CPU Scheduling Terminologies

- **Arrival Time:** Time at which the process arrives in the ready queue.
- **Completion Time:** Time at which process completes its execution.
- **Burst Time:** Time required by a process for CPU execution.
- **Turn Around Time:** Time Difference between completion time and arrival time.

$$Turn\ Around\ Time = Completion\ Time - Arrival\ Time$$

- **Response time**

Response time is the time spent when the process is in the ready state and gets the CPU for the first time. For example, here we are using the First Come First Serve CPU scheduling algorithm for the below 3 processes:

Process	Arrival time	Burst time
P1	0 ms	8 ms
P2	1 ms	7 ms
P3	2 ms	10 ms

Here, the response time of all the 3 processes are:

- **P1:** 0 ms
- **P2:** 7 ms because the process P2 have to wait for 8 ms during the execution of P1 and then after it will get the CPU for the first time. Also, the arrival time of P2 is 1 ms. So, the response time will be 8-1 = 7 ms.
- **P3:** 13 ms because the process P3 have to wait for the execution of P1 and P2 i.e. after 8+7 = 15 ms, the CPU will be allocated to the process P3 for the first time. Also, the arrival of P3 is 2 ms. So, the response time for P3 will be 15-2 = 13 ms.

Response time = Time at which the process gets the CPU for the first time - Arrival time

- **Waiting time**

Waiting time is the total time spent by the process in the ready state waiting for CPU. For example, consider the arrival time of all the below 3 processes to be 0 ms, 0 ms, and 2 ms and we are using the First Come First Serve scheduling algorithm.

Process	Arrival time	Burst time
P1	0 ms	8 ms
P2	0 ms	7 ms
P3	2 ms	10 ms

Gantt Chart

P1		P2		P3	
0 ms	8 ms	8 ms	15 ms	15 ms	25 ms

Then the waiting time for all the 3 processes will be:

- **P1:** 0 ms
- **P2:** 8 ms because P2 have to wait for the complete execution of P1 and arrival time of P2 is 0 ms.
- **P3:** 13 ms becuase P3 will be executed after P1 and P2 i.e. after 8+7 = 15 ms and the arrival time of P3 is 2 ms. So, the waiting time of P3 will be: 15-2 = 13 ms.

Waiting time = Turnaround time - Burst time

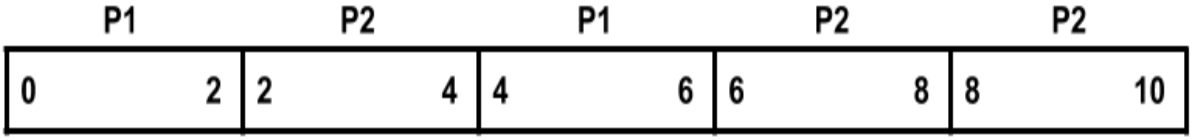
In the above example, the processes have to wait only once. But in many other scheduling algorithms, the CPU may be allocated to the process for some time and then the process will be moved to the waiting state and again after some time, the process will get the CPU and so on.

There is a difference between waiting time and response time. Response time is the time spent between the ready state and getting the CPU for the first time. But the waiting time is the total time taken by the process in the ready state. Let's take an example of a round-robin scheduling algorithm. The time quantum is 2 ms.

Process	Arrival time	Burst time
P1	0 ms	4 ms
P2	0 ms	6 ms

Time Quantum = 2ms

Gantt Chart



In the above example, the response time of the process P2 is 2 ms because after 2 ms, the CPU is allocated to P2 and the waiting time of the process P2 is 4 ms i.e turnaround time - burst time (10 - 6 = 4 ms).

- **Waiting Time (W.T):** Time Difference between turn around time and burst time.

$$Waiting\ Time = Turn\ Around\ Time - Burst\ Time$$

First Come First Serve (Criteria: Arrival Time, Mode: Non-preemtive)

Characteristics of FCFS

- FCFS supports non-preemptive and preemptive CPU scheduling algorithms.
- Tasks are always executed on a First-come, First-serve concept.
- FCFS is easy to implement and use.
- This algorithm is not very efficient in performance, and the wait time is quite high.

