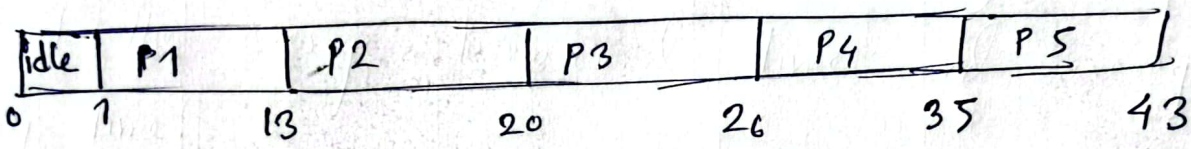
Q. Find the average waiting time and average turn around time for the following jobs by First Come First Severed job scheduling algorithm: [SP-22]

Sl. of jobs	Arrival time	Service time
P1	1	12
P2	7	7
P3	11	6
P4	12	9
P5	17	8

⇨ Gantt chart:



Process	Arrival time	Service time	Completion Time	Turn around Time	Waiting time
P1	1	12	13	12	0
P2	7	7	20	13	6
P3	11	6	26	15	9
P4	12	9	35	23	14
P5	17	8	43	26	18

∴ Average waiting time $= \frac{47}{5} = 9.4$ // Ans

∴ Average turn around time $= \frac{89}{5} = 17.8$ // Ans

Ex. Imagine a busy computer lab with four students waiting to use the printer. Schedule the students using Shortest Remaining Time First (SRTF) Scheduling Algorithm. Calculate the average waiting time. [SP-24]

Student	Arrival Time	Burst Time (Initial)	Burst Time (Remaining)
Alice	10:00 A.M	20 min	20 min
Bob	10:02 A.M	10 min	10 min
Charlie	10:05 A.M	30 min	30 min
David	10:10 A.M	5 min	5 min

⇒

Student	Arrival Time	Burst Time	Completion Time	Turn Around Time	Waiting Time
Alice	10:00 A.M	20 min	35 min	35 min	15 min
Bob	10:02 A.M	10 min	12 min	10 min	2 min
Charlie	10:05 A.M	30 min	65 min	60 min	35 min
David	10:10 A.M	5 min	17 min	7 min	12 min

[T.A.T = Comp − Arrival
W.T = T.A.T − Burst]

Gantt Chart:

Alice	Bob	David	Alice	Charlie

0 2 12 17 35 65

∴ Average waiting time $= \frac{64}{4} = 16$ min // (Ans)

Q1. Consider the following set of processes, their arrival times, and burst times. Use the preemptive Shortest Job First (SJF) scheduling algorithm to find: [MID CT Aut — 24]

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

i) Draw the Gantt Chart for the execution of the processes.

ii) Find the average turnaround time and average waiting time.

⇨ i) Gantt Chart:

P4	P1	P5	P2	P5	P1	P3

0 3 4 5 7 10 15 23

ii)

Process Queue	Burst Time	Arrival Time	Completion Time	Turn Around Time	Waiting Time
P1	6	2	15	13	7
P2	2	5	7	2	0
P3	8	1	23	22	14
P4	3	0	3	3	0
P5	4	4	10	6	2

$\therefore$ Average Turn Around Time $= \frac{46}{5} = 9.2$ (Ans)

$\therefore$ Average Waiting Time $= \frac{23}{5} = 4.6$ (Ans)

四. In the following example, there are five processes named as P1, P2, P3, P4 and P5. Their arrival time and burst time are given below in the table. [Aut – 22]

Process	Arrival Time	Burst Time
P1	1	7
P2	3	3
P3	6	2
P4	7	10
P5	9	8

Find the average waiting time, turn around time and response time using shortest job first scheduling.

⇨

Process	Arrival time	Burst time	Completion time	turn around time	Waiting time	Response Time
P1	1	7	13	12	5	0
P2	3	3	6	3	0	0
P3	6	2	8	2	0	0
P4	7	10	31	24	14	14
P5	9	8	21	12	4	4

Gantt chart:

	P1	P2	P3	P1	P5	P4

0 1 3 6 8 13 21 31

$\therefore$ Average turn around time $= \frac{53}{5} = 10.6$ " (Ans)

$\therefore$ Average waiting time $= \frac{23}{5} = 4.6$ " (Ans)

$\therefore$ Average response time $= \frac{18}{5} = 3.6$ " (Ans)

Q. Find the average waiting time and average turnaround time for the following jobs by Round Robin job scheduling algorithm:

Process Queue	Service time	~~Burst~~ Arrival time
P1	12	1
P2	7	9
P3	1	11
P4	1	12
P5	8	19

[MID CT - Aut 24]

Gantt chart:

~~P1~~	~~P2~~	~~P1~~	~~P3~~	~~P2~~	~~P4~~	~~P1~~	~~P5~~

	P1	P2	P1	P3	P2	P4	P1	P5

0 1 9 11 12 13 18 19 21 ~~29~~ 30

12

Process Queue	Service time	~~Burst~~ Arrival time	Completion time	Turn Around Time	Waiting time
P1	12	1	21	20	8
P2	7	9	18	9	2
P3	1	11	13	2	1
P4	1	12	19	7	6
P5	8	19	29	10	2

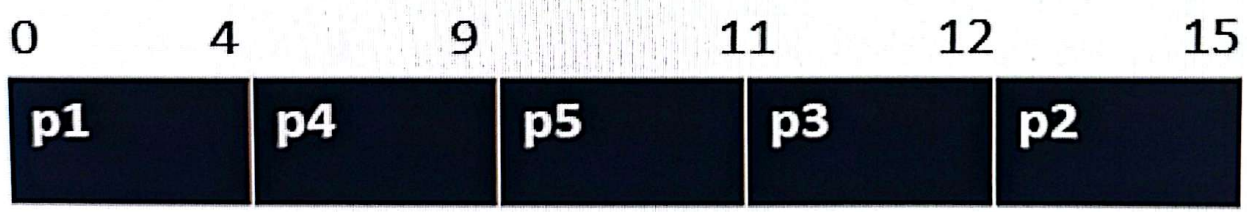
$\therefore$ Average waiting time $= \frac{19}{5} = 3.8$ " (Ans)

$\therefore$ Average turn around time $= \frac{48}{5} = 9.6$ " (Ans)

Priority Scheduling:

Higher number represents higher priority

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



田. Imagine a busy internet cafe with 5 customers:

Alice (Priority 3): Needs to check her email quickly (short burst)

Bob (Priority 2): Working on a presentation (medium burst)

Charlie (Priority 1): Urgent online banking transaction (short burst)

David (Priority 3): Playing an online game (long burst)

Eve (Priority 2): Researching for a project (medium burst)

The scheduling algorithm: Preemptive Priority-based, round robin scheduling algorithm.

Process queue: Initially, all processes are in the queue

(Alice, Bob, Charlie, David, Eve).

CPU Time Slice : 2 seconds

Short Burst : 4 seconds

Medium Burst : 7 seconds [SP-24]

Long Burst : 10 seconds

Now draw the Gantt chart, calculate the average waiting time and CPU utilization.

⇨

Process Queue	Priority	Burst time	Completion time	Turn Around time	Waiting Time
Alice	3	4	24	24	20
Bob	2	7	17	17	10
Charile	1	4	4	4	0
David	3	10	32	32	22
Eve	2	7	18	18	11

Gantt chart:

Charlie	Bob	Eve	Bob	Eve	Bob	Eve	Bob	Eve	Alice	David

0 2 4 6 8 10 12 14 16 17 18 20 22

Alice	David

24 26 28 30 32

$\therefore$ Average waiting time $= \frac{63}{5} = 12.6$ seconds. (Ans)

$\therefore$ CPU utilization $= \frac{\text{Time CPU spent on tasks}}{\text{Total CPU time}} \times 100\%$

$= \frac{32}{32} \times 100\% = 100\%$

[There has no idle time]

III. The following processes are being scheduled using a preemptive priority based round robin scheduling algorithm. [SP-23]

Process	Priority	Burst	Arrival
P1	1	5	0
P2	1	4	0
P3	2	7	6
P4	1	4	10
P5	2	2	11

Each process is assigned a numerical priority, with a lower number indicating a higher relative priority. The length of a time quantum is 2 units. If a process is preempted by a higher-priority process, the preempted process is placed at the end of the queue.

1. Show the scheduling order of the processes using a Gantt chart.
2. What is the average turn around time?
3. What is the average waiting time?
4. What is the CPU utilization rate?

