MLQ - Multilevel Queue Scheduling (preemptive)

➡ Associate a priority with each thread and execute highest priority thread first. If same priority, do round-robin.

high-priority queue (e.g system thread) ⟶ T1 → T3 → T6

medium priority (eg. user thread) ⟶ T4

low priority (e.g background thread) ⟶ T2 → T5

◉ **Problem 1 :** starvation of low priority thread

◉ **Problem 2 :** (possibly) starvation of high priority thread

◉ **Problem 3 :** how to decide on the priority?

MLQ - Starvation of high priority thread

1. T1 (low priority) starts, runs and acquires the lock `l`

2. T2 (medium priority) starts, preempts the CPU and runs

3. T3 (high priority) starts, preempts the CPU, runs
   but gets blocked while trying to acquire the lock `l`

4. T2 is elected to run (highest priority thread to be ready to run)

⊙ **Problem** : starvation of a high priority thread

✓ **Solution** : priority donation