CPU Scheduling

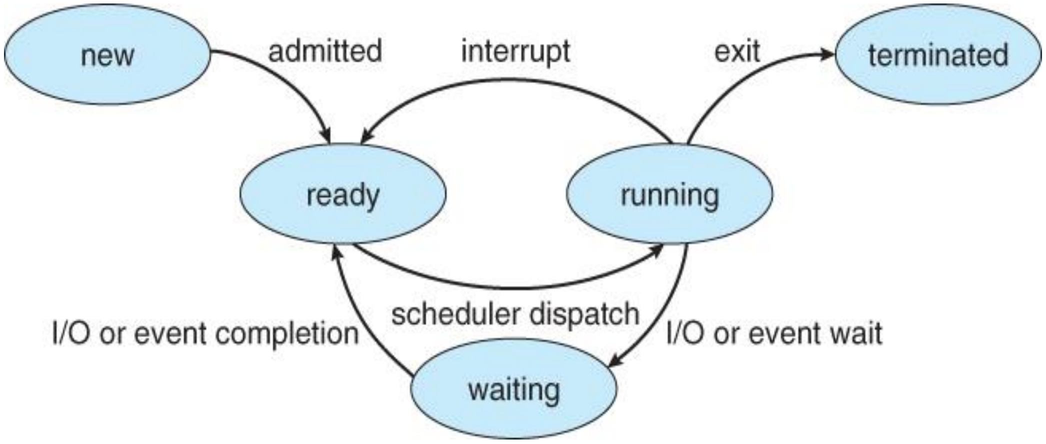
Fill in the blanks

1) A program in execution is called ------------ . process
2) Interval between the time of submission and completion of the job is called turn around time
3) A scheduler which selects processes from secondary storage device is called…… long term schedular
4) -------- system is used when there are rigid time requirements on the operation of processor.
5) The ------------- memory is a technique that allows the execution of processes that may not be completely in memory.
6) In ------- scheduling algorithm, a small unit of time, called time quantum, for each process. round robin

Type of CPU Scheduling

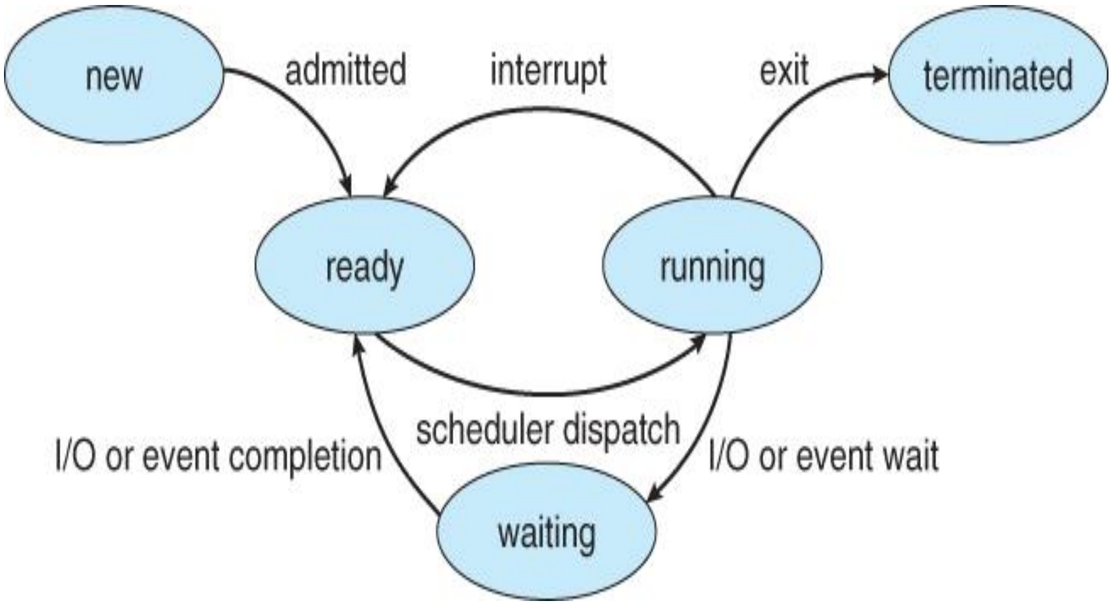
CPU scheduling decisions may take place under the following four circumstances:

1. When a process switches from the **running** state to the **waiting** state (for I/O request or invocation of wait for the termination of one of the child processes).
2. When a process switches from the **running** state to the **ready** state (for example, when an interrupt occurs).
3. When a process switches from the **waiting** state to the **ready** state (for example, completion of I/O).
4. When a process **terminates**.



