UNIT-3 (CPU SCHEDULING)

CPU SCHEDULING:- CPU Scheduling is a Process of Determining which Process will own CPU for execution while another Process is own Hold.

PURPOSE OF SCHEDULING ALGORITHM:-

(1) maximum CPU utilization
(2) fare- allocatication of CPU
(3) maximum throughput
(4) minimum waiting time

TYPES OF CPU SCHEDULING ALGORITHM:-

(1) Preemptive Algorithm
(2) Non Preemptive

PREEMPTIVE ALGORITHM

(1) Shortest Remaining time first (SRTF) (SJF) Shortest Job first.
(2) Round Robin (RR) Scheduling
(3) Priority Scheduling

NON-PREEMPTIVE SCHEDULING ALGORITHM:-

(1) First come first serve (FCFS)
(2) SHortest Job first (SJF)
(3) Priority Scheduling.
(4) multilevel queue
(5) Highest Response Ratio Next (HRRN)

PERFORMANCE CRITERIA FOR CPU SCHEDULING :-

(1) Arrival time (A.T)
(2) Burst time (B.T)
(3) Completion time (C.T)
(4) Turn around time (T.A.T)
(5) Waiting time (W.T)
(6) Response time

ARRIVAL TIME (A.T) :- The time at which the process enters into the ready queue is called arrival time.

BURST TIME (B.T) :- The amount of time required by the CPU to execute the process is called Burst time. This does not include waiting time.

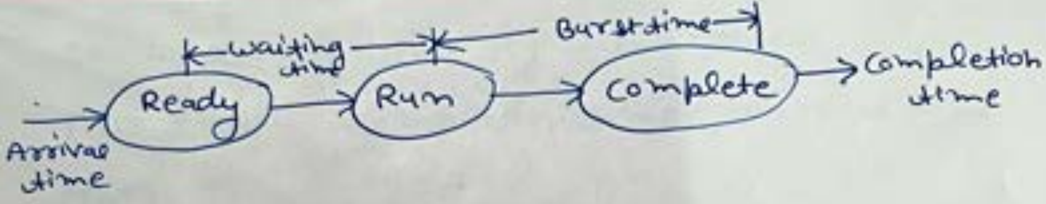
COMPLETION TIME :- The time at which the process completes its execution is called completion time (CT).

TURN-AROUND TIME :- The amount of time spent by the process from its arrival to its completion is called turn-around time (T.A.T)

WAITING TIME :- The amount of time for which the process waits for the CPU to be assigned is called waiting time. (W.T).

RESPONSE TIME:- Response time is the time spent when the Process is in the ready queue and gets the CPU the first time.

VARIOUS TIME RELATED TO THE PROCESS:-



NON-PREEMPTIVE ALGORITHM:-

(1) FIRST COME FIRST SERVE (FCFS) Scheduling Algorithm:-

Ex1 Consider the following Process

Process	Burst time
P_1	7
P_2	5
P_3	6
P_4	3

what is the average waiting time and turn around time for these Process with FCFS Scheduling.

Sol

Process	Burst time	Arrival time
P_1	7	0
P_2	5	0
P_3	6	0
P_4	3	0

Criteria ⇒ Arrival time

mode ⇒ Non-Preemptive

Step -1:

Ganttchart/Barchart

P1	P2	P3	P4
0 – 7	7 – 12	12 – 18	18 – 21

turn around time of all Process.

T.AT (P1) = completion time − Arrival time

(TAT) $P_1 = 7-0$

(TAT) $\boxed{P_1 = 7}$

(T.A.T) P_2 = CT − A.T

(T.A.T) $P_2 = 12-0$

(T.A.T) $\boxed{P_2 = 12}$

(T.A.T) P3 = CT− AT

(T.A.T) $P_3 = 18-0$

(T.A.T) $\boxed{P_3 = 18}$

(T.A.T) P_4 = CT− A.T

(T.A.T) $P_4 = 21-0$

(T.A.T) $\boxed{P_4 = 21}$

Average turn around time(T.A.T) $= \dfrac{P_1+P_2+P_3+P_4}{4}$

$= \dfrac{7+12+18+21}{4}$

$= 58/4 \Rightarrow 14.5$ unit

Waiting time = Turnaround time - Burst time

$W.T(P_1) = 7 - 7$

$\boxed{W.T(P_1) = 0}$

$W.T(P_2) = T.A.T - B.T$

$W.T(P_2) = 12 - 5$

$\boxed{W.T(P_2) = 7}$

$W.T(P_3) = T.AT - B.T$

$W.T(P_3) = 18 - 6$

$\boxed{W.T(P_3) = 12}$

$W.T(P_4) = T.AT - B.T$

$W.T(P_4) = 21 - 3$

$\boxed{W.T(P_4) = 18}$

$$\text{Average waiting time} = \frac{P_1 + P_2 + P_3 + P_4}{4}$$

$$= \frac{0+7+12+18}{4}$$

$$= \frac{37}{4}$$

$$= 9.25 \text{ units}$$

Average waiting time = 9.25 units. Ans

Example-2 Consider the following Process

Process	Burst time
P_1	2
P_2	5
P_3	3
P_4	9
P_5	4

What is the average waiting time and turn around time fore these Process with FCFS Scheduling.
(Do your Self)

FIRST COME FIRST SERVE(FCFS):-

Criteria ⇒ Arrival time
mode ⇒ Non- Preemptive

Process	Average time	Burst time
P_1	2	6
P_2	5	3
P_3	1	8
P_4	0	3
P_5	4	4

calculate average waiting time.

