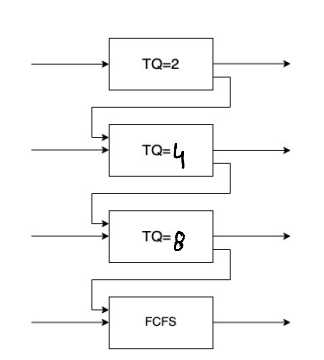
1. Multilevel Queue Scheduling (MLQ):
   1. System Processes: created by OS -> highest priority
   2. Interactive Process(fore ground process): User input required -> medium priority
   3. Batch Process(Background Process) : No Input - > lowest priority
   4. For three tyoes of process, creation of three different ready queue and has three different scheduling algorithms associated
      1. Each queue has its own scheduling algorithm. E.g., SP -> RR, IP -> RR & BP -> FCFS
   5. A process is permanently assigned to one of the queues (inflexible) based on some property of process, memory, size, process priority or process type.
   6. Scheduling among different sub-queues is implemented as fixed priority preemptive scheduling. E.g., foreground queue has absolute priority over background queu
   7. If an interactive process comes & batch process is currently executing. Then, batch process will be preempted.
   8. Problem: Only after completion of all the processes from the top-level ready queue, the further level ready queues will be scheduled. This came starvation for lower priority process
   9. Victim of convoy effect
2. Multi-level feedback queue scheduling (MLFQ
   1. These are flexible in nature
   2. Multiple subqueues
   3. Interqueue movement is allowed
   4. Separate the processes based on BT
   5. Move higher BT process into lower queues
   6. I/O bound & interactive higher priority
   7. In addition, a process that waits too much in a lower-priority queue may be moved to a higher priority queue. This form of ageing prevents starvation.
      1. Less starvation then MLQ.
      2. Can be configured to match a specific system design requirement
      3. Sampe mLFQ
         1. 
3. Design of MLFQ
   1. No. of Queue
   2. Scheduling Algo in each queue
   3. Method to upgrade a process to higher queue
   4. Demote a process to convert process
   5. Process P1 -> which it will be pushed??

Comaprison of all Scheduling Algorithm

