Priority Scheduling

- CPU is allocated to the particular process with the highest priority .
- Priority range be 0 to 7 (say), with 0 representing the highest or the lowest priority
- Priority may depend on internal factors (time limit, memory requirement, number of open files, etc.) and external factors (user, department, etc.)
- May be preemptive or non-preemptive .
- SJF is an important case of priority scheduling, with priority inversely proportional to predicted next CPU burst length.
- May cause starvation, i.e. indefinite blocking of processes
- Aging: gradually increase the priority of a process waiting for a long time
- Priority inversion: a low-priority process gets the priority of a high-priority process waiting for it

Example:

Process	Burst Time	Priority
P1	10	3
P2	1	1
P3	2	4
P4	1	5
P5	5	2

Gantt chart:

p2	p5	p5	p3	p1

0 1 6 16 18 19

AWT = 8.2 mS

Problem with priority scheduling algorithms is indefinite blocking or starvation. A solution to the problem of indefinite blockage of low priority processes is aging. Aging is a technique of gradually increasing the priority of processes that wait in the system for a long time. For example if priorities range from 0 (low) to 127 (high), we could increment the priority of a waiting process by 1 every 15 mins.

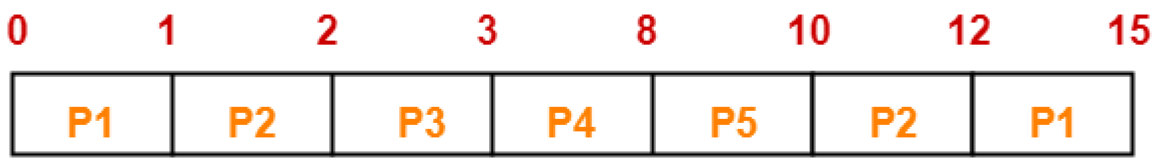
User Problems:

Problem 1:
Consider the set of 5 processes whose arrival time and burst time are given below-

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**

If the CPU scheduling policy is priority preemptive, calculate the average waiting time and average turnaround time. (Higher number represents higher priority)
Solution-

Gantt Chart-



Gantt Chart

Now, we know-

- Turn Around time = Exit time – Arrival time
- Waiting time = Turnaround time – Burst time

Process Id	Exit time	Turn Around time	Waiting time

P1	15	15 – 0 = 15	15 – 4 = 11
P2	12	12 – 1 = 11	11 – 3 = 8
P3	3	3 – 2 = 1	1 – 1 = 0
P4	8	8 – 3 = 5	5 – 5 = 0
P5	10	10 – 4 = 6	6 – 2 = 4

Now,

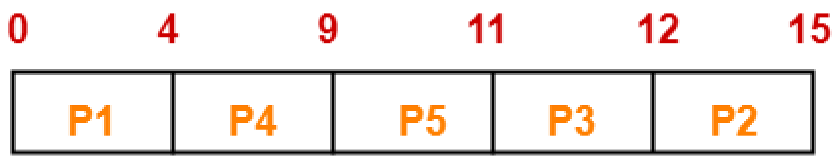
- Average Turnaround time = (15 + 11 + 1 + 5 + 6) / 5 = 38 / 5 = 7.6 unit
- Average waiting time = (11 + 8 + 0 + 0 + 4) / 5 = 23 / 5 = 4.6 unit

Problem 2:
Consider the set of 5 processes whose arrival time and burst time are given below-

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**

If the CPU scheduling policy is priority non-preemptive, calculate the average waiting time and average turnaround time. (Higher number represents higher priority)

Solution-



Gantt Chart

Now, we know-

- Turn Around time = Exit time – Arrival time
- Waiting time = Turnaround time – Burst time

Process Id	Exit time	Turn Around time	Waiting time
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

Now,

- Average Turnaround 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