Gantt Chart/ Barchart :-

Step-1

P_4	P_3	P_1	P_5	P_2
0–3	3–11	11–17	17–21	21–24

turn around time of P_1 = completion Time − Arrival time

$T.A.T(P_1) = CT - AT$

$T.A.T(P_1) = 17 - 2$

$\boxed{T.A.T(P_1) = 15}$

$T.A.T(P_2) = CT - AT$

$T.A.T(P_2) = 24 - 5$

$\boxed{T.A.T(P_2) = 19}$

$T.A.T(P_3) = CT - AT$

$T.A.T(P_3) = 11 - 1$

$\boxed{T.A.T(P_3) = 10}$

$T.A.T(P_4) = CT - AT$

$T.A.T(P_4) = 3 - 0$

$\boxed{T.A.T(P_4) = 3}$

$T.A.T(P_5) = CT - AT$

$T.A.T(P_5) = 21 - 4$

$\boxed{T.A.T(P_5) = 17}$

Average turn around time $= \dfrac{P_1 + P_2 + P_3 + P_4 + P_5}{5}$

$= \dfrac{15 + 19 + 10 + 3 + 17}{5}$

$= \dfrac{54}{5} = 10.8$ units

: waiting time = Turned around time - Burst time

$W.T(P_1) = T.A.T - B.T$

$W.T(P_1) = 15 - 6$

$\boxed{W.T(P_1) = 9}$

$W.T(P_2) = T.A.T - B.T$

$W.T(P_2) = 19 - 3$

$\boxed{W.T(P_2) = 16}$

$W.T(P_3) = T.A.T - B.T$

$W.T(P_3) = 10 - 8$

$\boxed{W.T(P_3) = 2}$

$W.T(P_4) = T.A.T - B.T$

$W.T(P_4) = 3 - 3$

$\boxed{W.T(P_4) = 0}$

$W.T(P_5) = T.A.T - B.T$

$W.T(P_5) = 17 - 4$

$\boxed{W.T(P_5) = 13}$

Average waiting time $= \dfrac{P_1 + P_2 + P_3 + P_4 + P_5}{5}$

$= \dfrac{9 + 16 + 2 + 0 + 3}{5}$

$= \dfrac{40}{5} = 8$ units Ans

SHORTEST JOB FIRST (SJF) NON PREMPTIVE SCHEDULING ALGORITHM

Process	Arrival time	Burst time
P_1	2	6
P_2	5	2
P_3	1	8
P_4	0	3
P_5	4	4

Criteria ⇒ Burst-time
mode ⇒ Non-Preemptive

Calculate Avg waiting time and avg turn around time.

