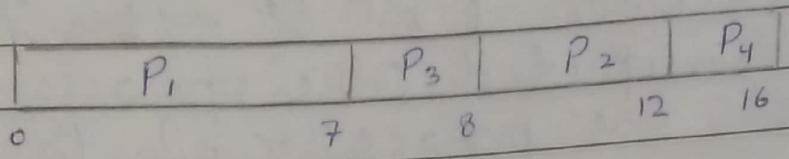
Q4

Date:

Non-Preemptive :-

a) Gantt Chart :-



b) response time:

$P_1 = 0$, $P_2 = 6$, $P_3 = 3$, $P_4 = 7$

waiting time:

$P_1 = 8$ $P_2 = 6$, $P_3 = 3$, $P_4 = 7$

Average waiting time:

$$\frac{0+6+3+7}{4} = 4$$

Turnaround time:

$P_1 = 0+7 = 7$

$P_2 = 6+4 = 10$

$P_3 = 3+1 = 4$

$P_4 = 7+4 = 11$

c). Since P_1 arrive at 0, it will immediately get processor.

- P_2 arrives at 2 and get CPU at 8, so response time is $8-2 = 6$ ms.
- P_3 arrives at 4 & gets CPU at 7 so its response time is 3
- P_4 arrives at 5, get CPU at 12.

Scanned with CamScanner

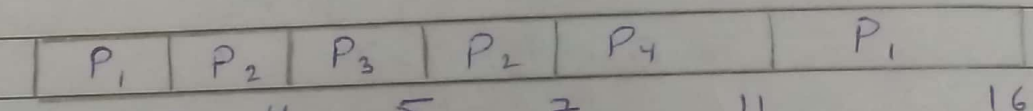
Date:

Conclusion:-

The processes having shortest burst time will get the processor first & after their execution is completed. Other process will get the processor. In this manner, average waiting & turnaround time is improved.

Preemptive:-

a)



b) response time:

$P_1 = 0$, $P_2 = 0$, $P_3 = 0$, $P_4 = 2$

Waiting time:

$P_1 = 9$, $P_2 = 1$, $P_3 = 0$, $P_4 = 2$

Turnaround time:

$P_1 = 9 + 7 = 16$

$P_2 = 1 + 4 = 5$

$P_3 = 0 + 1 = 1$

$P_4 = 2 + 4 = 6$

Average waiting time:

$9 + 1 + 0 + 2 / 4 = 3\text{ms}$

Scanned with CamScanner

Date:

c) Since P_1 was the first to arrive it got CPU.

- P_1, P_2 was given CPU immediately
- P_3 arrive at 4, & since it was shorter than P_2 & P_1, it get the CPU & completes it execution at 5
- P_4 arrived at 5. Among P_1, P_2, P_4, P_2 was shortest. It got the processor again.
- P_4 got CPU & completed execution & P_1 get the CPU at last.

Conclusion:

Preemptive proved to reduce starvating, the waiting time is improved.

Scanned with CamScanner