

## Single-Core CPU Scheduling Exercises

1). Consider the following set of processes with their *arrival time*, *CPU burst time*, and *priority* details:

Process	Arrival Time (ms)	Burst (ms)	Priority
$P_1$	0	10	3
$P_2$	1	4	4
$P_3$	2	8	1
$P_4$	3	6	2

Draw the *Gantt chart* and show the *average waiting time* for the execution of these processes using the following scheduling algorithms:

1.1) First-Come First-Served (FCFS)

1.2) Shortest-Job-First (SJF)

1.3) Priority

1.4) Round-Robin (RR) (time quantum = 4).

2). Consider the following set of processes with their *arrival time*, *CPU burst time*, and *priority* details:

Process	Arrival Time (ms)	CPU Burst (ms)	Priority
$P_1$	0	8	4
$P_2$	1	4	1
$P_3$	2	12	3
$P_4$	3	3	2

Draw the *Gantt chart*, and show the *average waiting time* of the processes based on the following scheduling algorithms:

2.1) First-Come First-Served (FCFS)

2.2) Shortest-Job-First (SJF)

2.3) Priority

2.4) Round-Robin (time quantum = 3).

3). Consider the following set of processes with their *arrival time*, *CPU burst time*, and *priority* details:

Process	Arrival Time (ms)	CPU Burst (ms)	Priority
$P_1$	0	14	3
$P_2$	1	8	1
$P_3$	2	6	4
$P_4$	3	4	2

Draw the *Gantt chart* and show the *average waiting time* of the processes based on the following scheduling algorithms:

3.1) First-Come First-Served (FCFS)

3.2) Shortest-Job-First (SJF)

3.3) Priority

3.4) Round-Robin (time quantum = 4).

4). Explain the difference between **preemptive** and **non-preemptive** scheduling with examples.

5). Why is it important for the scheduler to distinguish **I/O-bound** programs from **CPU-bound** programs?