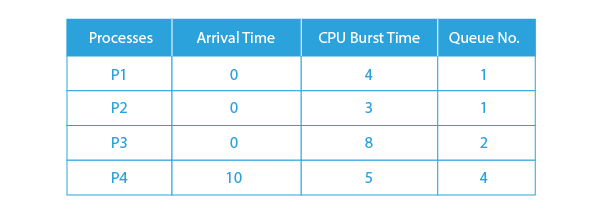
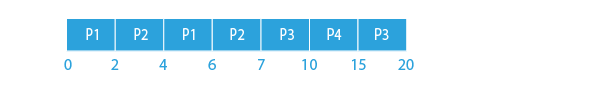
## Example of Multilevel Queue Scheduling

Let’s take an example that explains how Multilevel Queue Scheduling works.



As shown in the preceding table, Queue 1 has a higher priority than Queue 2, and Round Robin is used in Queue 1 (Time Quantum=2), whereas first come, first served is used in Queue 2.

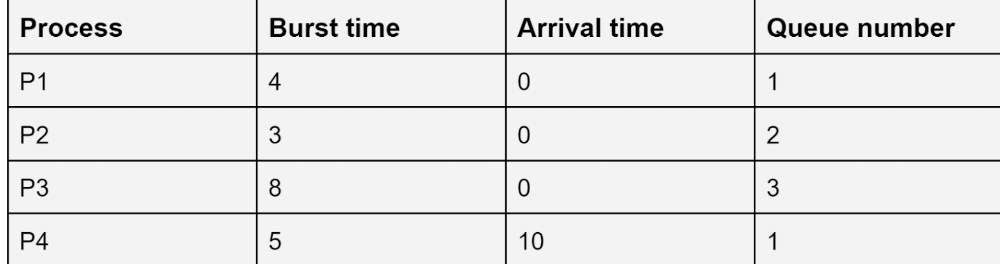
**Lets see the GANT CHART:**



**Working**:  
Both queues are processed from the beginning. As a result, queues (P1, P2) run first (due to higher priority) in a Round Robin manner, with 1 finishing after 7 units. Because there are no processes in Queue 1, the process in Queue 2 (Process P3) starts running, but while it is running, P4 enters Queue 1 and interrupts P3, and then P3 takes the CPU and completes the execution.

<https://www.prepbytes.com/blog/queues/multilevel-queue-mlq-cpu-scheduling/>

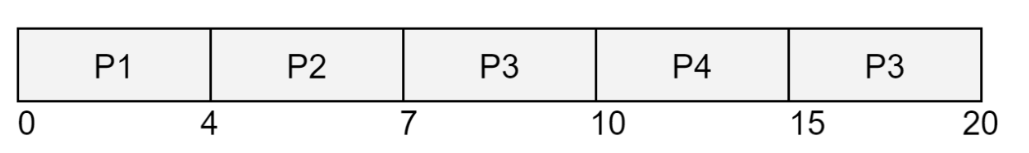
Let us consider the following four processes.



Let's assume that the priority order of the queues is as follows:

Queue1 > Queue2 > Queue3

The Gantt chart will look as follows:



In this example, the processes P1, P2, and P3 arrive at t=0, but still, P1 runs first as it belongs to queue number 1, which has a higher priority. After the P1 process is over, the P2 process in queue 2 runs before the process P3 which is present in queue 3 because queue 2 has a higher priority than queue 3, and then P3 runs. While the P3 process is running, the process P4 belonging to queue 1 arrives. Since queue 1 has higher priority than queue 3, process P3 is stopped (paused), and P4 starts executing. After the execution of P4 is completed, the execution of P3 is resumed.