Gantt chart :-

P_4	P_1	P_2	P_5	P_3

0   3   9   11   15   23

Turn around time of = Completion time − Arrival time

T.A.T of P_1 = CT − A.T

T.A.T (P_1) = 9 − 2

T.A.T (P_1) = 7

T.A.T of P_2 = CT − A.T

T.AT of (P_2) = 11 − 5

T.A.T (P_2) = 6

Turn around time of P_3 = CT − A.T

$T.A.T(P_3) = 23 - 1$

$\boxed{T.A.T(P_3) = 22}$

$T.A.T(P_4) = CT - AT$

$T.A.T(P_4) = 0 - 0$

$\boxed{T.A.T(P_4) = 0}$

$T.A.T(P_5) = CT - AT$

$T.A.T(P_5) = 15 - 4$

$\boxed{T.A.T(P_5) = 11}$

Average turn Around time = $\dfrac{P_1 + P_2 + P_3 + P_4 + P_5}{5}$

$$= \frac{7 + 6 + 22 + 3 + 11}{5}$$

$$= \frac{13 + 25 + 11}{5}$$

$$= \frac{38 + 11}{5}$$

$$= \frac{49}{5}$$

$$= 9.8 \text{ units}$$

Waiting time = Turnaround time − Burst time

W.T (P1) = T.A.T − B.T

W.T (P1) = 7 − 6

$\boxed{W.T\ (P_1) = 1}$

W.T (P2) = T.A.T − B.T

W.T (P2) = 6 − 2

$\boxed{W.T\ (P_2) = 4}$

W.T (P3) = T.A.T − B.T

W.T (P3) = 22 − 8

$\boxed{W.T\ (P_3) = 14}$

W.T (P4) = T.A.T − B.T

W.T (P4) = 3 − 3

$\boxed{W.T\ (P_4) = 0}$

W.T (P5) = T.A.T − B.T

W.T (P5) = 11 − 4

$\boxed{W.T\ (P_5) = 7}$

Average waiting time $= \dfrac{P_1 + P_2 + P_3 + P_4 + P_5}{5}$

$= \dfrac{1 + 4 + 14 + 0 + 7}{5}$

$= \dfrac{26}{5} = 5.2$ units. Ans

SHORTEST JOB FIRST (SJF) NON PREEMPTIVE:-

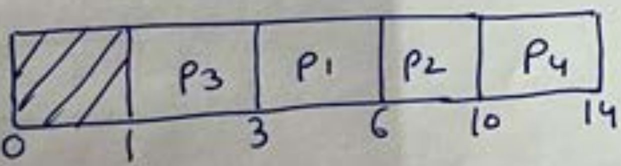
Ex Calculate the Average Turn around time and Average waiting time.

Process	Arrival time	Burst time
P_1	1	3
P_2	2	4
P_3	1	2
P_4	4	1

Criteria ⇒ Burst-time
mode ⇒ Non Preemptive

Step 1:

Gantt Chart:



Turn around timime = completion time − Arrival time

completion time $P_1 = 6$

$CT \; P_2 = 10$

$CT \; P_3 = 3$

$CT \; P_4 = 14$

$T.A.T \,(P_1) = CT - A.T$

$T.A.T \,(P_1) = 6 - 1$

$\boxed{T.A.T \,(P_1) = 5}$

$T.A.T\ (P_2) = CT - A.T$

$T.A.T\ (P_2) = 10 - 2$

$\boxed{T.A.T\ (P_2) = 8}$

$T.A.T\ (P_3) = CT - AT$

$T.A.T\ (P_3) = 3 - 1$

$\boxed{T.A.T\ (P_3) = 2}$

$T.A.T\ (P_4) = CT - AT$

$T.A.T\ (P_4) = 14 - 4$

$\boxed{T.A.T\ (P_4) = 10}$

Average turn around time $= \dfrac{P_1 + P_2 + P_3 + P_4}{4}$

$= \dfrac{5 + 8 + 2 + 10}{4}$

$= \dfrac{25}{4}$

$= 6.25$ units.

Waiting time = turnaround time − Burst time

$W.T\ (P_1) = T.A.T - B.T$

$W.T\ (P_1) = 5 - 3$

$\boxed{W.T\ (P_1) = 2}$

waiting time of P_2 = Turn around time − Burst time

$WT(P_2) = T.A.T - B.T$

$W.T(P_2) = 8 - 4$

$W.T(P_2) = 4$

$W.T(P_3) = T.A.T - B.T$

$W.T(P_3) = 2 - 2$

$W.T(P_3) = 0$

$W.T(P_4) = T.A.T - B.T$

$W.T(P_4) = 10 - 1$

$W.T(P_4) = 9$

$$\text{Average waiting time} = \frac{P_1 + P_2 + P_3 + P_4}{4}$$

$$= \frac{2 + 4 + 0 + 9}{4}$$

$$= \frac{15}{4}$$

$$= 3.75$$ Ans

SJF (Shortest Job First):-

In this algorithm allocates CPU Based on CPU Burst time.

⇒ Process with the smallest burst time will be allocated to the CPU first.

⇒ if two Process is Having the same Burst-time then FCFS (first come - first serve is follow.

⇒ It Also called Shortest - next CPU Burst - algorithm.

PRIORITY SCHEDULING (NON-PREEMPTIV):-

Higher the No - Higher the Priority. Calculate the average W.T & Averg T.AT?

Priority	Process	Arrival time	Burst time
1	P_1	0	5
2	P_2	1	4
3	P_3	2	2
→ 4(H)	P_4	4	1

Criteria ⇒ Priority
mode ⇒ Non-Preemptive

Step 1:

Gantt Chart:

P_1	P_4	P_3	P_2

0 5 6 8 12

Average ~~waiting~~ turnaround time

= Completion time - Arrival time

P_1 completion time = 5

P_2 (C.T) = 12

P_3 (C.T) = 8

P_4 (C.T) = 6

Turnaround time = completion time − Arrival time

T.A.T (P_1) = CT − A.T

T.A.T (P_1) = 5 − 0

$\boxed{T.A.T\ (P_1) = 5}$

T.A.T (P_2) = CT − A.T

T.A.T (P_2) = 12 − 1

$\boxed{T.A.T\ (P_2) = 11}$

T.A.T (P_3) = C.T − A.T

T.A.T (P_3) = 8 − 2

$\boxed{T.A.T\ (P_3) = 6}$

T.A.T (P_4) = C.T − A.T

T.A.T (P_4) = 6 − 4

$\boxed{T.A.T\ (P_4) = 2}$

Average turn-around time = $\frac{P_1 + P_2 + P_3 + P_4}{4}$

$$= \frac{5 + 11 + 6 + 2}{4}$$

$$= \frac{13 + 11}{4}$$

$$= \frac{24}{4}$$

= 6 units.

