers. Anyone aged below 30 could participate in it. The contest took part in two rounds. Tasks of the first round were published in the daily newspaper *Komjaunimo tiesa*, in the journals *Informatika* and *Mokslas ir technika* (*Science and Technology*). The tasks even used to be announced through the national television channel. In the first round there were five tasks to solve using one of the three allowed programming languages: Pascal, Fortran, PL/1. Later Fortran and PL/1 languages were removed from the list and C added. The participants had one month to solve the tasks and send the algorithms together with justification to the scientific committee. Sources of the programs could have been written by hand, printed with typewriter or printer.

The Project of Distance Teaching of Informatics using Electronic Mail got support from UNESCO in 1993 (Dagys, 1993). Teaching via electronic mail was mainly performed indirectly through programming contests. The project also took advantage of transmission of source code using media which was quite new and educative in those times. The effect of program execution were observed and evaluated through distance.

In 1996–1997 *The School Of Teaching Algorithms Via El. Mail In Baltic Countries* was established (Dagiene, 1997). Over 100 students from Estonia, Latvia and Lithuania took part in the school. There were two learning and one contest sessions.

2. Young Programmers' School – Search for Talent

The core curricula of the *Young Programmers' School* is teaching algorithms. Programming languages as well as the computer are considered to be learning tools and only the basic information regarding those tools (as much as it is needed to write down and execute the algorithm) is provided. Using Pascal as a working language minimizes the time cost to learn basic constructions of programming languages.

All the teaching materials of the Young Programmers' School consisted of several chapters: 1) identifiers, variables, constants, assignment statement and sequence of statements, 2) conditional statements, 3) repetitions of actions, 4) programs and their execution by a computer, 5) logical values, 6) functions and procedures, 7) recursion, 8) discrete data types, 9) real numbers and records, 10) arrays, 11) programming style, 12) program design. We would like to emphasize again that all the theory is taught only through program comprehension (e.g., given fragments of a program which has to be completed, corrected, etc.) and program design assignments.

There exists a steady attitude in the Young Programmers' School that a student has not only to become acquainted with the basic constructions of programming but also has to learn how to justify and describe an algorithm, and design clear and simple solution.

For most children, theory is less attractive than practical activities. Thus the basic principles of the theory were delivered in an indirect way through problem solving. The set of programming problems was chosen in accordance with the requirements dictated by theory and good programming style (Grigas, 1990).

From the start of the Young Programmers' School until now there have been many changes in the teaching of informatics in general and in programming in particular, due to the increase in the number of computers in educational institutions and the introduction of informatics as a compulsory discipline in secondary schools. The changes in the structure of the Young Programmers' School may be characterized by five periods: 1) Universal (general) programming teaching (1981–1986), 2) Learning effectively: differentiation by students' abilities (1986–1993), 3) Intensive teaching of gifted students (1993–1999), 4) Preparing students for the olympiads (since 1999), and 5) Using new media while learning algorithms (since 2005).

The first Informatics Olympiad for the enrolled students of the Young Programmers' School took place in summer 1982. The tasks were designed with extreme accuracy and forethought. They were attractive and challenging (Dagienė, 1991). Programming olympiads used to take place every year but only for the students studying in Young Programmers' School. Therefore the Young Programmer's School was an impulse to start the *Lithuanian Olympiad in Informatics*.

3. Evolution of the Lithuanian Olympiad in Informatics

The first Lithuanian nation-wide Olympiad in Informatics was organized in 1990, i.e., the year after the first International Olympiad in Informatics (IOI).

In the beginning the olympiad consisted of the three rounds: 1) school round 2) regional round (about 60 regions), and 3) national round. Since 1993 the national round has been split into two parts. The first part was organized using e-mail, the second was on-site competition. The structure of four rounds is convenient and has been kept until now.

The final stage of the national olympiad is organized in a different region each year (Fig. 1). About 50 participants from all over Lithuania are invited. Organizing the event in different regions not only allows the contestants to get to know the region but also gives a possibility to the teachers of local schools to look at the olympiad from inside – to observe how the final versions of tasks are being prepared, and to look closer at the competition system and grading.

3.1. *Using E-mail for On-Line Contests*

The organization part of the national round of the first national olympiad was quite complicated. Each of the sixty counties in Lithuania designation winners of their regional

Fig. 1. Regions that hosted finals of the national olympiad in Informatics.

competition for the national round. As it was not possible to arrange an on-site competition for about two hundred students, the first part of the national round used to be arranged in several selected municipalities. The organizers of the olympiad would send their representative with tasks to each municipality, the representative would observe and coordinate the competition and bring back solutions to the scientific committee for evaluation.