Algorithm for FCFS Scheduling

- The waiting time for the first process is 0 as it is executed first.
- The waiting time for the upcoming process can be calculated by:

wt[i] = (at[i – 1] + bt[i – 1] + wt[i – 1]) – at[i]

where

- ***wt[i]*** *= waiting time of current process*
- ***at[i-1]*** *= arrival time of previous process*
- ***bt[i-1]*** *= burst time of previous process*
- ***wt[i-1]*** *= waiting time of previous process*
- ***at[i]*** *= arrival time of current process*

Examples to Show Working of Non-Preemptive First come First Serve CPU Scheduling Algorithm

Example-1: Consider the following table of arrival time and burst time for five processes **P1, P2, P3, P4** and **P5**.

Processes	Arrival Time	Burst Time
P1	0	4
P2	1	3
P3	2	1

Processes	Arrival Time	Burst Time
P4	3	2
P5	4	5

The **First come First serve** CPU Scheduling Algorithm will work on the basis of steps as mentioned below:

Step 0: At time = 0,

- The process begins with **P1**
- As it has an arrival time 0

Time Instance	Process	Arrival Time	Waiting Table	Execution Time	Initial Burst Time	Remaining Burst Time
0-1ms	P1	0ms		1ms	4ms	3ms

Step 1: At time = 1,

- The process **P2** arrives
- But process **P1** still executing,
- Thus, **P2 is** kept on a waiting table and waits for its execution.

Time Instance	Process	Arrival Time	Waiting Table	Execution Time	Initial Burst Time	Remaining Burst Time
1-2ms	P1	0ms		1ms	3ms	2ms
	P2	1ms	P2	0ms	3ms	3ms

Step 3: At time = 2,

- The process **P3** arrives and kept in a waiting queue
- While process **P1** is still executing as its burst time is 4.

Time Instance	Process	Arrival Time	Waiting Table	Execution Time	Initial Burst Time	Remaining Burst Time
	P1	0ms		1ms	2ms	1ms
2-3ms	P2	1ms	P2	0ms	3ms	3ms
	P3	2ms	P2, P3	0ms	1ms	1ms

Step 4: At time = 3,

- The process **P4** arrives and kept in the waiting queue
- While process **P1** is still executing as its burst time is **4**

Time Instance	Process	Arrival Time	Waiting Table	Execution Time	Initial Burst Time	Remaining Burst Time
	~~P1~~	~~0ms~~		~~1ms~~	~~1ms~~	~~0ms~~
3-4ms	P2	1ms	P2	0ms	3ms	3ms
	P3	2ms	P2, P3	0ms	1ms	1ms
	P4	3ms	P2, P3, P4	0ms	2ms	2ms

Step 5: At time = 4,

- The process **P1** completes its execution
- Process **P5** arrives in waiting queue while process **P2** starts executing

Time Instance	Process	Arrival Time	Waiting Table	Execution Time	Initial Burst Time	Remaining Burst Time
4-5ms	P2	1ms		1ms	3ms	2ms

Time Instance	Process	Arrival Time	Waiting Table	Execution Time	Initial Burst Time	Remaining Burst Time
	P3	2ms	P3	0ms	1ms	1ms
	P4	3ms	P3, P4	0ms	2ms	2ms
	P5	4ms	P3, P4, P5	0ms	5ms	5ms

Step 6: At time = 5,

- The process **P2** completes its execution

Time Instance	Process	Arrival Time	Waiting Table	Execution Time	Initial Burst Time	Remaining Burst Time
	~~P2~~	~~1ms~~		~~2ms~~	~~2ms~~	~~0ms~~
5-7ms	P3	2ms	P3	0ms	1ms	1ms
	P4	3ms	P3, P4	0ms	2ms	2ms
	P5	4ms	P3, P4, P5	0ms	5ms	5ms

Step 7: At time = 7,

- Process **P3** starts executing, it has burst time of 1 thus, it completes execution at time interval 8

Time Instance	Process	Arrival Time	Waiting Table	Execution Time	Initial Burst Time	Remaining Burst Time
7-8ms	~~P3~~	~~2ms~~		~~1ms~~	~~1ms~~	~~0ms~~

Time Instance	Process	Arrival Time	Waiting Table	Execution Time	Initial Burst Time	Remaining Burst Time
	P4	3ms	P4	0ms	2ms	2ms
	P5	4ms	P4, P5	0ms	5ms	5ms

Step 8: At time 8,

- The process of **P3** completes its execution
- Process **P4** starts executing, it has burst time of 2 thus, it completes execution at time interval 10.