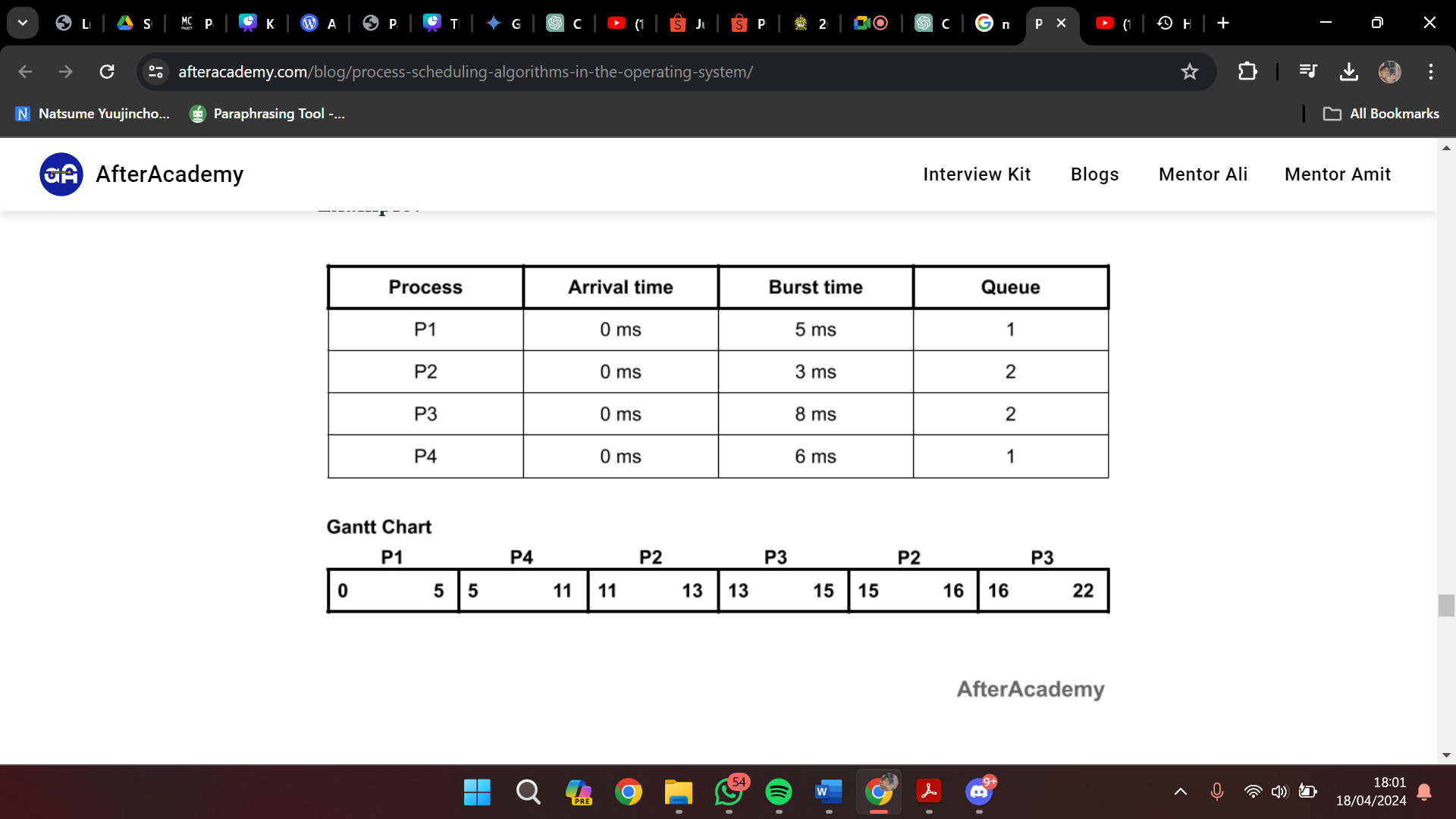
<https://www.naukri.com/code360/library/multilevel-queue-scheduling>

Let us consider a scenario where Queue1 has been assigned processes P1, P2, and P3. Queue2 has been assigned processes P4, P5, P6, and P7; and Queue3 has been assigned processes P8, P9, and P10. No process of Queue2 can be executed unless and until all the processes of Queue1 have been executed. Similarly, no process of Queue3 can be executed unless all processes in Queue2 have been executed i.e. both Queue1 and Queue2 have become empty.