A significant breakthrough became possible in 1993, when the Fidonet computer network became available for some schools in each region. It was decided to organize the first part of the national round in each region using e-mail. Then it was both a brave and innovative decision. On the one hand even though e-mail was available at schools many teachers still did not know how to use it nor that there was a real need for it. Organizing a competition in such a way stimulated teachers to learn how to use e-mail.

However in the first years there was not an easy job for organizers of the olympiad to manage the contest. Some e-mails would not reach the contest organizers and the organizers had to call the region and find out what happened, there were lots of problems with attaching solutions. Teachers either did not know how to attach a file or the attachment received was un-decodable and the organizers spent a lot of time consulting teachers how to do it. A lot of problems were caused due to ignorance of file naming instructions. Many solution files would be given random names and contained no contestant name in the comments.

It took several years until the teachers got accustomed to using e-mail and the transfer of solutions to the scientific committee became fluent. Olympiads had positive educational effect also on promoting the use of e-mail in Lithuanian schools.

Solutions were delivered through e-mail and afterwards graded using black-box testing for the ten years from 1993 till 2002.

3.2. *Contest Management and Grading Systems*

In order to test solutions automatically programs have to comply with certain input/output formatting requirements. In several IOIs there were severe problems regarding automated grading when programs with typographic errors in the data file name or those leaving extra space at the end of line were not given any points. Similar problems were also encountered in Lithuanian olympiads.

However in order to motivate students the scientific committee tried to be more objective and to make distinctions between formatting errors and more serious mistakes. This was a huge load of work for the scientific committee. Nearly every program which did not get full score had to be checked manually searching for formatting errors.

This was especially important for younger students as for many of them this was their first competition and getting zero points for a program with a good algorithm and a typographic error might have resulted in a decision to quit the olympiad. Later the grading became more strict, especially in the senior division. On the one hand it was decided that part of this workload could be done by the contestants themselves (i.e., they should analyze their solution and write an appeal), on the other hand in some cases it was hard to distinguish between formatting and non-formatting errors and it was decided not to change the code of the contestant at all.

The first Contest and Grading System that allowed the submission of programs via a web-interface during contest time, and to check whether they compile and comply with format requirements, was designed and used in Finland in 2001. Such a system has been used in Lithuanian contests since 2003.

In the IOI the use of a Contest and Grading system during the competition was accepted with support from the contestants. However this was not so in Lithuania. The top contestants who had participated in IOIs supported the use of a Contest system in Lithuanian olympiads. However many inexperienced (especially younger) contestants were shocked when they tried the system for the first time. Typically they tried to submit a program which produced correct output on their computer and when the contest system rejected the program they were lost and did not even know how to try to debug it as their debugging skills were quite low. Some of them could not even understand what happened, as they could not believe that a program which works on one computer might have failed on another. Even if the problem was a typographic error in the file name for some students it was impossible to find it out. Again there started coming a huge numbers of e-mails to the technical committee asking for help and complaining that the system did not work correctly. The Contest and Grading system has been used in Lithuania for five years and many teachers have got accustomed to using it. However still in some schools some people find it difficult to use and can not understand that error message during the contest is actually a hint that they should use (Skūpienė, 2004).

3.3. *Participants*

All students in secondary education under the age of 20 are invited to compete in Lithuanian Olympiads in Informatics. Approximately 3000 students take part in national competition each year.

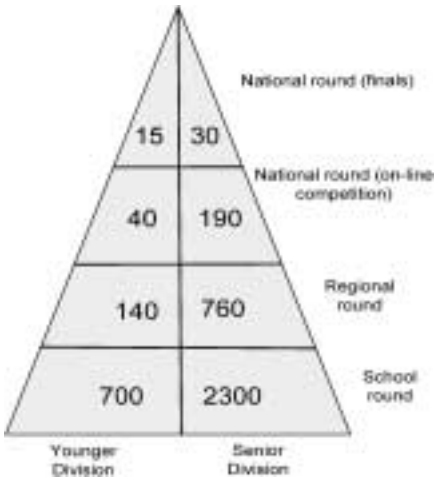

Fig. 2. Fig. 2. Average number of participants in Lithuanian Olympiads in Informatics.

