CHIRAG MISHRA
2021201106

Operating system

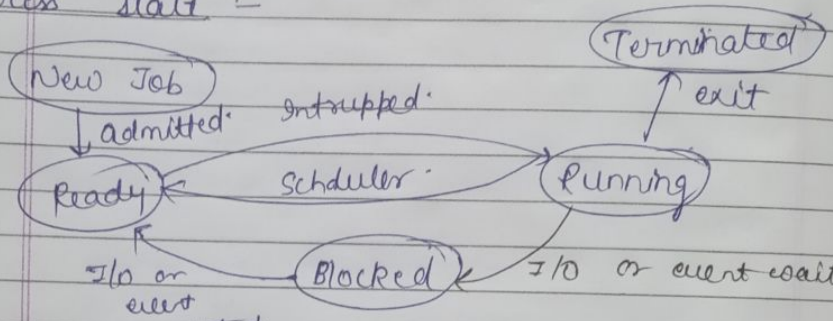
Date:
Page:



Solution : 2

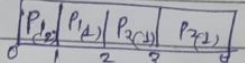
| NO | Arrival time. | CPU B.T. | I/O/Burst) W.T. | C.PU B.T. |
|---|---|---|---|---|
| P1 | 0 | 2 | 2 | 2 |
| P2 | 2 | 2 | 4 | 1 |

Process state :-

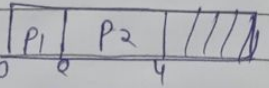



Here we use SJF (Shortest Job first algorithm) preemtive nature -

| No. | Arrival time (A.T) | CPU B.T. (B.T) |
|---|---|---|
| P1 | 0 | 2 |
| P2 | 2 | 2 |




Ready queue




Gant chart.

Turned around time = Completion Time - Arrival time

Turned around time for $P_1 \Rightarrow 2 - 0 = 2$

Turned around time for $P_2 \Rightarrow 4 - 2 = 2$ { Completion time - Arrival time

Date:
Page:

Waiting time = Turned around time − CPU Burst time.
(C.B-T)

Waiting time for $P_1$ = 2 − 2 = (0)

Waiting time for $P_2$ = 2 − 2 = (0)

Here there is no waiting time.

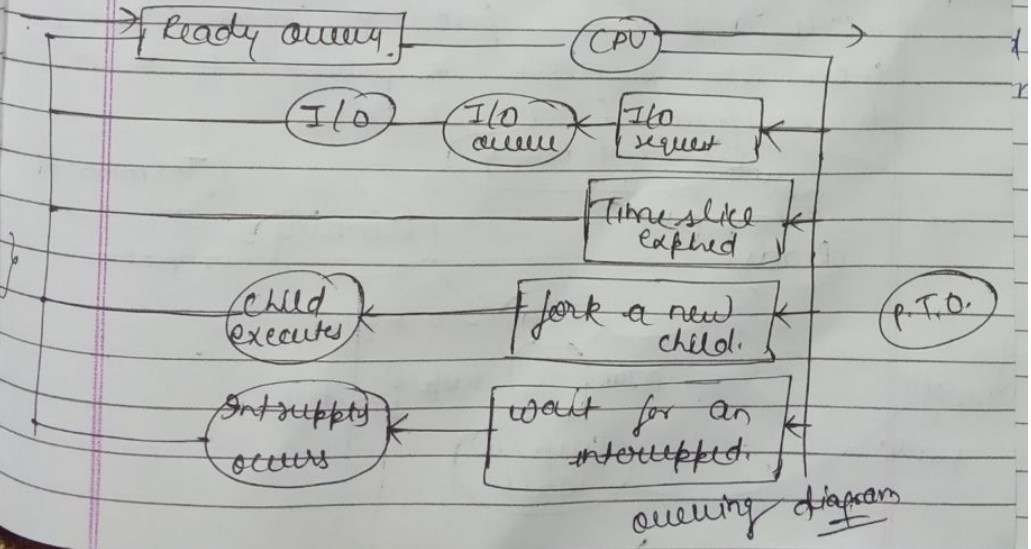
Now take one more statement :— Algorithm :-

Once process is allocated CPU, one of the several events occur -

(i) Process can issue an I/O request and then be placed in I/O queue.

(ii) The process could create a new sub process and wait for the subprocess termination.

(iii) The process could be removed forcefully from CPU, as a result of an interrupt and put back in ready que.