waiting time = turn around time - Burst time

$W.T(P_1) = T.A.T - B.T$

$W.T(P_1) = 5 - 5$

$\boxed{W.T(P_1) = 0}$

$W.T(P_2) = T.A.T - B.T$

$W.T(P_2) = 11 - 4$

$\boxed{W.T(P_2) = 7}$

$W.T(P_3) = T.A.T - B.T$

$W.T(P_3) = 6 - 2$

$\boxed{W.T(P_3) = 4}$

$W.T(P_4) = T.A.T - B.T$

$W.T(P_4) = 2 - 1$

$\boxed{W.T(P_4) = 1}$

Average waiting time $= \dfrac{P_1 + P_2 + P_3 + P_4}{4}$

$= \dfrac{0 + 7 + 4 + 1}{4}$

$= \dfrac{12}{4}$

$= 3$ units Ans

Ex2 calculate the Average turn around time and waiting time given Lesser the Number Highest Priority.

Process	Priority	Arrival time	Burst time
P_1	→ 2 (H)	0	3
P_2	6	2	5
P_3	3	1	4
P_4	5	4	2
P_5	7	6	9
P_6	4	5	4
P_7	→ 10 (L)	7	10

Step 1:

Gantt Chart:

P_1	P_3	P_6	P_4	P_2	P_5	P_7

0 3 7 11 13 18 27 37

completion time of

P_1 (C.T) = 3

P_2 (C.T) = 18

P_3 (C.T) = 7

P_4 (C.T) = 13

P_5 (C.T) = 27

P_6 (C.T) = 11

P_7 (C.T) = 37

Turn around time = completion time - Arrival time

T.A.T (P_1) = CT - A.T

T.A.T (P_1) = 3 - 0

$\boxed{\text{T.A.T } (P_1) = 3}$

T.A.T (P_2) = C.T - A.T

T.A.T (P_2) = 18 - 2

$\boxed{\text{T.A.T } (P_2) = 16}$

T.A.T (P_3) = C.T - A.T

T.A.T (P_3) = 7 - 1

$\boxed{\text{T.A.T } (P_3) = 6}$

T.A.T (P_4) = C.T - A.T

T.A.T (P_4) = 13 - 4

$\boxed{\text{T.A.T } (P_4) = 9}$

T.A.T (P_5) = C.T - A.T

T.A.T (P_5) = 27 - 6

$\boxed{\text{T.A.T } (P_5) = 21}$

T.A.T (P_6) = C.T - A.T

T.A.T (P_6) = 11-5

$\boxed{\text{T.A.T } (P_6) = 6}$

T.A.T (P_7) = C.T - A.T

T.A.T (P_7) = 37 - 7

$\boxed{\text{T.A.T } (P_7) = 30}$

Average turn around time = $\frac{P_1+P_2+P_3+P_4+P_5+P_6+P_7}{7}$

$= \frac{3+16+6+9+21+6+30}{7}$

$= \frac{25+30+36}{7}$

$= \frac{45+36}{7}$

$= \frac{81}{7}$

$= 11.5$ units

Average waiting time = Turnaround time − Burst time

$WT(P_1) = T.A.T - B.T$

$W.T(P_1) = 3 - 3$

$\boxed{W.T(P_1) = 0}$

$W.T(P_2) = T.A.T - B.T$

$W.T(P_2) = 16 - 5$

$\boxed{W.T(P_2) = 11}$

$WT(P_3) = T.A.T - B.T$

$W.T(P_3) = 6 - 4$

$\boxed{WT(P_3) = 2}$

waiting time of P_4 = T.A.T − B.T

$W.T(P_4) = 9 - 2$

$\boxed{W.T(P_4) = 7}$

$W.T(P_5) = T.A.T - B.T$

$W.T(P_5) = 21 - 9$

$\boxed{W.T(P_5) = 12}$

$W.T(P_6) = T.A.T - B.T$

$W.T(P_6) = 6 - 4$

$\boxed{W.T(P_6) = 2}$

$W.T(P_7) = T.A.T - B.T$

$W.T(P_7) = 30 - 10$

$\boxed{W.T(P_7) = 20}$

Average waiting time $= \dfrac{P_1+P_2+P_3+P_4+P_5+P_6+P_7}{7}$

$= \dfrac{0+11+2+7+12+2+20}{7}$

$= \dfrac{20+34}{7}$

$= \dfrac{54}{7}$

$= 7.7$ units Ans

ROUND- ROBIN SCHEDULING :-

⇒ Round - Robin Scheduling algorithm is a Preemptive algorithm. CPU selects the Process from the ready queue. To implement RR Scheduling ready queue is maintained as a FIFO queue (first in first out) of the Processes. New Processes are added to the tail of the ready queue.

