MLFQ Notes

* If a process uses too much CPU time, it will be moved to a lower priority-queue.
* A process that waits too long in a lower priority-queue may be moved to a higher priority-queue.
  + As the process waits in a lower priority queue, it starts “aging”, and it’s age determines when it should be moved to a higher priority queue
  + Prevents process starvation
* Processes in queues of lower priority can execute only when the queues of higher priority are empty
* Queue
  + For each queue, there will be a round robin scheduler
  + The process will run for (example 8 milliseconds)
    - If it completes execution, it is removed from the queue and that’s it.
    - If it does not complete execution in the 8 milliseconds, the process will be moved down to the next queue of the next highest priority
* Parameters:
  + **# of queues**
  + **Length of time a process runs per queue**
  + **Total process running length**
  + Scheduling algorithm for each queue (lottery or round robin)
  + The method used to determine when to move a process to a lower priority queue
  + The method used to determine which queue a process will enter when that process needs service
  + **# of time slots (example 9 slots, each one lasting 8 milliseconds). When this runs out, every process is moved to the highest priority**