queuing diagram

Date:
Page:

actually what happens in CPU/I/O burst cycle-

load store
add store
read from file.
} CPU burst

[wait for I/O] } I/O burst

store increment index
write to file
} CPU burst

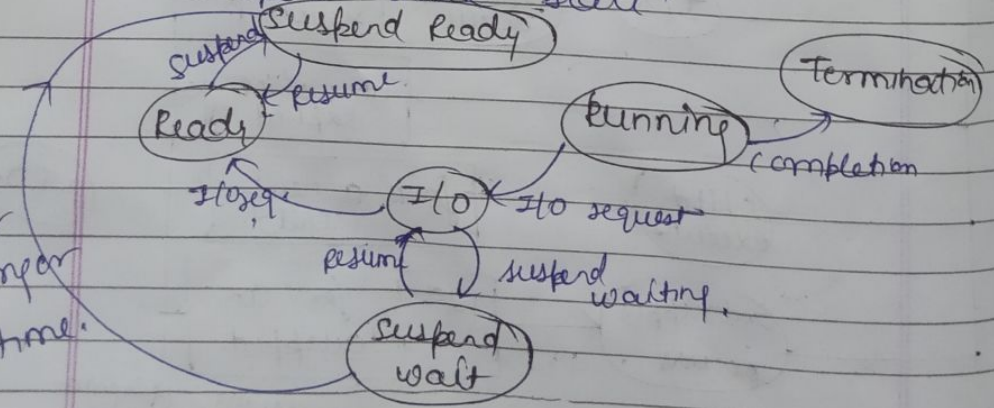
[wait for I/O] } I/O burst

load store
add store
read from file
} CPU burst

[wait for CPU] } I/O burst

suspend ready - If a process is in ready state and a more priority process arrives then the previous process is send to suspend ready state.

suspend wait - If a process is in wait or block state and more priority process arrives then the previous is process is sent to suspend block state.



Date:
Page:

What is happing here in operating system:-

(i) first. when arrival time is 0, process $P_1$ come to ready state for execution, and got CPU.

(ii) It executes at 1 st sec and then it take 2 sec to complete Process bec. its CPU burst time is 2 after that.

(iii) when at 2 sec. $P_2$ process came and after execution of $P_1$, $P_1$ go to I/O. and $P_2$ come to ready que.

(iv) $P_1$ after I/O go to suspend wait and when I/O is free then it again go to I/O and after completion it will go to termination.

(v) and $P_2$ is in ready que at 2 sec so it starts executing and it takes 2 sec to complete process and then go to I/O.

(vi) if I/O is busy then go suspend wait until I/O is free.

(vii) when I/O is free, it jump from suspend wait to I/O and then go to Terminate on state.

P.T.O

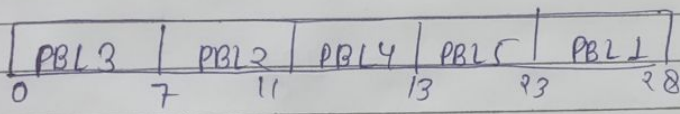
Solution → 1

Date:
Page:

(1) (a)

| project Name | Burst time. |
|---|---|
| PBL 1 | 5 |
| PBL 2 | 4 |
| PBL 3 | 7 |
| PBL 4 | 2 |
| PBL 5 | 10 |

| project Name | AT | BT | CT | TAT |
|---|---|---|---|---|
| PBL 1 | 0 | 5 | 20 | 20 |
| PBL 2 | 0 | 4 | 11 | 11 |
| PBL 3 | 0 | 7 | 7 | 7 |
| PBL 4 | 0 | 2 | 13 | 13 |
| PBL 5 | 0 | 10 | 23 | 23 |



Gantt chart.

TAT = CT - AT

avg. TAT = 20 + 11 + 7 + 13 + 23 / 5

= (16.4) days

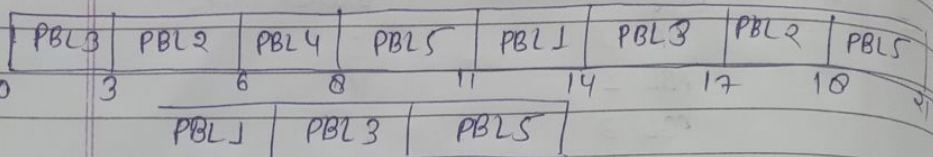
(1) (b) Time slice = 3.

| Project | (CBT) |
|---|---|
| PBL 1 | 5 |
| PBL 2 | 4 |
| PBL 3 | 7 |
| PBL 4 | 2 |
| PBL 5 | 10 |

P.T.O

Date:
Page:

Gant Chart




| | (CBT) | | TAT. |
|---|---|---|---|
| PBL3 | 7 | -4-1 | 24 |
| PBL2 | 4 | -1 | 18 |
| PBL4 | 2 | -0 | 0 |
| PBL5 | 10 | -7-4 | 20 |
| PBL1 | 5 | -2 | 23 |

Total avg = $\frac{24+18+0+20+23}{5}$

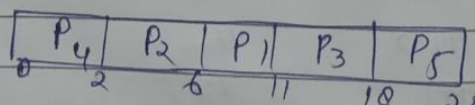
$= \frac{101}{5}$

$= (20.2)$ Ans

(1) (c)

| NO. | AT | BT | CT | TAT. |
|---|---|---|---|---|
| P1 | 0 | 5 | 11 | 11 |
| P2 | 0 | 4 | 6 | 6 |
| P3 | 0 | 7 | 18 | 18 |
| P4 | 0 | 2 | 2 | 2 |
| P5 | 0 | 10 | 28 | 28 |

Gant Chart,




Total TAT = 65

No. of Job = 5

avg. TAT = $65/5 = (13)$