

Assignment 3 Additional Statements

For MLFQ implementation details, some students have concerned about several corner cases. More specific statements will be given in this document.

At a given time t , when multiple simultaneous events occur in the system, the MLFQ scheduler is assumed to handle these events in the following priority order:

1. **Dequeue and Enqueue of Old Process:** When a process exhausts its time slice, it's removed from its current queue. If it has also used up its allotment time, it's enqueued into a lower-priority queue.
2. **Enqueue New Process:** Upon its arrival, a new process is enqueued at the appropriate priority level.
3. **Periodical Boost:** Periodically, all tasks are boosted to the top-most priority level to prevent starvation.
4. **Task Execution:** The currently running task continues its execution until its time slice expires.
5. **Task Selection:** A new task is selected for scheduling from the available tasks in the priority queues.

Based on the **event order list**, in some corner cases, scheduler works as follows:

Situation 1: At time t , if a process A finishes a time slice but not allotment at top queue, and a new process B arrives Result: A is enqueued first and then B, which means process B is at the tail of this queue at time t . You can refer to testcase5 (process 15) for this situation.
Situation 2: if a process A enters at the pre-defined Time Period Result: we assume to sort all processes including process A according to Rule 5. You can refer to testcase4 (process 142) for this situation.
Situation 3: If process A gets a time slice and has not used up its time slice, and a new process B arrives Result: Scheduler will remain to execute A until to Time-slice end.
Situation 4: When the time is S , if there is a process still being execute Result: Scheduler will interrupt the running task immediately and then boost all tasks.
Situation 5: In our assignments, we assume that the allotment time of the lowest priority queue is infinite . You can ignore the given allotment time of the lowest queue in queue.cfg, which is just prepared for you to program more simply.