The CPU Scheduler Picks the first Process from the ready queue through the system.

if the time quantum is very short, then short Processes will move through the system relatively quickely. It increases the Processing overhead involved in Handling the clock interrupt and performing the Scheduling and dispatch function.

Ex Let us consider the set of Process with Burst time in milliseconds. All the Processes are arrived at time 0, we can draw the Gantt chart calculate W.T and so on.

Process	Burst time

critieria ⇒ time quantum

mode ⇒ Preemptive

Q4 Round-Robin Preemptive Algorithm:-
time quantum = 2 units
calculate average T.A.T & avg W.T

Process	Arrival time	Burst time	Completion time	T.A.T	W.T
P_1	0	~~3~~ ~~1~~ 0	7	7	4
P_2	1	~~6~~ ~~4~~ ~~2~~ 0	19	18	12
P_3	2	~~2~~ 0	6	4	2
P_4	3	~~2~~ 0	9	5	4
P_5	4	~~5~~ ~~3~~ ~~1~~ 0	22	18	13
P_6	5	~~4~~ ~~2~~ 0	21	15	11

Ready Queue ⇒

~~P1~~	~~P2~~	~~P3~~	~~P1~~	~~P4~~	~~P5~~	~~P2~~	~~P6~~	~~P5~~	~~P2~~	~~P6~~	~~P5~~

Gantt Chart ⇒

P1	P2	P3	P1	P4	P5	P2	P6	P5	P2	P6	P5
0–2	2–4	4–6	6–7	7–9	9–11	11–13	13–15	15–17	17–19	19–21	21–22

completion time of P_1 = 7

P_2 = 19

P_3 = 6

P_4 = 9

P_5 = 22

P_6 = 21

Turn around time = Completion time − Arrival time

T·A·T of (P_1) = 7 − 0

$\boxed{T \cdot A \cdot T\,(P_1) = 7}$

T·A·T of (P_2) = CT − A·T

T·A·T (P_2) = 19 − 1

$\boxed{T \cdot A \cdot T\,(P_2) = 18}$

T·A·T (P_3) = C·T − A·T

T·A·T (P_3) = 6 − 2

$\boxed{T \cdot A \cdot T\,(P_3) = 4}$

T·A·T (P_4) = CT − A·T

T·A·T (P_4) = 9 − 3

$\boxed{T \cdot A \cdot T\,(P_4) = 6}$

T·A·T (P_5) = CT − AT

T·A·T (P_5) = 22 − 4

$\boxed{T \cdot A \cdot T\,(P_5) = 18}$

T·A·T (P_6) = CT − A·T

T·A·T (P_6) = 21 − 5

$\boxed{T \cdot A \cdot T\,(P_6) = 16}$

$$\text{Average turn around time} = \frac{P_1 + P_2 + P_3 + P_4 + P_5 + P_6}{6}$$

$$= \frac{7 + 18 + 4 + 6 + 18 + 16}{6}$$

$$= \frac{25 + 10 + 34}{6}$$

$$= \frac{25 + 44}{6}$$

$$= \frac{69}{6}$$

$$= 11.5 \text{ units.}$$

Waiting time = Turnaround time − Burst time

$W.T(P_1) = T.A.T - B.T$

$WT(P_1) = 7 - 3$

$WT(P_1) = 4$

$W.T(P_2) = T.A.T - B.T$

$W.T(P_2) = 18 - 6$

$W.T(P_2) = 12$

$WT(P_3) = T.A.T - B.T$

$W.T(P_3) = 4 - 2$

$WT(P_3) = 2$

$W.T(P_4) = T.A.T - B.T$

$W.T(P_4) = \cancel{18 - 5}$

$W.T(P_4) = 4 - 2$

$W.T(P_4) = 2$

$W.T(P_5) = T.A.T - B.T$

$W.T(P_5) = 18 - 5$

$= 13$

$W.T(P_6) = T.A.T - B.T$

$W.T(P_6) = 15 - 4$

$W.T(P_6) = 11$

$$\text{Average waiting time} = \frac{P_1 + P_2 + P_3 + P_4 + P_5 + P_6}{6}$$

$$= \frac{4 + 12 + 2 + 4 + 14 + 11}{6}$$

$$= \frac{47}{6}$$

$$= 7.8 \text{ units Ans}$$

Round Robin Algorithm :-

if CPU Scheduling policy is RR mode- preemptive with the time- quantum = 2 units calculate the average waiting time and average turn around time.