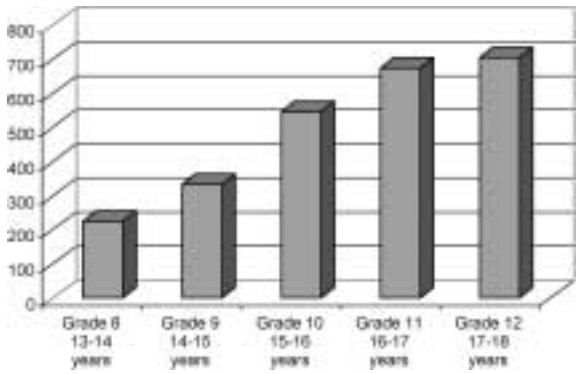

Fig. 3. Distribution of participants in the first round of LOI'2006 according to grades.

The number of younger (grades 8–9) participants has significantly increased since 1997 when a separate division for younger students was established and 30% of the places in the finals of the national competition were reserved for students from younger division. This motivated both younger students and their teachers.

4. Tasks – Keystone of Contests and Resources for Problem Solving

Interest and engagement is very important in competitions as well as in teaching problem solving (e.g., in assignments in the Young Programmers' School) and it essentially depends on problems. Therefore the problems have to be designed taking into account different aspects of each problem, i.e., its educational power and how to interpret its attractiveness to the students (i.e., whether it stimulates learning or not). Attraction,

invention, discoveries are the desired features of a good problem set. Here are some other desired characteristics of problems:

- interesting and attractive formulation;
- algorithmic solution lies behind;
- variety in difficulty;
- do not require specific, especially technical knowledge;
- short, elegant formulation.

Therefore, one should try to present problems from various spheres of science and life, with a lot of real data. Processing large amounts of data becomes challenging and important aspect when learning programming.

However, many the textbooks and teaching materials do not contain actual problems but just small exercises. They are mainly oriented towards checking syntax of a particular programming language. The selection of tasks at the distance education school is very important: they must cover as many theoretical problems as possible, teach students algorithms and programming methods, and what is most important, to acquire the skills of using them (Dagienė, 1993; Dagienė, 1999; Mayer, 1990).

While developing the methodology of teaching algorithms for the School of Young Programmers, we have raised the principle that it is highly important to classify the problems into the sets of problems that would actualize the purposes of teaching algorithms. Two large groups of problems were distinguished: 1) reading problems (for analysis); and 2) writing or design problems (Grigas, 1989).

When someone starts learning programming, active and passive learning methods can be distinguished. This deals with fixing priorities: whether they are taught just programming language constructions or problem solving using programming languages. An active teaching is when problems are solved while the languages constructions are mastered gradually when they are needed in the solution. Therefore it is highly important that the tasks of the first rounds in the national olympiads take this into account, i.e., we would seriously consider which language constructions are needed to solve the problem, so that the beginners would be able to solve the problems.

A variety of task books with problems and their solutions have been prepared and published.

(a) (b) (c) (d) (e)

Fig. 4. Books in problem solving for olympiads in informatics: (a) *A Hundred Tasks in Programming*; (b, c) Programming tasks; (d, e) *Olympiads in Informatics* (volume 1 and 2).

5. Achievements of Lithuanian Students in IOI

The first Lithuanian contestant Andrius Čepaitis participated in IOI in 1989 where he was awarded a gold medal. At that time Lithuania was still part of Soviet Union and Andrius was included into the team of Soviet Union. Lithuania joined the IOI in 1992 soon after restoring independence. Since then each year Lithuania selects sends a team of four contestants to IOI and the Lithuanian contestants were awarded 44 medals in total.

Team selection for the IOI is conducted in the following way. The best six contestants from the national competition senior division are invited to take part in Baltic Olympiad in Informatics which usually takes place in April or May. In cases where it is not possible to do a fair selection of the BOI team due to a small difference in points, extra competitors take part in the BOI on-line, under surveillance of members of NOI. The team of four contestants to represent Lithuania at the IOI is selected taking into account points gained in finals of national competition, points gained in BOI, medals in International Olympiads in previous years and age (younger contestants have priority).

6. Other Contests in Informatics

The national olympiad in informatics is the main however not the only contest for Lithuanian students interested in algorithms and problem solving.