⇨

Process	Priority	Burst	Arrival	Completion Time	Turnaround Time	Waiting Time
P1	1	5	0	9	9	4
P2	1	4	0	8	8	4
P3	2	7	6	23	17	10
P4	1	4	10	14	4	0
P5	2	2	11	18	7	5

1) Gantt chart:

P3

P1	P2	P1	P2	P1	idle	P4	P3	P5	P3

0 2 4 6 8 9 10 14 16 18 23

2) Average turn around time $= \frac{45}{5} = 3$ (Ans)

3) Average waiting time $= \frac{23}{5} = 4.6$ (Ans)

4) The CPU utilization rate $= \frac{\text{Time CPU spent on tasks}}{\text{Total CPU time}} \times 100\%$

$= \frac{22}{23} \times 100\%$

$= 95.65\%$ (Ans)

Q. The following processes are being scheduled using a preemptive round robin scheduling algorithm. [Aut – 22]

Process	Priority	Burst	Arrival
P1	40	20	0
P2	30	25	25
P3	30	25	30
P4	35	15	60
P5	5	10	100
P6	10	10	105

Each process is assigned a numerical priority, with a higher number indicating a higher relative priority. In addition to the processes listed below, the system also has an idle task (which consumes no CPU resources and is identified as 'P_{idle}'). This task has priority 0 and is scheduled whenever the system has no other available processes to run. The length of a time quantum is 10 units. If a process is preempted by a higher-priority process, the preempted process is placed at the end of the queue.

1. Show the scheduling order of the processes using a Gantt chart.

2. What is the turnaround time for each process?
3. What is the waiting time for each process?
4. what is the CPU utilization rate?

⇨

Process	Priority	Burst	Arrival	Completion time	Turnaround Time	Waiting Time
P1	40	20	0	20	20	0
P2	30	25	25	~~55~~ 80	55	30
P3	30	25	30	~~60~~ 90	60	35
P4	35	15	60	~~60~~ 75	15	0
P5	5	10	100	~~90~~ 120	20	10
P6	10	10	105	~~10~~ 115	10	0

1) Gantt chart:

P1	idle	P2	P3	P2	P3	P4	P2	P3	idle	P5	P6	P5

0 20 25 35 45 55 60 75 80 90 100 105 115 120

(Ans)

2) Turn around time, P1 = 20
P2 = 55
P3 = 60
P4 = 15
P5 = 20
P6 = 10 (Ans)

3) Waiting time, P1 = 0
P2 = 30
P3 = 35

P4 = 0

P5 = 10

P6 = 0

" (Ans)

4) The CPU utilization rate = $\frac{\text{Time CPU spent on tacks}}{\text{Total CPU time}} \times 100\%$

$= \frac{105}{120} \times 100\%$

$= 87.5\%$ " (Ans)

Rate-Monotonic Scheduling

A priority is assigned based on the inverse of its period

Shorter periods = higher priority;

Longer periods = lower priority

P_1 is assigned a higher priority than P_2

Example:

p1 = 50 and p2 = 100.

The processing times are t1 = 20 for P1 and t2 = 35 for P2.

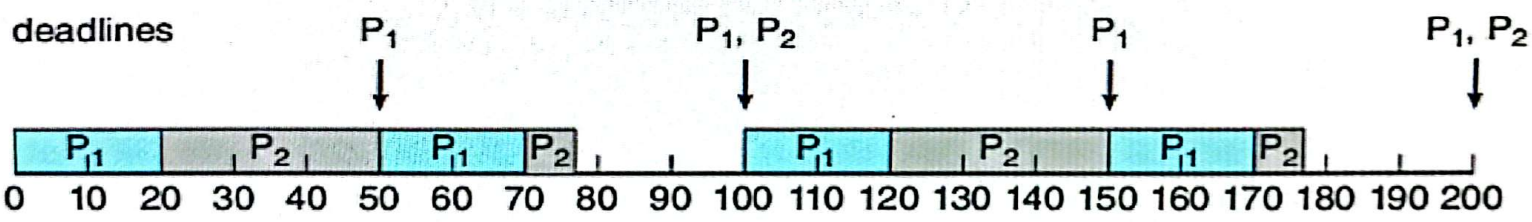


Figure 5.22 Rate-monotonic scheduling.

11. Consider two processes, P1 and P2, where P1 = 50, t_1 = 25, P2 = 75 and t_2 = 30. Can these two processes be scheduled using rate-monotonic scheduling for real time system? Illustrate your answer using gantt chart. [SP-23]

or,

Illustrate the scheduling of these two processes using earliest deadline first (EDF) scheduling.

