CPU Scheduling Exercises

NOTE: All time in these exercises are in msec.

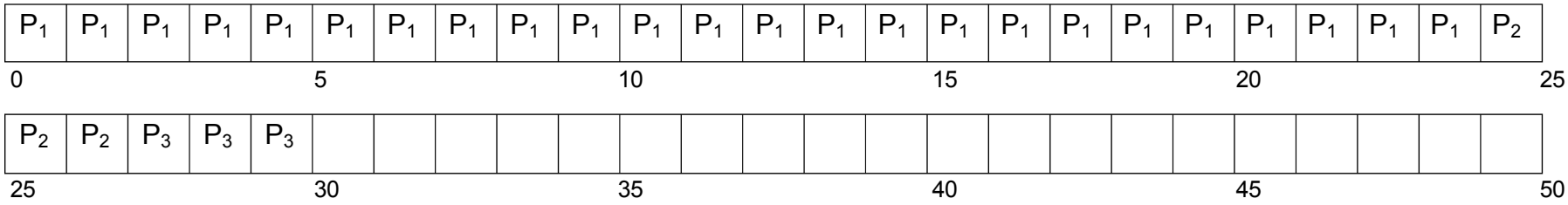
FCFS: The process that request the CPU first is allocated the CPU first.

Processes P_1, P_2, P_3 arrive at the same time, but enter the job queue in the order presented in the table.

Process	Arrive Time	Burst Time	Waiting Time
P_1	0	24	0
P_2	0	3	24
P_3	0	3	27

Time	Queue
0	**P_1**, P_2, P_3
24	**P_2**, P_3
27	**P_3**

Gantt chart:



Average waiting time:

(0 + 24 + 27) / 3 = 17 msec

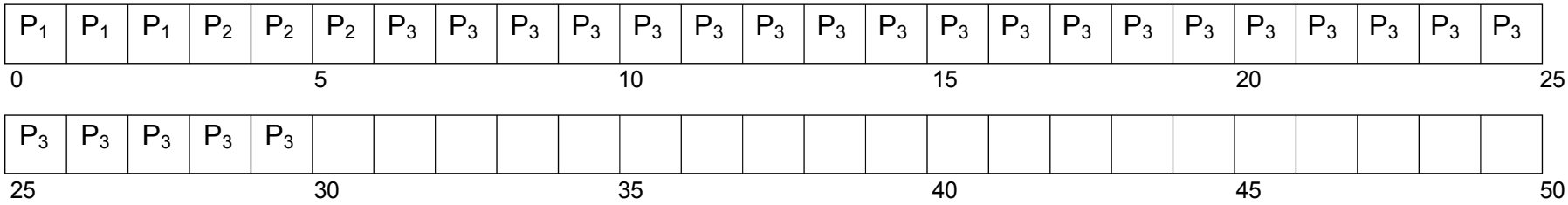
FCFS: The process that request the CPU first is allocated the CPU first.

Processes P_1, P_2, P_3 arrive at the same time, but enter the job queue in the order presented in the table.

Process	Arrive Time	Burst Time	Waiting Time
P_1	0	3	0
P_2	0	3	3
P_3	0	24	6

Time	Queue
0	**P_1**, P_2, P_3
3	**P_2**, P_3
6	**P_3**

Gantt chart:



Average waiting time:

(0 + 3 + 6) / 3 = 3 msec

Compare this example with the previous example, what do you observe?

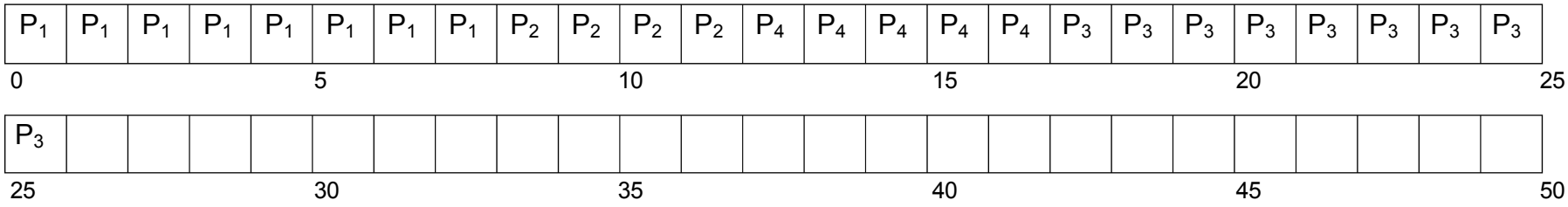
The average waiting time reduced substantially. The average waiting time under a FCFS policy is generally not minimal, and may vary substantially according to the order in the queue and the CPU-burst time.

Shortest Job First: When the CPU is available, it is assigned to the process that has the smallest CPU burst.

Process	Arrive Time	Burst Time	Waiting Time
P_1	0	8	8 – 8 = 0
P_2	1	4	12 – 4 – 1 = 7
P_3	2	9	26 – 9 – 2 = 15
P_4	3	5	17 – 5 – 3 = 9

Time	Queue
0	**P_1 = 8**
8	**P_2 = 4**, P_3 = 9, P_4 = 5
12	P_3 = 9, **P_4 = 5**
17	**P_3 = 9**

Gantt chart:



Average waiting time:

(0 + 7 + 15 + 9) / 4 = 7.75 msec

Shortest Remaining Time First : A preemptive version of the shortest job first algorithm.

Note: We use FCFS to break the tie when comparing the remaining time of the processes.

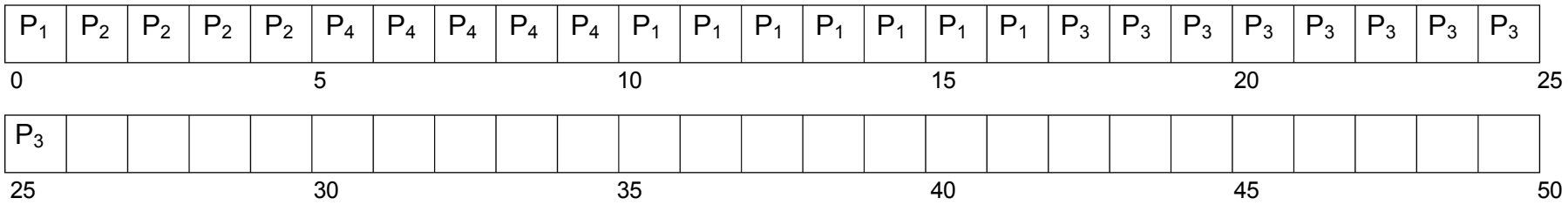
Process	Arrive Time	Burst Time	Waiting Time
P_1	0	8	17 – 8 = 9
P_2	1	4	5 – 4 – 1 = 0
P_3	2	9	26 – 9 – 2 = 15
P_4	3	5	10 – 5 – 3 = 2

Time	Queue
0	**P_1 = 8**
1	P_1 = 7, **P_2 = 4**
2	P_1 = 7, **P_2 = 3**, P_3 = 9
3	P_1 = 7, **P_2 = 2**, P_3 = 9, P_4 = 5
5	P_1 = 7, P_3 = 9, **P_4 = 5**
10	**P_1 = 7**, P_3 = 9
17	**P_3 = 9**

Average waiting time:

(9 + 0 + 15 + 2) / 4 = 6.5 msec

Gantt chart:



Compare this example with the result from the SRTF example, what do you observe?

Compare to the SJF example, we can achieve better average waiting time. Again, we see the benefits of preemptive scheduling.