Table 1
Achievements of Lithuanian students of in IOI's

	Gold	Silver	Bronze
IOI 1989	1		
IOI 1992			3
IOI 1993			3
IOI 1994		2	2
IOI 1995	1		3
IOI 1996	1	2	
IOI 1997			1
IOI 1998		1	2
IOI 1999		1	2
IOI 2000		2	1
IOI 2001		1	2
IOI 2002		1	2
IOI 2003		1	1
IOI 2004		1	2
IOI 2005		1	1
IOI 2006		2	1
Total	3	15	26

In 1997 scientists from Kaunas Technology University together with an American Lithuanian Dr. J.P. Kazickas, whose fund sponsors the events, established the *Dr. J.P. Kazickas Regional Competition of Students Computerists*. Ethnically Lithuania is divided into four regions (Aukštaitija, Žemaitija, Dzūkija ir Suvalkija) and the contest takes place in each region once a year. The winners of regional contests are invited to the final round which take place in Kaunas Technology University. The winners of the competitions are given the priority against other candidates if they want to study in the University (Kazicko forumas, 2007).

In order to bring programming and concepts of computer science to younger students, Logo was chosen to be introduced into Lithuanian schools many years ago. As it is known that competition makes learning more attractive, in 2000 it was decided to establish *Logo Competition-Olympiad*. Students in both primary and secondary education are invited to compete. The competition consists of several separate parts and the contestants decide in which particular part to participate. There are contests of pictures, uncontrolled animation, controlled animation, as well as algorithms. Thus younger students become involved in studying algorithms at quite a young age, even before they learn procedural programming language.

A Saturday school of participants of informatics olympiads was established in 2003. The school has three levels. Young students who have no experience in programming, but proved themselves to be good in math competitions, can study at the introductory level. The curricula of the first level covers introductory to programming course as well as some very basic algorithms (e.g., sorting). At the end of the first level (which lasts for one year) the students take a test and those who pass can study at the intermediate level. Also the students who successfully participated in the national informatics olympiad in younger division can study in this level.

During the intermediate level the course covers the main algorithms and methods used to solve tasks for informatics olympiads (e.g., Dijkstra algorithm, dynamic programming, etc.).

At the advanced level the students who passed the final test of the intermediate level as well as participants of finals of national informatics olympiad can study. At the advanced level the students solve complicated and advanced algorithmic tasks and it is possible to study at this level for several years (i.e., until they graduate school). Currently about 50 students from all over Lithuania study in the school.

In 2004 Kaunas Technology University Gymnasium together with charity and support fund of M. Rostropovich initiated the project *National Student Academy* for gifted students. Students having high achievements in various areas (mathematics, chemistry, biochemistry, physics, informatics, economics, writing and music) can enrol at the Academy. During the year they are working on assignments from their subject. There are two or three sessions throughout the year which last from one to two weeks. Scientists are invited to give lectures to the Academy students as well. The exceptional feature of this project is that students from different areas of interest join together to work, create and learn.

In order to achieve high results in informatics olympiads, one has to study and practice for several years, and not so many students are able to do it. We felt that there was a