Type of CPU Scheduling

- In circumstances 1 and 4, there is no choice in terms of scheduling. A new process (if one exists in the ready queue) must be selected for execution. There is a choice, however in circumstances 2 and 3.
- When Scheduling takes place only under circumstances **1** and **4**, we say the scheduling scheme is **Non-preemptive**; otherwise the scheduling scheme is **Preemptive**.



1. **Running to Waiting**
2. **Running to Ready**
3. **Waiting to Ready**
4. **Process Terminate**

Type of CPU Scheduling

Non-Preemptive Scheduling

Under non-preemptive scheduling, once the CPU has been allocated to a process, the process keeps the CPU until

1. **Process is terminated**
2. **Switches to the waiting state.**

Preemptive Scheduling

The resources (mainly CPU cycles) are allocated to the process for the limited amount of time and then is taken away, and the process is again placed back in the ready queue if that process still has CPU burst time remaining. That process stays in ready queue till it gets next chance to execute.

When preemptive scheduling is done:

1. High Priority Process Arrives
2. Times Slice Expires
3. Process completes it I/O Operation
4. Generate a child process

CPU Scheduling Criteria

- **Throughput:** It is the total number of processes completed per unit time or rather say total amount of work done in a unit of time. This may range from 10/second to 1/hour depending on the specific processes.
- **Turnaround Time:** It is the amount of time taken to execute a particular process, i.e. the interval from time of submission of the process to the time of completion of the process.
- **Waiting Time:** The sum of the periods spent waiting in the ready queue amount of time a process has been waiting in the ready queue to acquire get control on the CPU.

CPU Scheduling Criteria cont…

- **Load Average:** It is the average number of processes residing in the ready queue waiting for their turn to get into the CPU.
- **Response Time:** Amount of time it takes from when a request was submitted until the first response is produced. Remember, it is the time till the first response and not the completion of process execution (final response).

In general CPU utilization and Throughput are maximized and other factors are reduced for proper optimization.

- **CPU Utilization:** To make out the best use of CPU and not to waste any CPU cycle, CPU would be working most of the time (Ideally 100% of the time).

Considering a real system, CPU usage should range from 40% (lightly loaded) to 90% (heavily loaded).

FCFS

- It is a Non-Preemptive CPU scheduling algorithm, so after the process has been allocated to the CPU, it will never release the CPU until it finishes executing.

Process	Burst time
P0	7
P1	5
P2	2
P3	9

Gantt chart:

P0	P1	P2	P3	
0	7	12	14	23

- Average Waiting time=8.25
- Average TAT=14

FCFS

- What is the average waiting time for the processes running under FCFS?

Process	Burst time	Arrival time
P1	6	2
P2	3	5
P3	8	1
P4	3	0
P5	4	4

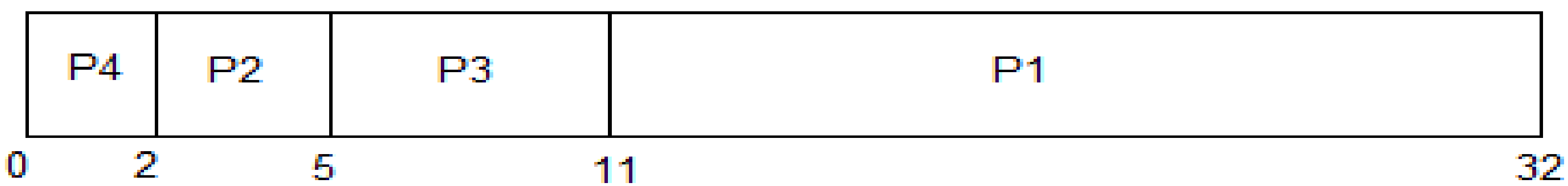
- Waiting time=8

Shortest-Job-First (SJF) Scheduling

- Out of all available (ready queue) processes, It selects the process with the smallest burst time to execute next.
  - Preemptive
  - Non-preemptive

