# Minimization of Total Completion Time on a Single Processor Subject to Dynamic Arrivals

Institute of Information Management
National Yang Ming Chiao Tung University, Taiwan

October 3, 2023

The considered problem is common to computer operating systems and operations management (manufacturing as well as service). Given $n$ jobs $N = \{1, 2, \dots, n\}$ to process on a single processor. Each job $j \in N$ is characterized by positive processing length $p_j$ and non-negative arrival time $r_j$. Any job cannot start its processing before its arrival time. The processor processes at most one job at a time. The objective is to construct a feasible schedule that has the minimum sum of job completion times, or called total completion time. Let $C_j$ denote the completion time of job $j$ in a schedule. The objective function is $\sum_j C_j$, called the total completion time or the sum of completion times of the jobs. To determine an optimal schedule is unfortunately NP-hard.

Consider the following set of 6 jobs.

| job $j$ | 1 | 2 | 3 | 4 | 5 | 6 |
|---|---|---|---|---|---|---|
| $r_j$ | 0 | 2 | 2 | 6 | 7 | 9 |
| $p_j$ | 6 | 2 | 5 | 2 | 8 | 2 |

- **Q1: Implement a full-enumeration tree to list all** $6!$ **schedules and their associated objective values.**

- **Q2: Construct the schedule obtained from the Shortest Remaining Processing Time First (SRPT) rule. Deploy a min-heap to facilitate your implementation of the SRPT rule.**

- **Q3: What is the difference between the optimal value of Q2 and the optimal value of Q1.**

- **Q4: By the way, what is the run time of the SRPT algorithm?**