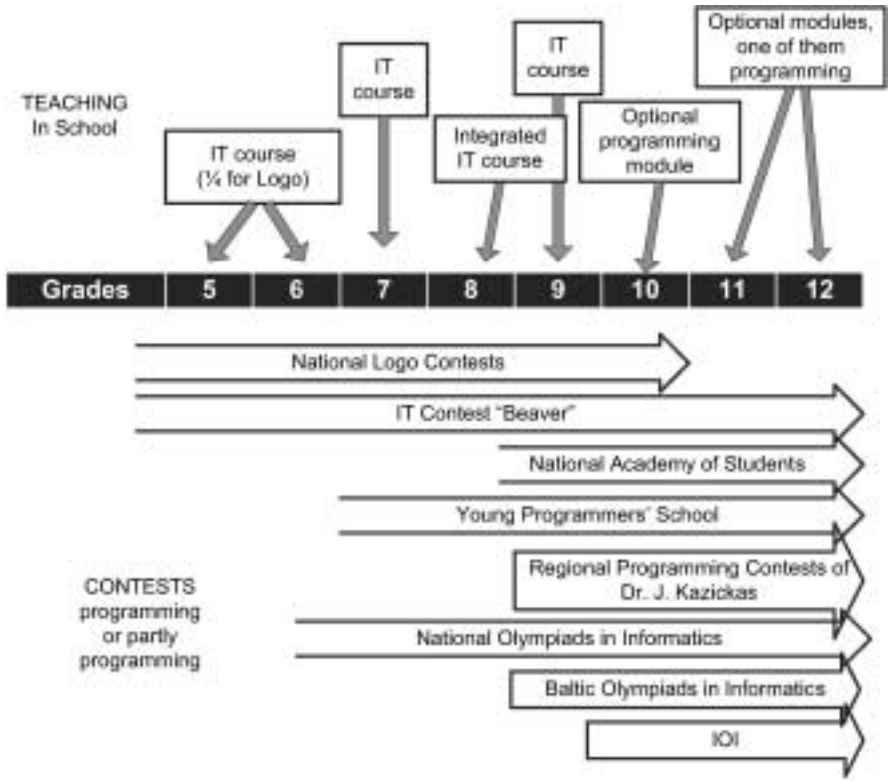

Fig. 5. Teaching Information Technology (IT) and programming in schoolls and various of contests.

lack of a competition in informatics where everyone could participate and have fun. Such a competition existed in mathematics – "Kangaroo". In 2004 it was decided to arrange a similar competition in informatics. The history of "Beaver" began on September 25, 2004, when an experimental trial, in which 779 school students participated, was held. The aim was to check selected technologies of the contest and to evaluate the level of complexity of the presented problems. After a month, on October 21, the first Lithuanian "Beaver" contest took place. As many as 3470 pupils from 146 schools participated. In 2006, the International "Beaver" organizing committee was established which included representatives of seven countries. Tasks are designed for three different age groups (11–14 years, 15–16 years, 17–18 years) and they are multiple-choice tasks. Grading is automated. Average time for solving one task is 2–3 minutes, so task descriptions are short in form. There is still no defined syllabus for the contest, but the discussed topics are: a) general logic; b) ICT in everyday life; c) practical and technical issues; d) information comprehension; e) algorithms and programming; f) mathematics and underlying CS; and g) history and trivia (Opmanis, 2006).

In order to ensure better preparation for the IOI and to strengthen regional relations, various regional olympiads are being organized. Baltic Olympiads in Informatics (BOI) were established by the initiative of the three Baltic countries (Estonia, Latvia, and

Lithuania) in 1995. Year by year six other Baltic countries (Denmark, Finland, Germany, Poland, Sweden and Norway) joined the BOI. Compared to the IOI, BOI is a short-term (the duration is 5 days) and inexpensive event. It can be distinguished by a cosy and good neighbourly atmosphere.

Even though the BOI is a mini-model of IOI it differs significantly. The organization of the scientific part of the BOIs is based on mutual trust of participating countries. The leaders of all the participating countries take part in proposing and selecting problems for the coming BOI. After draft problem formulations are presented, the problems are discussed via e-mail and the each country votes for the problem set for the competition. Most of the problems are translated into native languages by the leaders before leaving to BOI. During the competition leaders are involved in solving various problems which might occur, for example, some misrepresentation in the formulations of contest problems. This a unique possibility for country representatives to gain experience in organizing scientific part of a small international olympiad (Dagienė, 2004).

The BOI is also a form of learning for its participants. The organizers of BOIs try to follow as close as possible the newest IOI trends in problem types, compilers, platforms, contest systems. It is not always possible to do that in national contests. Besides, the competition tasks are always proposed by different countries. Even though all the tasks are of an algorithmic nature they represent cultural and methodical differences. Many students come to the BOI to gain international experience after participating in domestic contest. The BOI can be considered as a pre-arranged international form of learning.

7. Conclusions

Lithuania has long traditions of both teaching programming and algorithms in secondary schools and organizing informatics contests. We have noticed three basic challenges:

1) initiating students to start to learn programming and to do it in attractive and proper way;
2) when students learn the basics of programming they start trying to find an event where they could demonstrate their skills – contests and olympiads are the right place to do this;
3) there are many students who are interested in problem solving and would be interested in participating in the contests but they need instructors or some kind of schools to help them to grow their skills.

We try to investigate and to work in those three directions. In primary education, we introduce Logo: kids aged 11–12 years have chance to develop their own procedures using very simple programming statements (mainly primitive commands, loops and procedures). In lower and upper secondary school students have opportunity to choose optional modules of programming (each are for 70 hours). While learning programming students have possibilities to participate in various contests: Logo Olympiad, IT contest “Beaver”, Lithuanian Olympiad in Informatics, etc.

Consequently both teaching of programming (it can be optional, but it should be available) and contests complement and stimulate each other. At the same time variety in

contests give more possibilities for interested and talented students and allows them to choose the contests which are most interesting and challenging for them.