Q-1

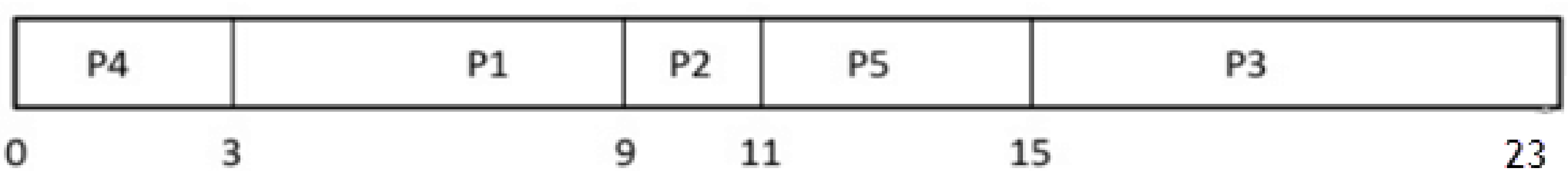
PROCESS	BURST TIME
P1	21
P2	3
P3	6
P4	2



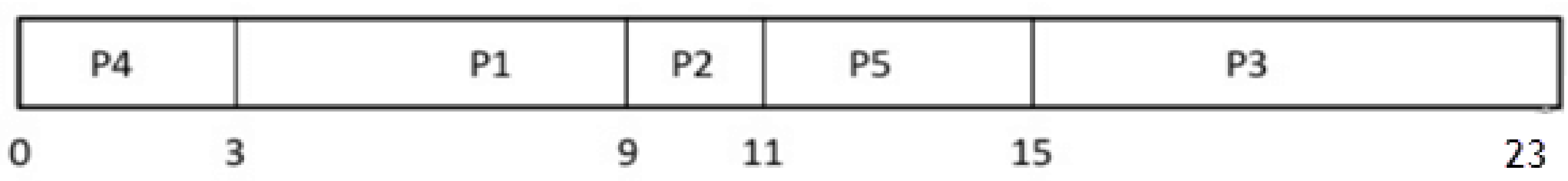
Now, the average waiting time will be = (0 + 2 + 5 + 11)/4 = 4.5 ms

Example 2

Process	Queue Burst time	Arrival time
P1	6	2
P2	2	5
P3	8	1
P4	3	0
P5	4	4



Average Waiting Time= (0+1+4+7+14)/5 = 26/5 = 5.2



- Waiting time
  - P4= 0-0=0
  - P1= 3-2=1
  - P2= 9-5=4
  - P5= 11-4=7
  - P3= 15-1=14
- Average Waiting Time= (0+1+4+7+14)/5 = 26/5 = 5.2

Q1

- Consider the following CPU processes with arrival times (in milliseconds) and length of CPU bursts (in milliseconds) as given below: **(GATE-2017)**

Process	Arrival time	Burst time
P1	0	7
P2	3	3
P3	5	5
P4	6	2

- If the pre-emptive shortest remaining time first scheduling algorithm is used to schedule the processes, then the average waiting time across all processes is __________ milliseconds.
- Answer: 3

Priority Scheduling

- Priority Scheduling is a method of scheduling processes based on priority.
- The processes with higher priority should be carried out first, whereas jobs with equal priorities are carried out on FCFS basis.
- Priority depends upon memory requirements, time requirements, etc.