Now, suppose P9 of Queue3 is being executed and a process P11 comes in Queue2, then CPU time will be assigned to P11 i.e CPU will be preempted from P9 and assigned to P11; since Queue2 has a higher priority than Queue3.

So, here we see that if a new process enters a queue with higher priority, preemptive scheduling takes place and a process in the queue with the lowest priority has to be kept waiting for the allocation of CPU, indefinitely. This indefinite waiting period is termed starvation which is one of the biggest drawbacks of Multilevel Queue scheduling.

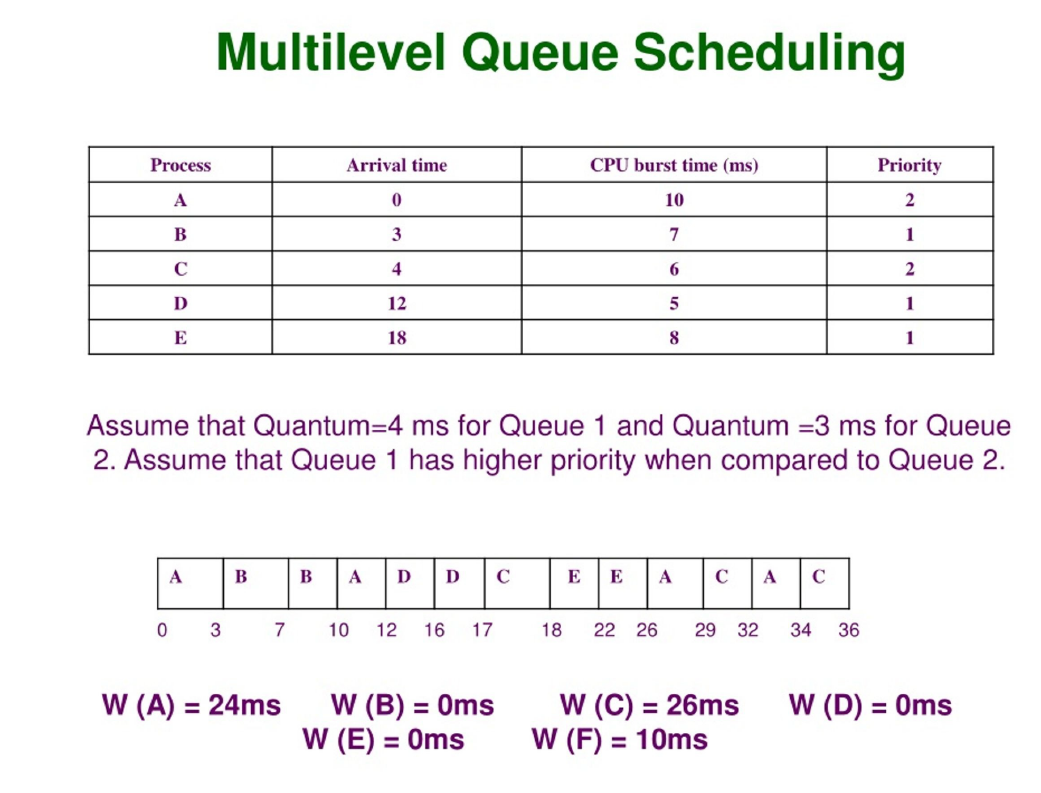
<https://testbook.com/operating-system/multilevel-queue-scheduling>



In the above example, we have two queues i.e. queue1 and queue2. Queue1 is having higher priority and queue1 is using the FCFS approach and queue2 is using the round-robin approach(time quantum = 2ms).

Since the priority of queue1 is higher, so queue1 will be executed first. In the queue1, we have two processes i.e. P1 and P4 and we are using FCFS. So, P1 will be executed followed by P4. Now, the job of the queue1 is finished. After this, the execution of the processes of queue2 will be started by using the round-robin approach.

<https://afteracademy.com/blog/process-scheduling-algorithms-in-the-operating-system/>



<https://www.slideserve.com/nschofield/chapter-6-cpu-scheduling-powerpoint-ppt-presentation>