Time Instance	Process	Arrival Time	Waiting Table	Execution Time	Initial Burst Time	Remaining Burst Time
8-10ms	~~P4~~	~~3ms~~		~~2ms~~	~~2ms~~	~~0ms~~
	P5	4ms	P5	0ms	5ms	5ms

Step 9: At time 10,

- The process **P4** completes its execution
- Process **P5** starts executing, it has burst time of 5 thus, it completes execution at time interval 15.

Time Instance	Process	Arrival Time	Waiting Table	Execution Time	Initial Burst Time	Remaining Burst Time
10-15ms	~~P5~~	~~4ms~~		~~5ms~~	~~5ms~~	~~0ms~~

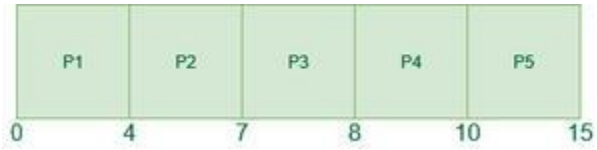
Step 10: At time 15,

- Process P5 will finish its execution.
- The overall execution of the processes will be as shown below:

Time Instance	Process	Arrival Time	Waiting Table	Execution Time	Initial Burst Time	Remaining Burst Time
0-1ms	P1	0ms		1ms	4ms	3ms
1-2ms	P1	0ms		1ms	3ms	2ms
	P2	1ms	P2	0ms	3ms	3ms
2-3ms	P1	0ms		1ms	2ms	1ms
	P2	1ms	P2	0ms	3ms	3ms
	P3	2ms	P2, P3	0ms	1ms	1ms
3-4ms	~~**P1**~~	~~**0ms**~~		~~**1ms**~~	~~**1ms**~~	~~**0ms**~~
	P2	1ms	P2	0ms	3ms	3ms
	P3	2ms	P2, P3	0ms	1ms	1ms
	P4	3ms	P2, P3, P4	0ms	2ms	2ms
4-5ms	P2	1ms		1ms	3ms	2ms
	P3	2ms	P3	0ms	1ms	1ms
	P4	3ms	P3, P4	0ms	2ms	2ms

Time Instance	Process	Arrival Time	Waiting Table	Execution Time	Initial Burst Time	Remaining Burst Time
	P5	4ms	P3, P4, P5	0ms	5ms	5ms
	~~**P2**~~	~~**1ms**~~		~~**2ms**~~	~~**2ms**~~	~~**0ms**~~
	P3	2ms	P3	0ms	1ms	1ms
5-7ms	P4	3ms	P3, P4	0ms	2ms	2ms
	P5	4ms	P3, P4, P5	0ms	5ms	5ms
	~~**P3**~~	~~**2ms**~~		~~**1ms**~~	~~**1ms**~~	~~**0ms**~~
7-8ms	P4	3ms	P4	0ms	2ms	2ms
	P5	4ms	P4, P5	0ms	5ms	5ms
	~~**P4**~~	~~**3ms**~~		~~**2ms**~~	~~**2ms**~~	~~**0ms**~~
8-10ms	P5	4ms	P5	0ms	5ms	5ms
10-15ms	~~**P5**~~	~~**4ms**~~		~~**5ms**~~	~~**5ms**~~	~~**0ms**~~

Gantt Chart for Above Execution

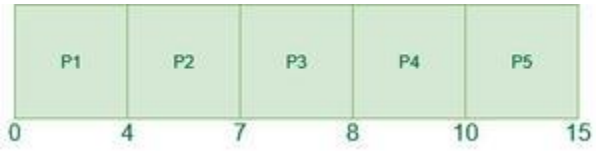


Gantt chart for First come First serve Scheduling

Waiting Time = Start time – Arrival time
P1 = 0 – 0 = 0
P2 = 4 – 1 = 3
P3 = 7 – 2 = 5
P4 = 8 – 3 = 5
P5 = 10 – 4 = 6
Average waiting time *= (0 + 3 + 5 + 5+ 6)/ 5 = 19 / 5 =* ***3.8***

Process ID	AT	BT	CT	TAT(CT-AT)	WT(TAT-BT)	RT(First Response time-AT)
P1	0	4	4	4	0	0
P2	1	3	7	6	3	3
P3	2	1	8	6	5	5
P4	3	2	10	7	5	5
P5	4	5	15	11	6	6
				Avg TAT=	Avg. WT=	

Gantt Chart:



Task: DIY…..

Process ID	AT	BT
P1	0	2
P2	1	2
P3	5	3
P4	6	4