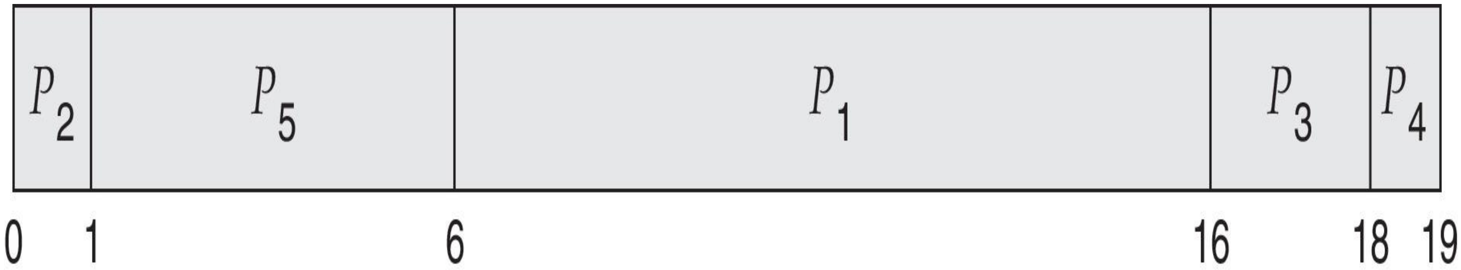
Types of Priority Scheduling

- **Preemptive**
  - A preemptive priority scheduling will preempt the CPU if the priority of the newly arrived process is higher than the currently running process.
- **Non-preemptive**
  - A non-preemptive priority scheduling algorithm will simply execute the process till its termination.

Example of Priority Scheduling

Process	Burst Time	Priority
P_1	10	3
P_2	1	1
P_3	2	4
P_4	1	5
P_5	5	2

Assume low numbers represent high priority

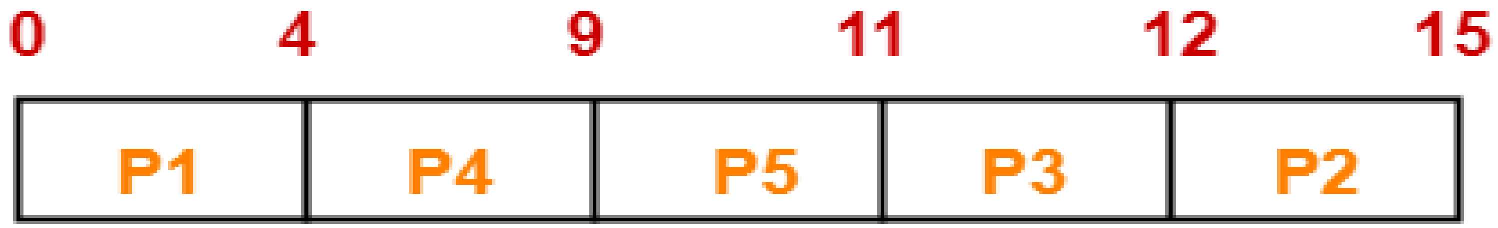


- Average waiting time = 8.2

P

Process Id	Arrival time	Burst time	Priority
P1	0	4	2
P2	1	3	3
P3	2	1	4
P4	3	5	5
P5	4	2	5

Calculate the average waiting time and average turn around time. (Higher number represents higher priority)



0	4	9	11	12	15
P1	P4	P5	P3	P2	

Process Id	Arrival time	Burst time	Priority
P1	0	4	2
P2	1	3	3
P3	2	1	4
P4	3	5	5
P5	4	2	5

Process Id	CT	TAT(CT-AT)	WT(TAT-BT)
P1	4	4 – 0 = 4	4 – 4 = 0
P2	15	15 – 1 = 14	14 – 3 = 11
P3	12	12 – 2 = 10	10 – 1 = 9
P4	9	9 – 3 = 6	6 – 5 = 1
P5	11	11 – 4 = 7	7 – 2 = 5

- Average Turn Around time = (4 + 14 + 10 + 6 + 7) / 5 = 41 / 5 = 8.2 unit
- Average waiting time = (0 + 11 + 9 + 1 + 5) / 5 = 26 / 5 = 5.2 unit

Q2

- Consider the set of processes with arrival time (in milliseconds), CPU burst time (in milliseconds) , and priority (0 is the highest priority) shown below. None of the processes have I/O burst time. **(GATE-2017)**

Process	Arrival time	Burst Time	Priority
P1	0	11	2
P2	5	28	0
P3	12	2	3
P4	2	10	1
P5	9	16	4

- The average waiting time (in milliseconds) of all the processes using preemptive priority scheduling algorithm is __________________
- Answer: 29

Explanation

P1	P4	P2	P4	P1	P3	P5

0 2 5 33 40 49 51 67

Waiting Time = 0 + (33 - 5) + (40 - 2) + (49 - 12) + (51 - 9) = 145

Average waiting time: 145/5 = 29

P-1

- Which scheduling policy is most suitable for a time-shared operating systems?

