Round Robin Scheduling | Operating System – M02 P07

This is a multipart blog article series, and in this series I am going to explain you the concepts of operating system. This article series is divided into multiple modules and this is the second module which consists of 11 articles.

Round robin is one of the most famous CPU scheduling algorithm. We will see a question on it and try to understand the concepts of round robin CPU scheduling in an operating system.

**Round Robin Scheduling:**

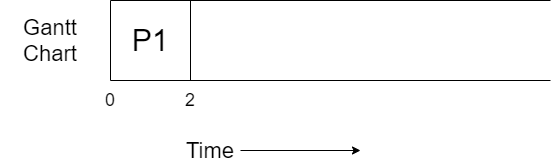
* Criteria of round robin is “time quantum”
* Mode of round robin scheduling is pre-emptive.

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Process No. | Arrival time | Burst time | Completion time | Turnaround time | Waiting time | Response time |
| P1 | 0 | 5 |  |  |  |  |
| P2 | 1 | 4 |  |  |  |  |
| P3 | 2 | 2 |  |  |  |  |
| P4 | 4 | 1 |  |  |  |  |

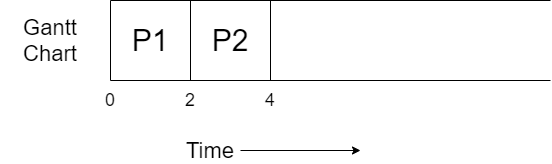
Given time quantum: - 2 unit of time.

Before solving this question let’s see some points, which will help to solve this question.

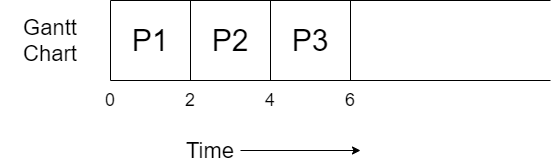
* In round robin context switching is done.
* Context switching means to save the context of a running process. Context means PCB (process complete block), and sending back the process to ready queue and start executing next process, we save the context because we do not want to execute it from beginning rather we want to start executing that process from where we left.



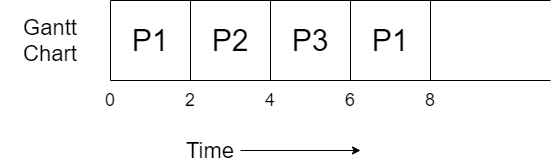
* At time unit 0 process P1 arrives to CPU start executing P1, till 2 unit of time because the time quantum given in the question is 2.



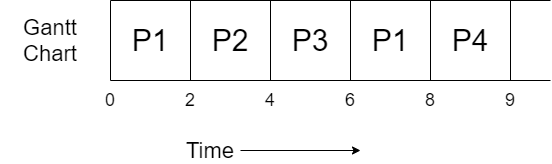
* P2 process arrives at 1 time unit so it was present in the ready queue, so when P1 get preempt then P2’s execution start.
* P2 execute for 2 time unit and then it will also get preempt.



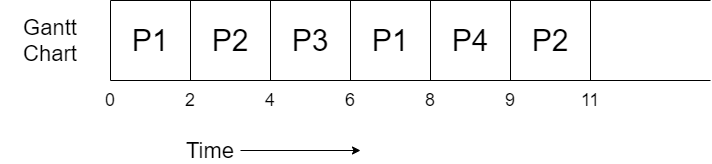
* Process P3 has arrived at time unit 2, all the processes are arrived in the ready queue.
* P3 will also get executed for 2 unit of time, and then got terminated as its burst time was also 2 time unit.



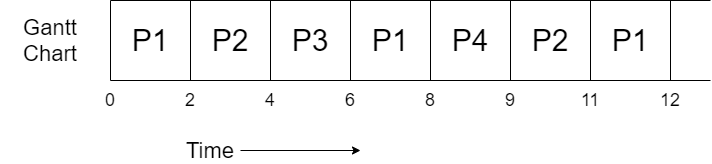
* Now, P1 was in ready queue before P4 that’s why we will execute P1 for 2 time unit and then it will get pre-empted.



* P4 will execute for 1 time unit because its burst time is only 1, and get terminated.



* Now P2 will execute for 2 time unit and after 2 time unit it will get terminated; now only P1 is in the ready queue.



* P1 will get executed for 1 time unit and then get terminated.
* Six time context switching happened while calculating. We do not count first and last (which is 0 and 12 in our case.)

All the process got completely executed. Now we will complete the table, but for it you need to know some formulas.

* Completion time: The time at which process completes its execution.
* Turnaround time = Completion time – arrival time.
* Waiting time = turnaround time – burst time.
* Response time = CPU first time allocated – Arrival time.

So, the final table obtained will be.

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Process No. | Arrival time | Burst time | Completion time | Turnaround time | Waiting time | Response time |
| P1 | 0 | 5 | 12 | 12 | 7 | 0 |
| P2 | 1 | 4 | 11 | 10 | 6 | 1 |
| P3 | 2 | 2 | 6 | 4 | 2 | 2 |
| P4 | 4 | 1 | 9 | 5 | 4 | 4 |

So, this was all about Round Robin CPU scheduling. Hope you like it and learned something new from it.

If you have any doubt, question, queries related to this topic or just want to share something with me, then please feel free to contact me.