Process	Arrival time	B.T	C.T	T.A.T	W.T	R.T
P_1	0	~~5~~ ~~3~~ ~~1~~ 0	13	13	8	0
P_2	1	~~3~~ ~~1~~ 0	12	11	8	2
P_3	2	~~1~~ 0	5	3	2	4
P_4	3	~~2~~ 0	9	6	4	7
P_5	4	~~3~~ ~~1~~ 0	14	10	7	9

Ready queue :-

P_1	P_2	P_3	P_1	P_4	P_5	P_2	P_1	P_5

Gantt Chart :⇒

P_1	P_2	P_3	P_1	P_4	P_5	P_2	P_1	P_5

0 2 4 5 7 9 11 12 13 14

Completion time of P_1 = 13

$C.T\ (P_2) = 12$

$C.T\ (P_3) = 5$

$C.T\ (P_4) = 9$

$C.T\ (P_5) = 14$

Turn around time = Completion time − Arrival time

$T.A.T\ (P_1) = C.T - A.T$

$T.A.T\ (P_1) = 13 - 0$

$\boxed{T.A.T\ (P_1) = 13}$

$T.A.T\,(P_2) = C.T - A.T$

$T.A.T\,(P_2) = 12 - 1$

$\boxed{T.A.T\,(P_2) = 11}$

$T.A.T\,(P_3) = C.T - A.T$

$T.A.T\,(P_3) = 5 - 2$

$\boxed{T.A.T\,(P_3) = 3}$

$T.A.T\,(P_4) = C.T - A.T$

$T.A.T\,(P_4) = 9 - 3$

$\boxed{T.A.T\,(P_4) = 6}$

$T.A.T\,(P_5) = C.T - A.T$

$T.A.T\,(P_5) = 14 - 4$

$\boxed{T.A.T\,(P_5) = 10}$

Average Turn around time = $\dfrac{P_1 + P_2 + P_3 + P_4 + P_5}{5}$

$$= \frac{13 + 11 + 3 + 6 + 10}{5}$$

$$= \frac{42}{8}$$

$= 8.4$ units Ans

A waiting time = Turnaround time − Burst time

W.T of (P_1) = T.A.T − B.T

W.T (P_1) = 13 − 5

$\boxed{\text{W.T}(P_1) = 8}$

W.T of (P_2) = T.A.T − B.T

W.T of (P_2) = 11 − 3

$\boxed{\text{W.T}(P_2) = 8}$

W.T (P_3) = T.A.T − B.T

W.T (P_3) = 3 − 1

$\boxed{\text{W.T}(P_3) = 2}$

W.T of (P_4) = T.A.T − B.T

W.T of (P_4) = 6 − 2

$\boxed{\text{W.T of}(P_4) = 4}$

W.T of (P_5) = T.A.T − B.T

W.T (P_5) = 10 − 3

$\boxed{\text{W.T}(P_5) = 7}$

Average waiting time $= \dfrac{P_1 + P_2 + P_3 + P_4 + P_5}{5}$

$= \dfrac{8 + 8 + 2 + 4 + 7}{5}$

$= \dfrac{29}{5} = 5.8$ units Ans

PRIORITY SCHEDULING (Preemptive):

Ex consider the set of 5 Processes whose arrival time and burst time are given below: if the CPU Scheduling Policy is Preemptive Priority. Calculate the average waiting time. (Higher Priority number represents Higher Priority).

Process	Arrival time	BURST TIME	Priority
P_1	0	~~4~~ 3	2
P_2	1	~~3~~ ~~2~~ 0	3
P_3	2	~~1~~ 0	4
P_4	3	~~5~~ 4	5 ← (H)
P_5	4	~~2~~ 1	5

Gantt Chart ⇒

P_1	P_2	P_3	P_4	P_5	P_2	P_1
0–1	1–2	2–3	3–8	8–10	10–12	12–15

Ready Queue:

~~P_1~~	~~P_2~~	P_3	P_4	

$P_1 = 0 +$

Completion time of $P_1 = 15$

C.T of $P_2 = 12$

C.T of $P_3 = 3$

C.T of $P_4 = 8$

turn around time = completion time − Arrival time

T.A.T of P_1 = C.T − A.T

= 15 − 0

$\boxed{\text{T.A.T } (P_1) = 15}$

T.A.T of P_2 = C.T − A.T

T.A.T (P_2) = 12 − 1

$\boxed{\text{T.A.T } (P_2) = 11}$

T.A.T of P_3 = C.T − A.T

T.A.T of (P_3) = 8 − 2

$\boxed{\text{T.A.T } (P_3) = 6}$

T.A.T of (P_4) = C.T − A.T

T.A.T of (P_4) = 8 − 3

$\boxed{\text{T.A.T } (P_4) = 5}$

T.A.T of (P_5) = C.T − A.T

T.A.T (P_5) = 10 − 4

$\boxed{\text{T.A.T } (P_5) = 6}$

waiting time = turnaround time − Burst time

w.T of (P_1) = T.A.T − B.T

w.T (P_1) = 15 − 4

w.T (P_1) = 11

w.T of (P_2) = T.A.T − B.T

w.T (P_2) = 11 − 2

w.T (P_2) = 9

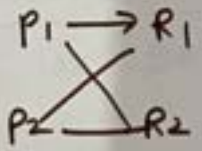
w.T of (P_3) = T.A.T − B.T

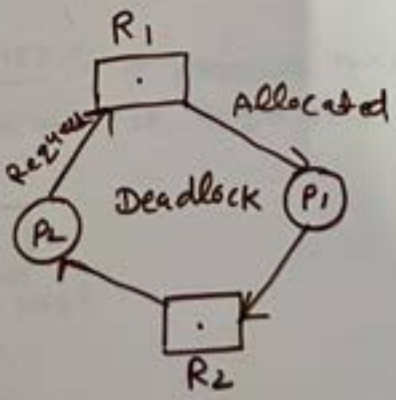
w.T (P_3) = 8 − 2

w.T (P_3) =

DEADLOCK :- A Deadlock is a Situation where a group of Processes are permanently blocked as a result of each Process having acquired a subset of the resources needed for its completion and waiting for release of the remaining resource held by others in the same group.

Deadlock is a state Between two or more Resources in which they are completing for each other Resources for completing their tasks. and no one is Ready to leave Resources that means waiting for that which cannot occur.





DEADLOCK CHARACTERIZATION :-

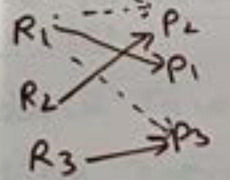
In a deadlock Processes never finish executing and System resources are tied-up preventing other Jobs from Starting.

NECESSARY CONDITIONS FOR DEADLOCK :-

Deadlock can arise if four conditions hold

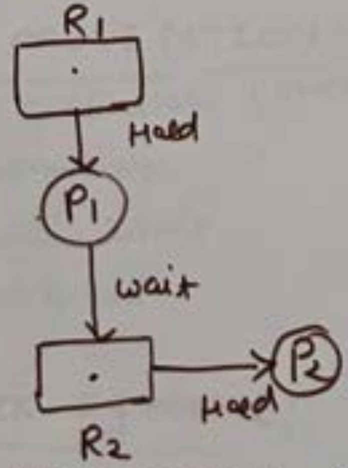
(1) Mutual exclusion
(2) Hold and wait
(3) No-Preemption
(4) circular wait

(1) MUTUAL EXCLUSION :-
only one Process may use a resource at a time. once a Process has been allocated a Particular resource, It has exclusive use of the resource. No other Process can use a resource while it is allocated to a Process.



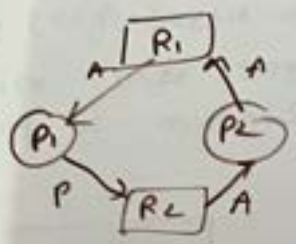
one-to-one

(2) HOLD AND WAIT :- A Process may Hold a resoure at the same time it requests another one.



(3) NO PRE-EMPTION :- No Resource can be forcibly removed from a Process Holding it. Resources can be released only one by the explicit action of the Process, rather than by the action of an external authority.

(4) CIRCULAR WAIT :- All the Processes must be waiting for the resource in a cyclic manner.



DEADLOCK HANDLING METHOD AND DEADLOCK PREVENTION :-

⇒ Deadlock Ignorance

⇒ Deadlock Prevention

⇒ Deadlock avoidance (Banker's Algo)

⇒ Deadlock detection and recovery. (RAG)

DEADLOCK PREVENTION :-

(1) Mutual Exclusion (Shared Resoure

(2) No Preemption

(3) Hold and wait

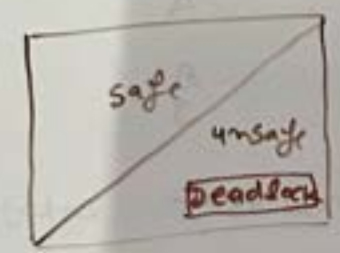
(4) circular wait.

(1) Deadlock Ignorance :- we Ignore the Problem as if it does not exist.

(2) Deadlock Prevention :-

DEADLOCK AVOIDANCE :-

(i) Safe State :- A State is safe if the system can allocate Resource upto its maximum in some sequence (safe sequence) and still Avoid Dead lock. Safe sequence may not be unique there may be multible safe sequence.



Safety Algorithm:

(i) work = Available
finish = false

(ii) Find an "i" such that
Finish [i] = false and
$Need_i \leq$ work
if no such i, go to step 4

(iii) Finish [i] = true
work = Work + Allocation

(iv) if Finish [i] = true for all i
then the system is safe.

