Scheduling: MLFQ

Chapter 8

Previously in CS212…

- Optimized for turn around time
  - Shortest Job First (SJF)
  - Shortest Time-to-Completion First (STCF) / Shortest Remaining Time First (SRTF)
- Optimized for response time
  - Round Robin (RR)
- Can we find a compromise for both with little (or no) information about the lifetime/compute requirements of a process in advance (*a priori*)?

Multi-Level Feedback Queue Scheduler

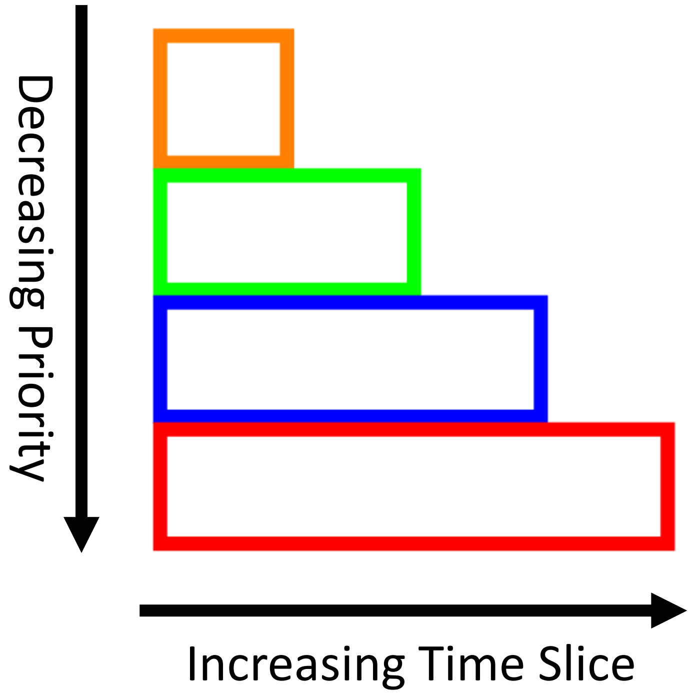
- Multiple Queues
  - Each Queue has a priority level
  - Can have different time quanta lengths for each priority
  - Tracks time quanta usage

- Processes in the same queue are scheduled with round robin

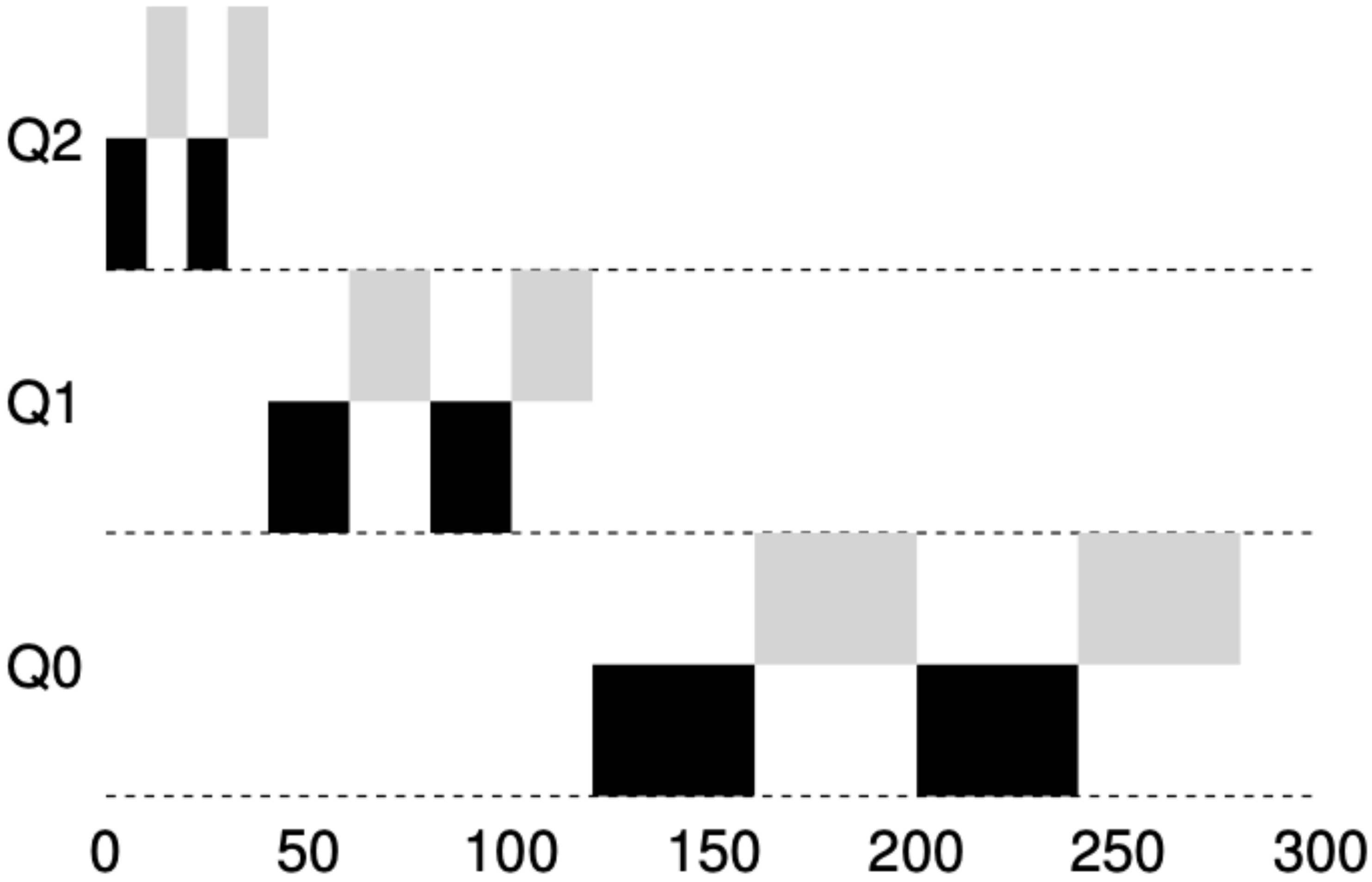
- Changes process priority levels based on observed behavior
  - Learn and predict based on past resource usage

MLFQ Rules

- **Rule 1:** If Priority(A) > Priority(B), A runs (B doesn't)
- **Rule 2:** If Priority(A) == Priority(B), A & B run in round-robin fashion using the time slice (quantum length) of the given queue
- **Rule 3:** When a job enters the system, it is placed at the highest priority
- **Rule 4:** Once a job uses up its time allotment **at a given level** (regardless of how many times it has given up the CPU), its priority is reduced
- **Rule 5:** After some time period *T*, move all the jobs in the system to the highest queue



Example



MLFQ Tuning

- How to selecting the best values for number of queues, time slice per queue, when to priority boost, etc.
    - Sometimes uses “magic numbers” for default values which can later be adjusted
    - This is a challenge

- Users may be permitted to provide some priority **advice/hints** (Unix nice command)