References

Baliūnaitė, A., V. Dagienė, G. Grigas (1979). Transliator jazyka PASCAL v operacionoj sisteme DOS/ES i jevo ispolzovanyje dlia učebnych celej. *Operativno-informacionyj material*. Novosibirsk: SO AN SSSR (in Russian).

Bulotaitė, J., K. Diks, M. Opmanis, R. Prank (1997). *Baltic Olympiads in Informatics*, Inst. of Math. and Inf., Vilnius, Lithuania.

Dagienė, V. (1991). *Lietuvos jaunųjų programuotojų olimpiados*. Kaunas, Šviesa.

Dagienė, V. (1997). Learning via electronic mail: what and how? *Education for the 21st Century*, 2–4 December, Cape Town, 1–10.

Dagiene, V. (1999). Programming-based solution of problems in informatics curricula. *Communications and Networking in Education: Learning in a Networked Society*, IFIP WG 3.1 and 3.5 (with 3.6). Aulanko, Hämeenlinna, Finland, June 13–18, 88–94.

Dagienė, V. (2006). *The Road of Informatics*. Vilnius, TEV.

Dagienė V., G. Grigas (1993). Development of problem solving skills and creativity through distance teaching of programming. In G. Davies and B. Samway (Eds.), *Teleteaching: IFIP Transactions (A–29)*. Elsevier, Sc. Pub., pp. 179–182.

Dagiene, V., J. Skūpiene (2004). Learning by competitions: olympiads in informatics as a tool for training high grade skills in programming. In T. Boyle, P. Oriogun, A. Pakstas (Eds.), *2nd International Conference Information Technology: Research and Education*. London, pp. 79–83.

Dagys, V. (1994). The work principles of Lithuanian Young Programmers School by correspondence. Human Resources, Human Potential, Human Development: the Role of Distance Education. In *Proceedings of the European Distance Education Network (EDEN) Conference*. Tallinn, 182–184.

Dagys, V., V. Dagienė, G. Grigas (2006). Teaching Algorithms and programming by Distance: Quarter Century's Activity in Lithuania In V. Dagienė, R. Mittermeir (Eds.), *Proc. of the 2nd Int. Conference "Informatics in Secondary Schools: Evolution and Perspectives"*, Vilnius, 402–412.

Dagys, V., A. Klupšaitė (1993). Distance teaching of programming and possibilities of e-mail. *Informatica*, **4**(3–4), 303–311.

Grigas, G. (1990). Some aspects of teaching the art of programming by correspondence. *Informatica*, **1**(1), 156–166.

Grigas, G. (1989). Informatics and creative Ttinking. *Third International Conf. "Children in the Information Age"*. Sofia, pp. 229–240.

Dr. J. Kazicko moksleivių kompiuterininkų forumas (accessed 2007.06.20). http://pilis.if.ktu.lt/˜forumas/

Mayer, R.E. (1990). Teaching for transfer of problem-solving skills to computer programming. Computer-based learning environments and problem solving. In E. de Corte etc. (Eds), *ATO ASI Seties*, Springer-Verlag, 193–206.

Opmanis, M., V. Dagienė, A. Truu (2006). Task types at "Beaver" Contests. *Infomatics Education*, the Bridge between Using and Understanding Computers. *Proceedings of International Conference in Informatics in Secondary Schools – Informatikon Technologies at School*, November 7–11, 2006, Vilnius, TEV, 509–519.

Skūpienė, J. (2004). Automatinis testavimas informatikos olimpiadose. *Informacinės technologijos 2004*, Konferencijos pranešimų medžiaga, Kaunas, Technologija, p. 37–41.

V. Dagienė is a head of Informatics Methodology Department at the Institute of Mathematics and Informatics, also Professor at Vilnius University. She published over 100 scientific papers, wrote more than 50 textbooks in informatics for secondary education. She coordinates the Young Programmer's School, has position of the Chair of The National Olympiads in Informatics, established IT contests "Beaver". V. Dagienė is the national representative of the IFIP for Education (TC3), member of the European Logo Scientific Committee, an elected member of the IOI Scientific Committee. She is an Executive Editor of international journal "Informatics in Education".

J. Skūpienė is younger research fellow in Informatics Methodology Department in the Institute of Mathematics & Informatics. She has published about 10 scientific papers. She is a member of the Scientific Committee of National Olympiads in Informatics since 1994 and a team leader in IOI since 1996. For a few years she was director of studies of Young Programmers' School, since 2004 she has been a coordinator of Informatics section in the National Academy of Students.

She is author/co-author of four books on algorithms and algorithmic problems.