⇒

Process	Burst time	Period
P1	25	50
P2	30	75

Check schedulability:

using utilization test,

$$U = \frac{t_1}{P_1} + \frac{t_2}{P_2} = \frac{25}{50} + \frac{30}{75} = 0.5 + 0.4$$
$$= 0.9$$

Conditions:

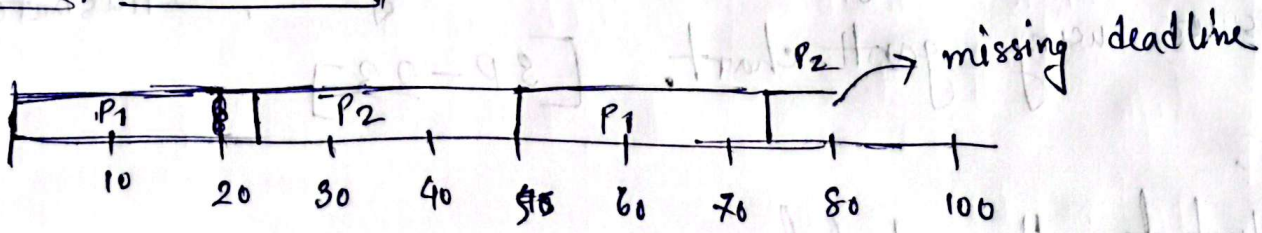
$$U \leq N(2^{1/N} - 1)$$

or, $0.9 \leq 2(2^{1/2} - 1)$

or, $0.9 \leq 0.828$ it's not true

So, Rate-monotonic scheduling cannot guarantee that the processes are scheduable

~~deadline~~: Gantt chart:



P2 misses it's deadline → so rate-monotonic scheduling fails to schedule these two process.

P.T.O

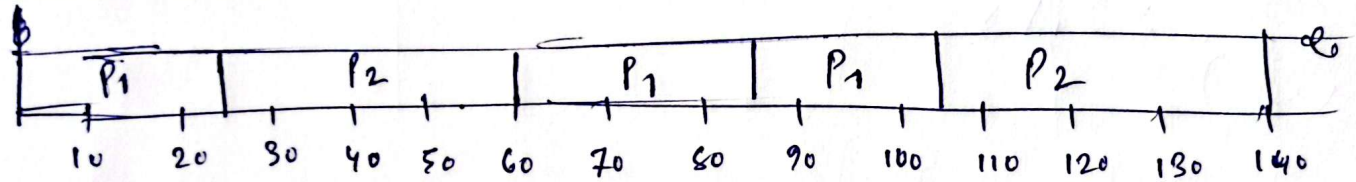
OR

Check schedulability for EDF using utilization test,

$$U = \frac{t_1}{P_1} + \frac{t_2}{P_2} = \frac{25}{50} + \frac{30}{75} = 0.5 + 0.4$$
$$= 0.9 \leq 1.$$

Since $0.9 \leq 1$, so EDF can schedule the processes

Gantt chart:



The traditional UNIX scheduler enforces an inverse relationship between priority numbers and priorities: the higher the number, the lower the priority. The scheduler recalculates process priorities once per second using the following function:

$$\text{Priority} = (\text{recent CPU usage} / 2) + \text{base}$$

where base = 60 and *recent CPU usage* refers to a value indicating how often a process has used the CPU since priorities were last recalculated.

Assume that recent CPU usage for process P_1 is 40, for process P_2 is 18, and for process P_3 is 10. What will be the new priorities for these three processes when priorities are recalculated? Based on this information, does the traditional UNIX scheduler raise or lower the relative priority of a CPU-bound process?

Answer:

The priorities assigned to the processes will be 80, 69, and 65, respectively. The scheduler lowers the relative priority of CPU-bound processes.

Many CPU-scheduling algorithms are parameterized. For example, the RR algorithm requires a parameter to indicate the time slice. Multilevel feedback queues require parameters to define the number of queues, the scheduling algorithms for each queue, the criteria used to move processes between queues, and so on.

These algorithms are thus really sets of algorithms (for example, the set of RR algorithms for all time slices, and so on). One set of algorithms may include another (for example, the FCFS algorithm is the RR algorithm with an infinite time quantum). What (if any) relation holds between the following pairs of algorithm sets?

a. Priority and SJF

b. Multilevel feedback queues and FCFS

c. Priority and FCFS

d. RR and SJF

Answer:

a. The shortest job has the highest priority.

b. The lowest level of MLFQ is FCFS.

c. FCFS gives the highest priority to the job that has been in existence the longest.

d. None.