Greedy algorithm design problems

- Eq 1 & 2 can be found online.

- Eq 3. Simple job scheduling problem

- N jobs \( J_1, J_2, J_3, \ldots, J_n \)
- Time for executing every job \( T_i \) (for job \( J_i \))
- We have one machine

Problem 1: What is the order of jobs in order to minimize the total running time?

Problem 2: Minimize the total/average job completion time?
$N = 4$  $T_1$  $T_2$  $T_3$  $T_4 = (4, 1, 6, 3)$

\[ \begin{array}{cccc}
T_1 = 4 & T_2 = 1 & T_3 = 6 & T_4 = 3 \\
0 & 4 & 5 & 11 & 14
\end{array} \]

\[ \frac{4 + 5 + 11 + 14}{4} = \]

\[ \begin{array}{cccc}
T_2 = 1 & T_4 = 3 & T_1 = 4 & T_3 = 6 \\
0 & 1 & 4 & 8 & 14
\end{array} \]

\[ \frac{1 + 4 + 8 + 14}{4} = \]

Solution: Run the quickest job first (the job with minimum $T_i$)

Proof:

"OPT"

\[ \begin{array}{cccc}
& j & \cdots & i \\
T_j & y & z
\end{array} \]

$T_i$ is minimum

"NEW"

\[ \begin{array}{cccc}
i & \cdots & j \\
T_i & y - \#(T_j - T_i) & z
\end{array} \]

$\#$: number of jobs between $j$ and $i$

$y$: completion time of all $\#$ jobs
"opt" = T_j + y + z

\text{new : } T_i + y - \#(T_j - T_i) + z

\text{new - "opt" = } (\# + 1)(T_i - T_j) \leq 0

\text{optimal substructure property : } S = S' + T_i