Assignment 2: Due now !

Activity Scheduling/Selection

- Input: A set of activities $\{A_1, \ldots, A_n\}$
- A_i: starting time s_i and finishing time f_i.

$$0 \leq s_i < f_i$$

- A single resource
- Output: As many activities as possible

Example: schedule as many lectures as possible in 1 room.

Maximize *the number of lectures*, as opposed to *the total time* the room is used !

Idea Make the resource *free as soon as possible.*

- Sort the activities in incresing *finishing time*
- Go through the list, take activities if no conflict.

Sort activities so that

$$f_1 \le f_2 \le \ldots \le f_n$$

$S \leftarrow \varnothing$ % output set

$t \leftarrow 0$ % last finishing time, for checking conflict

For $i = 1$ to n do

    If $t \le s_i$ then

        $S \leftarrow S \cup \{A_i\}$

        $t \leftarrow f_i$

    End If

End For

Output S

Proof of correctness

Promising partial solution: After iteration i, the partial solution S_i is promising if for some optimal solution OPT:

$$S_i \subseteq OPT \subseteq S_i \cup \{i+1, \ldots, n\}$$

Show that S_i is promising for all $i \leq n$.

Then S_n must be an optimal solution.

Induction step

Assume that S_i is promising, i.e.,

$$S_i \subseteq OPT \subseteq S_i \cup \{i+1, \ldots, n\}$$

for some optimal solution OPT.

Show that S_{i+1} is promising, i.e., for some optimal solution OPT':

$$S_{i+1} \subseteq OPT' \subseteq S_{i+1} \cup \{i+2, \ldots, n\}$$

IDEA:

- Look at how S_{i+1} is obtained from S_i.
- Each case tells us how to obtain OPT' from OPT.

Most interesting case: $i + 1 \in S_{i+1}$ and $i + 1 \notin OPT$.

Then: OPT contains request j that overlaps with $i + 1$.

- j must finish later than $i + 1$,
- j is the *only* request in OPT that overlap with $i + 1$

so we can swap j and $i + 1$ to obtain OPT'.

Minimum Spanning Tree Problem

Input: A weighted, undirected graph G (weights on the edges)

Output: A minimum-weight spanning tree (or just minimum spanning tree, or MST) of G.