MCA Semester- II

Subject: Operating System & Shell Programming

Paper Code: MCACS2T06

Unit:2 Process Scheduling- CPU SCHEDULING

First-Come-First-Served (FCFS) Scheduling

Consider four processes P1, P2, P3, P4 with their arrival times and required CPU burst(in ms) as shown in the following table.

Process	**P1**	**P2**	**P3**	**P4**
Arrival Time	0	2	3	5
CPU burst (ms)	15	6	7	5

How will these processes be scheduled according to FCFS scheduling algorithm? Compute the average waiting time and average turnaround time?

Solution

First we will create a Gantt Chart

P1	P2	P3	P4

0 P1 exits → **15**+6=21 P2 exits → **21**+7=28 P3 exits → **28**+5=33 **33**

Initially P1 enters the ready queue at t(Arrival time)=0 and CPU is allocated to it. While P1 is executingP2, P3 and P4 enter the ready queue at t=2, t=3 and t=5, respectively. When P1 completes CPU is allocated to P2 as it has entered before P3 and P4. When P2 completes, P3 gets the CPU after which P4 gets the CPU.

Waiting time for P1= 0 ms as P1 starts immediately

Waiting time for P2 = (15- 2) = 13 ms as P2 enters at t=2 and starts at t= **15**

Waiting time for P3 = (21-3) = 18 ms as P3 enters at t=3 and starts at t = **21**

Waiting time for P4 = (28-5) = 23 ms as P4 enters at t= 5 and starts at t= **28**

Average Waiting Time = (0+13+18+23)/4= 13.5 ms

Turnaround time for P1= (15-0)=15ms as P1 enters at t=0 and exits t=**15**

Turnaround time for P2 =(21-2)=19 ms as P2 enters at t=2 and exits at t=**21**

Turnaround time for P3=(28-3)=25 ms as P3 enters at t=3 and exits at t=**28**

Turnaround time for P4=(33-5)=28 ms as P4 enters at t=5 and exits at t=**33**

Average turnaround time =(15+19+25+28)/4=21.75 ms

Shortest Job First (SJF) Scheduling

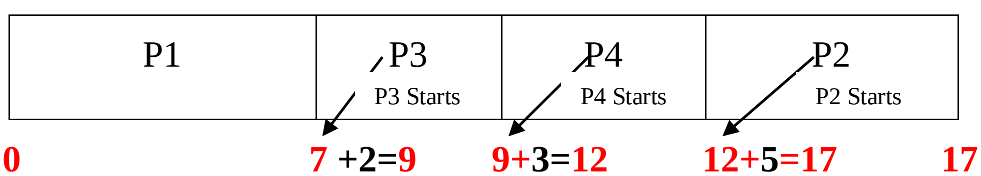
Consider four processes P1, P2, P3, P4 with their arrival times and required CPU burst(in ms) as shown in the following table.

Process	**P1**	**P2**	**P3**	**P4**
Arrival Time	0	1	3	4
CPU burst (ms)	7	5	2	3

How will these processes be scheduled according to SJF scheduling algorithm? Compute the average waiting time and average turnaround time?

Solution

First we will create a Gantt Chart



Initially P1 enters the ready queue at t=0 and gets the CPU as there are no other processes in the queue. While it is executing P2, P3 and P4 enter the queue at t=1, t=3 and t=4 respectively. When CPU becomes free that is at t=7, it is allocated to P3 because it is having the shortest CPU burst among the three processes. When P3 gets completed, CPU is allocated first to P4 and then to P2.

Waiting time for P1 = 0 ms as P1 starts immediately

Waiting time for P2=(12-1)= 11ms as P2 enters at t=1 and starts at t=**12**

Waiting time for P3= (7-3)=4 ms as P3 enters at t=3 and starts at t=**7**

Waiting time for P4= (9-4)5 ms as P4 enters at t=4 and starts at t=**9**

Average Waiting time = (0+11+4+5)/4= 5 ms

Turnaround time for P1= (7-0) =7ms as P1 enters at t=0 and exits at t=**7**

Turnaround time for P2= (17-1) =16 ms as P2 enters at t=1 and exits at t=**17**

Turnaround time for P3 = (9-3) =6 ms as P3 enters at t=3 and exits at t=**9**

Turnaround time for P4 = (12-4)=8 ms as p4 enters at t=4 and exits at t=**12**

Average Turnaround time =(7+16+6+8)/4=9.25 ms

Round Robin (RR) Scheduling

The round robin scheduling is one of the most widely used preemptive scheduling algorithms which considers all the processes as equally important and treats them in a favourable manner. Each process in the ready queue gets a fixed amount of CPU time (generally from 10 to 100 milliseconds) known as time slice or time quantum for the execution.

If the process does not execute completely till the end of time slice, it is preempted and the CPU is allocated to the next process in the ready queue. However, if the process blocks or terminates before the time slice expires, the CPU is switched to the next process in the ready queue at that moment only. Context switch is used to save states of preemptive processes

The performance of round robin scheduling is greatly affected by the size of the time quantum. If the time quantum is too small, a number of context switches accur which in turn increase the system overhead. The more time will be spent in performing context switching rather than executing the processes. On the other hand, if the time quantum is too large the performance of round robin simple degrades to FCFS.

It is efficient for time sharing systems where the CPU time is divided among the competing processes. The disadvantage is that the processes may take long time to execute. This decreases the system throughput.

The average wait time in the RR scheduling may be often quite long. Consider the following set of processes that **arrive at time 0**, with the required CPU time in milliseconds given under.

Queue	**Q1**	**Q2**	**Q3**
Process	P1	P2	P3
CPU burst (ms)	24 +	3 +	3 =30

Assuming that the CPU time slice is **4** ms , the scheduling takes place as explained

CPU burst time (24-**4**=20)ms.

Solution

First we will create a Gantt Chart

P1	P2	P3	P1	P1	P1	P1	P1

0   **4**+3=7   7+3=10   10+**4**   14+**4**   18+**4**   22+**4**   26+**4**   **30**

CPU Burst(ms)=(24+3+3=30)

Process P2 however completes it execution before the time slice expires & then P3 is allocated

Waiting time for P1= (10-4) 6 ms

Waiting time for P2 = 4 ms

Waiting time for P3 = 7 ms

Average Waiting Time = (6+4+7)/3= 5.66 ms

Turnaround time for P1= (30-0) =30 ms as

Turnaround time for P2 =7 ms as P2 exits at t=7

Turnaround time for P3= 10 ms as P3 exits at t=10

Average turnaround time =(30+7+10)/3= 15.6 ms

Note : Arrival time is 0(zero)

Faculty: Ms Sushmita Chakraborty
Assistant Professor
Deptt. of MCA, Patna Women’s College