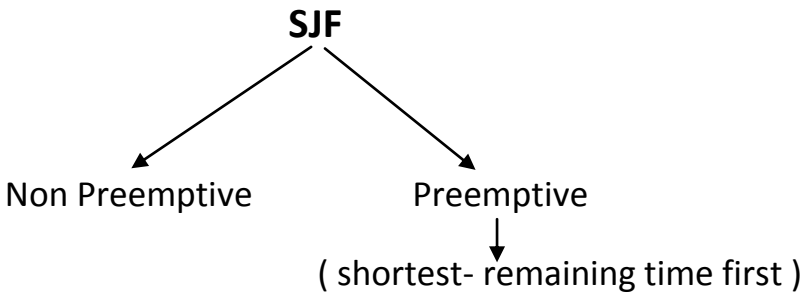
5.6.2 Shortest-Job-First Scheduling (SJF)



Advantages: SJF is optimal, because it gives the minimum A.W.T
(e.g. minimize waiting time)

Disadvantages:

1- The need to Know the length of the next CPU burst.
2- Not suitable for **interactive system** and **short-term scheduling**, Because there is no way to know the length of the next CPU burst.
3- Impossible to implement.

NOTE: **SJF** is used frequently in **long-term scheduling**.

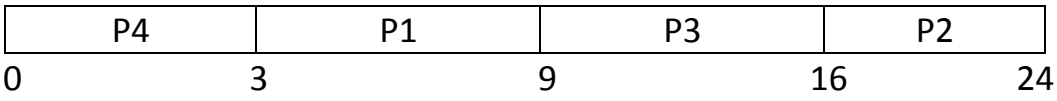
Example 5: consider the following set of processes that arrive at time 0, with the length of the CPU burst time given in milliseconds:

Process	Burst time
P1	**6**
P2	**8**
P3	**7**
P4	**3**

Use SJF scheduling (non-preemptive):

1. Draw Gantt Chart.
2. Find average waiting time.

Gantt Chart:



Waiting time for P4 = (0 – 0) = 0 ms
Waiting time for P1 = (3 – 0) = 3 ms
Waiting time for P3 = (9 – 0) = 9 ms
Waiting time for P2 = (16 – 0) = 16 ms
Average waiting time = (0 + 3 + 9 + 16)/4 = 7 ms

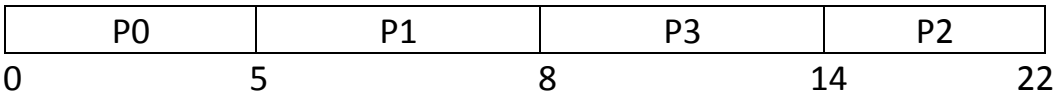
Example 6: Consider the following snapshot (table):

Process	Arrival time	Burst time
P0	0	5
P1	1	3
P2	2	8
P3	3	6

Use non-preemptive SJF scheduling:

1. Draw Gantt Chart.
2. Find average waiting time.

The Gantt Chart:



Waiting time for P0 = (0 – 0) = 0 ms
Waiting time for P1 = (5 – 1) = 4 ms
Waiting time for P3 = (8 – 3) = 5 ms
Waiting time for P2 = (14 – 2) = 12 ms

Average waiting time = (0 + 4 + 5 + 12)/4 = 5.25 ms