  (a) Shortest Job First
  (b) Round Robin
  (c) First Come First Serve
  (d) Elevator

- option (b)

P-2

- Which of the following scheduling algorithms is non-preemptive?(***GATE-2002***)

(a) Round-Robin

(b) First In First Out

(c) Multilevel Queue Scheduling

(d) Multilevel Queue Scheduling with Feedback

Explanation

- Round Robin – Preemption takes place when the time quantum expires
- First In First Out – No Preemption, the process once started completes before the other process takes over
- Multi Level Queue Scheduling – Preemption takes place when a process of higher priority arrives
- Multi Level Queue Scheduling with Feedback – Preemption takes a place when process of higher priority arrives or when the quantum of high priority queue expires and we need to move the process to low priority queue

P-3

- A major problem with priority scheduling is:

- A. Definite blocking
- B. Starvation
- C. Low Priority
- D. None of these

P-4

- An optimal scheduling algorithm in terms of minimizing average waiting time of a given set of processes is..

- A. FCFS
- B. Round Robin
- C. SJF
- None of these

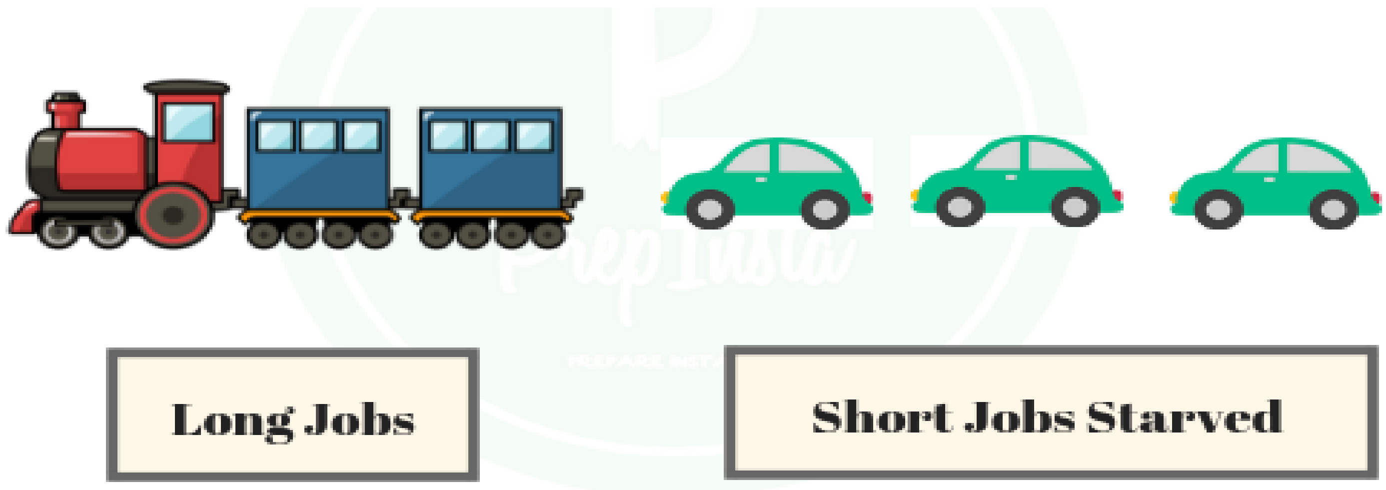
P-5

- In priority scheduling a priority number is associated with each process. The CPU is allocated to the process with the highest priority. The problem of starvation of low priority process may never execute is resolved by:

- A. Terminating the process
- B. Aging
- C. Mutual Exclusion
- D. Semaphore

- The Round Robin scheduling is essentially a preemptive version of ..
- A. FIFO
- B. SJF
- C. SRTF
- D. LTF

Name this phenomena?



Phenomena -> Starvation

which is occur in priority scheduling

Q-11

- Consider three processes (process id 0, 1, 2 respectively) with compute time bursts 2, 4 and 8 time units. All processes arrive at time zero. Consider the longest remaining time first (LRTF) scheduling algorithm. In LRTF ties are broken by giving priority to the process with the lowest process id. The average turn around time is:

  (a) 13 units (b) 14 units (c) 15 units (d) 16 units

- Ans: option (a)

Explanation

Question 48 Explanation:

Using SRTF:

A	B	A	C	E	D	
0	3	5	8	12	15	21

TAT(A) = 8-0 = 8, TAT(B)= 5-3=2, TAT(C)= 12-5=7, TAT(D)= 21-7= 14, TAT(E)=15-10=5

Average turnaround time = 8+2+7+14+5/ 5 = 7.2ms

Q-6

- Four jobs to be executed on a single processor system arrive at time 0 in the order A, B, C, D. their burst CPU time requirements are 4, 1, 8, 1 time units respectively. The completion time of A under round robin scheduling with time slice of one time unit is

  (a) 10 (b) 4 (c) 8 (d) 9

- Ans: option (d)

Q-9

- Three processes A, B and C each execute a loop of 100 iterations. In each iteration of the loop, a process performs a single computation that requires tc CPU milliseconds and then initiates a single I/O operation that lasts for tio milliseconds. It is assumed that the computer where the processes execute has sufficient number of I/O devices and the OS of the computer assigns different I/O devices to each process. Also, the scheduling overhead of the OS is negligible. The processes have the following characteristics:

Process id	t_c	t_{io}
A	100 ms	500 ms
B	350 ms	500 ms
C	200 ms	500 ms

- The processes A, B, and C are started at times 0, 5 and 10 milliseconds respectively, in a pure time sharing system (round robin scheduling) that uses a time slice of 50 milliseconds. The time in milliseconds at which process C would *complete* its first I/O operation is ____________.

Explanation

- There are three processes A, B and C that run in round robin manner with time slice of 50 ms.
- Processes atart at 0, 5 and 10 miliseconds. The processes are executed in below order
- A, B, C, A
- 50 + 50 + 50 + 50 (200 ms passed)
- Now A has completed 100 ms of computations and goes for I/O now
- B, C, B, C, B, C
- 50 + 50 + 50 + 50 + 50 + 50 (300 ms passed)
- C goes for i/o at 500ms and it needs 500ms to finish the IO.
- So C would complete its first IO at 1000 ms

Q-7

- Consider the 3 processes, P1, P2 and P3 shown in the table.

Process	Arrival time	Time Units Required
P1	0	5
P2	1	7
P3	3	4

- The completion order of the 3 processes under the policies FCFS and RR2 (round robin scheduling with CPU quantum of 2 time units) are**(GATE-2012)**
- (A) **FCFS:** P1, P2, P3 **RR2:** P1, P2, P3
- (B) **FCFS:** P1, P3, P2 **RR2:** P1, P3, P2
- (C) **FCFS:** P1, P2, P3 **RR2:** P1, P3, P2
- (D) **FCFS:** P1, P3, P2 **RR2:** P1, P2, P3
- **(C)**

Round Robin with Arrival time

Process Id	Arrival time	Burst time
P1	0	5
P2	1	3
P3	2	1
P4	3	2
P5	4	3

Gantt Chart

Process Id	Exit time	Turn Around time	Waiting time
P1	13	13 – 0 = 13	13 – 5 = 8
P2	12	12 – 1 = 11	11 – 3 = 8
P3	5	5 – 2 = 3	3 – 1 = 2
P4	9	9 – 3 = 6	6 – 2 = 4
P5	14	14 – 4 = 10	10 – 3 = 7
	Average	**8.6 Unit**	**5.8 Unit**

Q-12

- Which of the following statement(s) is/are correct in the context of CPU scheduling?(GATE-2021)
- Turnaround time includes waiting time
- The goal is to only maximize CPU
- utilization and minimize throughput Round-robin policy can be used even when the CPU time required by each of the processes is not known apriori
- Implementing preemptive scheduling needs hardware support
- A,C,D

Q13

- Three processes arrive at time zero with CPU bursts of 16,20 and 10 milliseconds. If the scheduler has prior knowledge about the length of the CPU bursts, the minimum achievable average waiting time for these three processes in a non-preemptive scheduler (rounded to nearest integer) is _______________ milliseconds. (GATE-2021)

- Required to find minimum avg. weighting time of non preemptive scheduler.
- SJF give minimum avg. weighting time

No	Process	B.T.	C.T.	T.A.T.	W.T.
1	P1	16	26	26	10
2	P2	20	46	46	26
3	P3	10	10	10	0

Q-14

- In real-time operating systems, which of the following is the most suitable scheduling scheme?
- round-robin
- first-come-first-served
- preemptive
- random scheduling

Explanation

- Generally We know that The Real Time Operating System jobs are highly associated with The Timings or Deadlines. So it is always preferable to use a Preemptive scheduling scheme.
- Therefore we can eliminate B option as FCFS is a Non-preemptive scheduling scheme.
- Option D is also not preferable because we should use only preemptive scheme. But Random scheduling means any Non preemptive or Preemptive.
- Now Option A, Yes it is preemptive but still it’s not preferable as we will not have any priority oriented mechanisms in it…It slightly follows FCFS.
- So,It is C..Preemptive scheduling scheme …

Q-15

Which of the following scheduling algorithms could result in starvation?

