SJF

(Shortest Job First)

1) When the CPU is available it is assigned to the process that has the smallest next CPU burst
2) If the next CPU burst of two processes are same, FCFS scheduling is used to break the tie
3) Non-preemptive scheduling, sometimes called non-preemptive SJF
4) It also has preemptive version which is called shortest remaining time first (SRTF) scheduling

SJF is optimal-gives minimum average waiting time for a given set of process

Frequently used in long term scheduler

Non-preemptive

Process	Burst time
P1	6
P2	8
P3	7
P4	3

As arrival time is not given, we are supposing every process has arrived at same time

Gantt chart

P4	P1	P3	P2

0 3 9 16 24

As p4 has the shortest burst time it has assigned to CPU first then p1, p3 and p2

Process	Burst time	Arrival time
P1	8	0
P2	4	1
P3	9	2
P4	5	3

**Here arrival time is given

Gantt chart

P1	P2	P4	P3

0 8 12 17 26

The arrival time of p1 is 0ms so CPU has given to p1, after 1ms p2 has arrived and it’s burst time is 4ms which is less than the remaining burst time of p1 (after 1ms p1’s burst time is 7ms) but as non-preemptive SJF does not support context switching p1 will complete its execution.

After completion of p1 8ms has passed and p2, p3 and p4 is in the queue (arrived at 1ms, 2ms and 3ms) so the process which has the lowest burst time will be assigned to CPU

Which is p2(4ms) then p4(5ms) and then p3(9ms)

WT for p1 = 0

WT for p2 = (8 - 1) = 7ms

WT for p3 = (17 - 2) = 15ms

WT for p4 = (12 - 3) = 9ms

TAT for p1 = 8ms

TAT for p2 = (7 + 4) = 11ms

TAT for p3 = (15 + 9) = 24ms

TAT for p4 = (9 + 5) = 14ms

Preemptive

Process	**Burst time**	**Arrival time**
P1	8	0
P2	4	1
P3	9	2
P4	5	3

P1	P2	P4	P1	P3

0 1 5 10 17 26

After executing p1 for 1 ms p2 has came -> p1’s remaining burst time(7ms) p2’s burst time (4ms), so CPU has given to p2. After 2ms p3 has came -> p2’s remining burst time (3 ms) is less than p3’s burst time(9ms).

After 3ms p4 has came -> p2’s remaining burst time (2ms) is lesser than p4’s burst time(5ms)

So p2 continues executing and ends after 5ms. Now p4 has the lowest burst time(5ms) so p4 has been assigned to CPU. No new process has come so p4 ends it's execution after 10ms.Now p1 has lesser remaining burst time(7ms) so again it has been assigned to CPU.

And after 17 ms p3 starts.

WT of p1 = (10 – 1) = 9ms

WT of p2 = (1 – 1) = 0ms (CPU assign time – Arrival time)

WT of p3 = (17 - 2) = 15 ms

WT of p4 = (5 - 3) = 2ms