Mar 10

Interval Scheduling Runtime Analysis

Recall $f(1) \leq f(2) \ldots \leq f(n)$

Greedy alg.

$O(n)$
0. $R = [n] \leftarrow O(n)$
1. $S = \emptyset \leftarrow O(1)$
2. While $R \neq \emptyset \leftarrow \leq n$

$O(n)$
- $O(n) \longrightarrow$ (2.1) Pick $i \in R$ with the smallest index
- $O(1) \longrightarrow$ (2.2) Add i to S
- $O(n) \longrightarrow$ (2.3) Remove all j that conflict with i from R

3. Return $S^* = S \leftarrow O(n)$

Overall runtime : $O(n) + n \cdot O(n) + O(n)$
$= O(n) + O(n^2) + O(n)$
$= O(n^2)$

can we do better?

Shortest path problem

Input: Directed graph $G = (V, E)$

$s \in V$

"length" $\rightarrow$ $\underset{\uparrow \text{ integer}}{l_e} \geq 0 \quad \forall e \in E$

Output: $\forall t \in V$, output <u>a</u> shortest s-t path

w.r.t. the length of the path

$$l(P) = \sum_{e \in P} l_e$$

Simpler version: Only output $d(t) \quad \forall t \in V$

length of the shortest path