Nov 8 — Weighted Interval Scheduling problem

Input: n intervals, i^{th} interval $i \in [n]$: (s_i, f_i, v_i)

(s_i = start time, f_i = finish time, v_i = value)

Output: A valid schedule $S \subseteq [n]$ s.t. its value $v(S) \stackrel{def}{=} \sum_{i \in S} v_i$ is maximized.

Note: Interval Scheduling is a special case $v_i = 1$ $\forall i \in [n]$

Output: Instead of outputting an optimal solution $\mathcal{O}$, output its value $v(\mathcal{O})$

(s_1, f_1, v_1)
$\vdots$
(s_j, f_j, v_j)

Def: $OPT(j)$ = value of an optimal solution for sub-problem $[j]$, $1 \le j \le n$

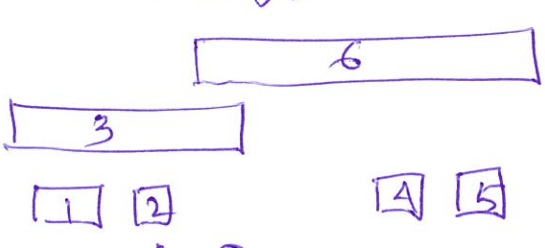
Assume: $f_1 \le f_2 \le \cdots f_n$

Q: Goal? **A:** $OPT(n)$

Def: $\forall j \in [n]$ let $\mathcal{O}_j$ be an optimal soln for $[j]$

$$v(\mathcal{O}_j) = OPT(j)$$

Ex:

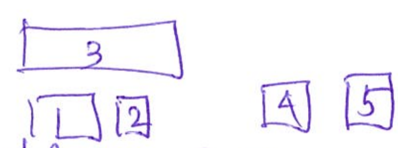


$OPT(6)$?

$6 \in \mathcal{O}_6$

$6 \notin \mathcal{O}_6$

Case 1: $j \notin \mathcal{O}_j$ **Ex:** $6 \notin \mathcal{O}_6 \longrightarrow$



Sub-problem: $[5]$

Claim 1: $\mathcal{O}_6$ is a valid schedule for $[5]$

(Ex) Claim 2: $\mathcal{O}_6$ is an optimal solution for $[5]$

(Ex) Claim 3: $j \notin \mathcal{O}_j \Rightarrow \mathcal{O}_j$ is also optimal for $[j-1]$

Case1: $\Longrightarrow$ $OPT(j) = OPT(j-1)$

$j \notin O_j$

Case2: $j \in O_j$ $\quad 6 \in O_6$ $\quad$ Sub-problem: $\boxed{1}$ $\boxed{2}$

$OPT(6) = v_6 + OPT(2)$

OVERALL: $OPT(6) = \max\{OPT(5), v_6 + OPT(2)\}$