1. Optimal BST's diff from text

2. Kruskal's algo, MST DFS, cycle detection, unique MST

3. Task scheduling

→ (s_i, f_i), nonconflicting
$f_i \le s_j$ or $f_j \le s_i$

⇒ Minimize # of machines

→ Order by s_i

4. Activity selection

→ (s_i, f_i), compatible

→ Maximum size subset of mutually compatible activities

$f_1 \le f_2 \le f_3 \quad \le f_n$

i	1	2	3	4	5	6	7	8	9	10	11
s_i	1	3	0	5	3	5	6	8	8	2	12
f_i	4	5	6	7	9	9	10	11	12	14	16

$\{a_1, a_4, a_8, a_{11}\}$ or $\{a_2, a_4, a_9, a_{11}\}$
are both ~~largest~~ optimal

→ optimal substructure

S_{ij} = { activities that start after a_i finishes and end before a_j starts }

We want to find a maximum set of mutually A_{ij} compatible activities in S_{ij}. Let $a_k \in S_{ij}$.

$A_{ik} = A_{ij} \cap S_{ik}$ $\quad A_{kj} = A_{ij} \cap S_{kj}$

$|A_{ij}| = |A_{ik}| + |A_{kj}| + 1$

Let $c[i,j]$ = optimal soln to S_{ij}

$$c[i,j] = \begin{cases} 0, & \text{if } S_{ij} = \emptyset \\ \max\limits_{a_k \in S_{ij}} \{ c[i,k] + c[k,j] + 1 \}, & \text{if } S_{ij} \neq \emptyset \end{cases}$$

Greedy choice

→ Choose activity with earliest finish time

Let $S_k = \{ a_i \in S : s_i \geq f_k \}$

S_1 is the only subproblem to solve if a_1 is to be included.

But is a_1 necessarily in the optimal solution?

Thm: a_1 is included in any maximum-size subset of mutually compatible activities.

Proof: Exchange argument

Elements of greedy strategy

→ Optimal substructure

→ Greedy choice

→ Safe to make greedy choice

→ Greedy versus dynamic programming