- Priority
- Round Robin
- FCFS
- None of the above

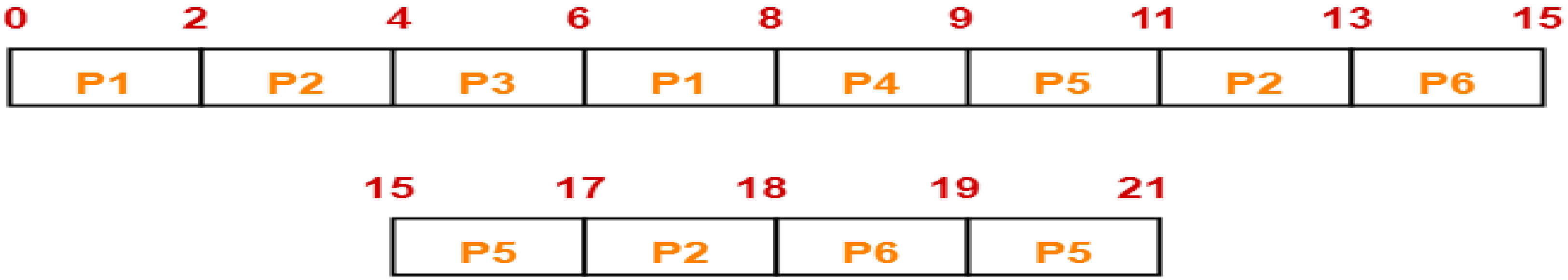
Explanation

- starvation in scheduling algorithm occurs when any process is favoured over other.

  since FCFS and RR doesnt favour any process therfore they are starvation free.in priority based scheduling more priority process gets favoured over low priority hence Ans: A

Another Example

Process Id	Arrival time	Burst time
P1	0	4
P2	1	5
P3	2	2
P4	3	1
P5	4	6
P6	6	3



Gantt Chart

Process Id	Exit time	Turn Around time	Waiting time
P1	8	8 – 0 = 8	8 – 4 = 4
P2	18	18 – 1 = 17	17 – 5 = 12
P3	6	6 – 2 = 4	4 – 2 = 2
P4	9	9 – 3 = 6	6 – 1 = 5
P5	21	21 – 4 = 17	17 – 6 = 11
P6	19	19 – 6 = 13	13 – 3 = 10
	Average	**10.84**	**7.33**

Q-10

- A uni-processor computer system only has two processes, both of which alternate 10ms CPU bursts with 90ms I/O bursts. Both the processes were created at nearly the same time. The I/O of both processes can proceed in parallel. Which of the following scheduling strategies will result in the least CPU utilization (over a long period of time) for this system ?

  (A) First come first served scheduling
  (B) Shortest remaining time first scheduling
  (C) Static priority scheduling with different priorities for the two processes
  (D) Round robin scheduling with a time quantum of 5 ms

Answer: (D)

Explanation:

- When Round Robin scheduling is used .We are given that the time slice is 5ms. Consider process P and Q.
  Say P utilizes 5ms of CPU and then Q utilizes 5ms of CPU. Hence after 15ms P starts with I/O And after 20ms Q also starts with I/O. Since I/O can be done in parallel, P finishes I\O at 105th ms (15 + 90) and Q finishes its I\O at 110th ms (20 + 90). Therefore we can see that CPU remains idle from 20th to 105th ms.
  That is when Round Robin scheduling is used,
  Idle time of CPU = 85ms
  CPU Utilization = 20/105 = 19.05%
- When First Come First Served scheduling scheduling or Shortest Remaining Time First is used
  Say P utilizes 10ms of CPU and then starts its I/O. At 11th ms Q starts processing. Q utilizes 10ms of CPU.
  P completes its I/O at 100ms (10 + 90)
  Q completes its I/O at 110ms (20 + 90)
  At 101th ms P again utilizes CPU. Hence,
  Idle time of CPU = 80ms
  CPU Utilization = 20/100 = 20%
- Since only two processes are involved and I\O time is much more than CPU time, “Static priority scheduling with different priorities” for the two processes reduces to FCFS or Shortest remaining time first.
  Therefore, Round robin will result in least CPU utilization.