⇒ A state is said to be safe if it is not deadlocked and there is some scheduling order in which every process can run to completion even if them suddenly request their maximum number of resource immediately.

ex

Process	Allocated	maximum
A	3	9
B	2	4
C	2	7

total resources are 10

7 resources already allocated

So there are three resou are free

Free Resource = 10 − 7

= 3

A need 6 resources more to complet it

Need = maximum − Allocation

= 9 − 3

= 6

B need = 4 − 2

= 2 Resources

C need = 7 − 2 = 5 resources more to complete

Process	Allocate	max	
A	3	9	9-3 = 6
B	2	4	4-2 = $\boxed{2}$
C	2	7	7-2 = 5

Free Resource = 3

Process	Allocd	max	
A	3	9	9-3=6
B	~~2~~	~~4~~ -	-
C	2	7	7-2=5

Free Resource = 5

Process	Allocated	max	
A	3	9	9-3=6
B	0	-	
C	0	-	

free resource = 7

Process	Allocated	max
A	0	-
B	0	-
C	0	-

free = 1

Safe sequence

<B, C, A>

BANKER'S ALGORITHM:-

⇒ The Banker's algorithm is the best known of the avoidance strategies. The strategy is modelled after the leading policies employed in banking system. The resource-allocation graph algorithm is suitable to a resource allocation system with single instances of each resource type.

⇒ Banker's Algorithm is suitable to a resource allocation system with multiple instances of each resource type. Algorithm makes decisions on granting a resource based on whether or not granting the request will put the system into an unsafe state.

⇒ Several data structure must be maintained to implement the Banker's algorithm. Let n be the number of processes in the system and m be the number of resource types. We need the following data-structure.

(1) Available
(2) max
(3) Allocation
(4) Need.

(1) Available:- A vector of length m indicates the number of available resources of each type. if available [j] = k there are k instances of resource type R_j available.

(2) max:- An n×m matrix defines the maximum demand of each process. if max [i, j] = k then process P_i may request at most k instances of resource type R_j.

(3) Allocation:- An n×m matrix defines the number of resources of each type currently allocated to each process. if allocation [i, j] = k, then process P_i is currently allocated k instances of resource type R_j.

(4) Need:- An n×m matrix indicates the remaining resource need of each process. if need [i, j] = k, then process P_i may need k more instances of resource type R_j to complete its task. Need[i j] - Allocation [i, j]

ALGORITHM:-

Bankers Algorithm is the combination of Safety Algorithm and the resource request Algorithm

Step1: Initialize work = available
finish [i] = false for i = 0, 1, 2, 3 ... n-1

Step2:

Check the availability
Need [i] <= work go to Step 3
else finish [i] = = false if i does not exist go to Step-4

Step-3: work = work + Allocation [i]
finish [i] = true then go to Step 2
~~else finish [i] = false if i~~

Step-4: if finish [i] = = true for all
System is safe state

$\boxed{\text{Need} = \text{Maximum} - \text{Allocation}}$

Example:
Consider a system that contain five process P_1, P_2, P_3, P_4, P_5, and three types A, B & C instance. A has 10, B has 5 and C has 7

Process	Allocation A B C	Maximum A B C	Available A B C	Need A B C
P_1	0 1 0	7 5 3	3 3 2	7 4 3
P_2	2 0 0	3 2 2		1 2 2
P_3	3 0 2	9 0 2		6 0 0
P_4	2 1 1	2 1 1		0 0 0
P_5	0 0 2	4 3 3		4 3 1

Step 1:

Available work = 332

$Need_{(1)} \leq work$

$743 \leq 332$ P_1 is not executed.

$Need_2 \leq work$

$122 \leq 332$ P_2 is executed

work = work + Allocation of P_2

work = 332 + 200

work = 532

$Need_3 \leq work$

$600 \leq 532$ false P_3 waiting

$Need_4 \leq work$

$000 \leq 532$ true P_4 executed.

New work = work + Allocation of P_4

= 532 + 211

= 743

$Need_5 \leq work$

431 ≤ 743 true P_5 executed.

$Need_1 \leq work$

743 ≤ 745 true.

work = work + Allocation of P_1

work = 745 + 010

= 755

$Need_3 \leq work$

600 ≤ 755 true P_3 is executed

new work = work + Allocation of P_3

new work = 755 + 302

= 1057

Safe Sequece

< P_2, P_4, P_5, P_1, P_3 > Ans

Ex2

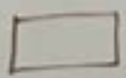
Process	Allocation A B C D	maximum A B C D	Available A B C D	Need A B C D
P_0	0 0 1 2	0 0 1 2	1 5 2 0	0 0 0 0
P_1	1 0 0 0	1 7 5 0	1 5 3 2	0 7 5 0
P_2	1 3 5 4	2 3 5 6	2 8 8 6	1 0 0 2
P_3	0 6 3 2	0 6 5 2	2 14 11 8	0 0 2 0
P_4	0 0 1 4	0 6 5 6	2 14 12 12	0 6 4 2
			3 14 12 12	

Do yourself

$< P_0, P_2, P_3, P_4, P_1 >$

RESOURCES ALLOCATION GRAPH:-

 ⇒ Process

 ⇒ Resources

 ⇒ Number of Instances

 ⇒ Request and Allocated arrow Line