Deep Dive into Scheduling Algorithms

CSCI 460 Group ID 4

The Problem and Context

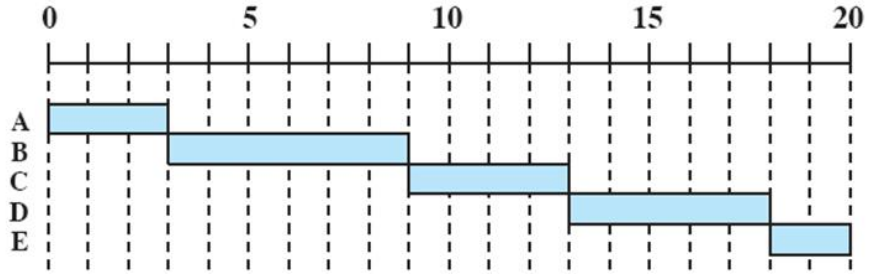
- Scheduling, as described in class, is arguably the most critical, and most complex, task that an operating system must undertake!!
- Thus, multiple algorithms, each with their own unique advantages and disadvantages, are developed in order to solve this complex problem.
- Dive into 4 different types of algorithms, showcasing their strengths and weaknesses!
- At the end, come up with conclusion on which ones are better for what tasks.

Algorithm 1: First Come First Served, Round Robin

Features:

- Easy to implement & understand
- Non-preemptive
- FIFO

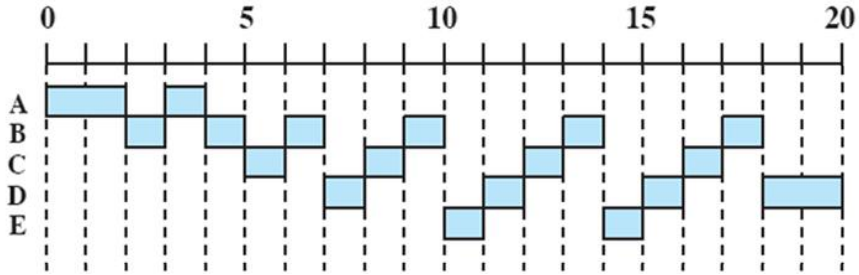
First-Come-First Served (FCFS)



Features:

- Easy to implement & understand
- Clock-based preemption
- Uses time quantum q

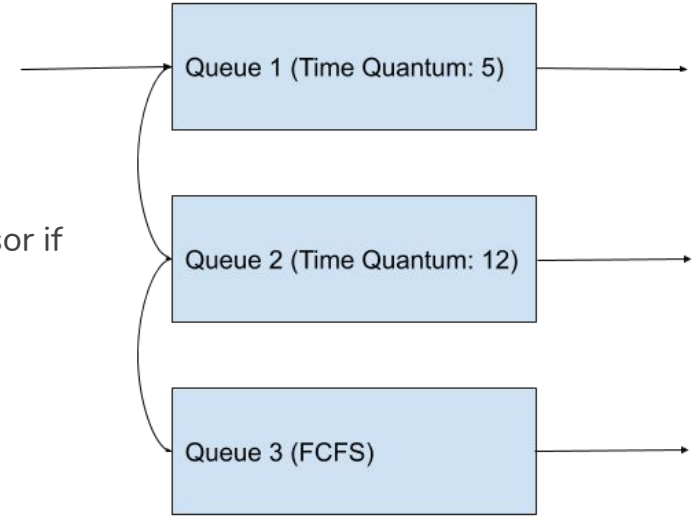
Round-Robin (RR), $q = 1$



Algorithm 2: Multilevel Feedback Queue

Features:

- Multiple Queues
- Different Time Quantums
- Process Booted Off Processor if Longer than Quantum
- Last Queue is FCFS



As a Consequence:

- Favors short processes
- Starvation is possible

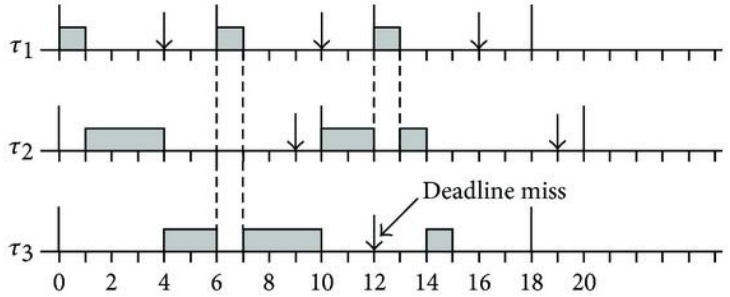
Algorithm 3: Monotonic Deadline

Features:

- Schedules processes based upon Deadlines (Shortest get higher priority)
- Calculations are based upon relationship between compute time, deadlines, and periods.
- Starvation is not possible (one of the 7 assumptions prevents it)

As a Consequence:

- Calculation is based upon 7 assumption that must hold for scheduling to work.
- O(n) execution time can be slow.



Algorithm 4: Highest Response Ratio Next (HRRN)

Features:

- Schedules processes based on response ratio
- Response ratio = (wait time + burst length)/burst length
- Each process must have its response ratio calculated before a process is selected to run

As a Consequence:

- Shorter processes are favored
- No starvation
- Overhead due to calculating response ratio

Results:

Process	Arrival Time	Service Time
A	0	3
B	2	6
C	4	4
D	6	5
E	8	2

Ordering:

- FCFS: A, B, C, D, E
- RR w/ Quantum 3: A, E, B, C, D
- MLFQ w/ Quantum 3 and 5: A,E,B,C,D
- Monotonic Deadline: A, C, D, B, E
- HRRN: A, B, C, E, D

Key Ideas:

- No one algorithm was “best”
  - Speed vs. complexity
  - Ensuring processes can’t starve
  - Some algorithms require different details about each process
  - The “best” scheduling algorithm is dependant on what kind of processes need cpu time

References:

- Deadline Monotonic Image demonstration: Dondo, Julio & Rincón, Fernando & Valderrama Sakuyama, C. & Villanueva, F. & Caba, Julian & López, Juan Carlos. (2014). Facilitating Preemptive Hardware System Design Using Partial Reconfiguration Techniques. TheScientificWorldJournal. 2014. 164059. 10.1155/2014/164059.