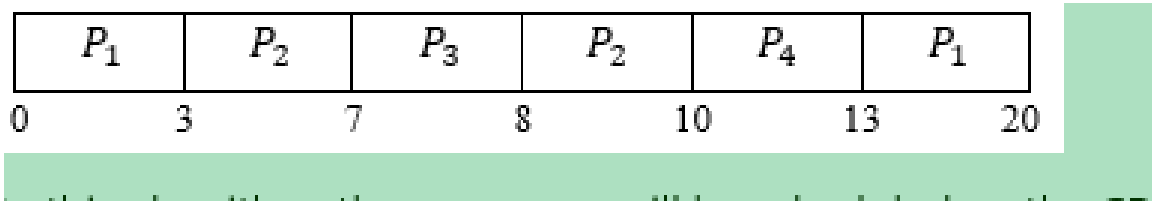
Q3

- Consider the following processes, with the arrival time and the length of the CPU burst given in milliseconds. The scheduling algorithm used is preemptive shortest remaining-time first.

Process	Arrival Time	Burst Time
P_1	0	10
P_2	3	6
P_3	7	1
P_4	8	3

- The average turn around time of these processes is milliseconds ________(**GATE 2016)**
- **Answer: 8.25**

Explanation



- Turnaround Time (TAT) = Completion Time (CT) - Arrival Time (AT)
  TAT for P_1 = 20 - 0 = 20,
  TAT for P_2 = 10 - 3 = 7,
  TAT for P_3 = 8 - 7 = 1,
  TAT for P_4 = 13 - 8 = 5.
  Total TAT = 20 + 7 + 1 + 5 = 33 / 4 = 8.25 (Avg. TAT)

Q4

- For the processes listed in the following table, which of the following scheduling schemes will give the lowest average turnaround time?(**GATE 2015)**

Process	Arrival Time	Processing Time
A	0	3
B	1	6
C	4	4
D	6	2

- (A) First Come First Serve
- (B) Non – preemptive Shortest Job First
- (C) Shortest Remaining Time
- (D) Round Robin with Quantum value two
- **Shortest Remaining Time**

Q5

- Consider the following set of processes that need to be scheduled on a single CPU. All the times are given in milliseconds.

Process Name	Arrival Time	Execution Time
A	0	6
B	3	2
C	5	4
D	7	6
E	10	3

- Using the *shortest remaining time first* scheduling algorithm, the average process turnaround time (in msec) is ______________________**(GATE -2014)**

- **7.2 to 7.2**

Q6

- A scheduling algorithm assigns priority proportional to the waiting time of a process. Every process starts with priority zero (the lowest priority). The scheduler re-evaluates the process priorities every *T* time units and decides the next process to schedule. Which one of the following is TRUE if the processes have no I/O operations and all arrive at time zero? **(GATE-2013)**
- (A) This algorithm is equivalent to the first-come-first-serve algorithm.
- (B) This algorithm is equivalent to the round-robin algorithm.
- C) This algorithm is equivalent to the shortest-job-first algorithm.
- (D) This algorithm is equivalent to the shortest-remaining-time-first algorithm.
- Answer : (B) The scheduling algorithm works as round robin with quantum time equals to T. After a process's turn comes and it has executed for T units, its waiting time becomes least and its turn comes again after every other process has got the token for T units.

Q-8

- Consider the following table of arrival time and burst time for three processes P0, P1 and P2.

Process	Arrival time	Burst Time
P0	0 ms	9 ms
P1	1 ms	4 ms
P2	2 ms	9 ms

- The pre-emptive shortest job first scheduling algorithm is used. Scheduling is carried out only at arrival or completion of processes. What is the average waiting time for the three processes?
- (A) 5.0 ms (B) 4.33 ms
- (C) 6.33 ms (D) 7.33 ms
- **(A) 5.0 ms**


Thank you