## Paper Name : Operating System

## Prepared by : Debanjali Jana

**Scheduling Criteria**

- CPU utilization – keep the CPU as busy as possible
- Throughput – number of processes that complete their execution per time unit
- Turnaround time – amount of time to execute a particular process. It is the sum of the periods spent waiting to get into memory, waiting in the ready queue, executing on the CPU, and doing I/0.
- Waiting time – amount of time that a process spends waiting in the ready queue
- Response time – amount of time it takes from when a request was submitted until the first response is produced, not output. Thus, it is the time from the submission of a request until the first response is produced.

## Scheduling Algorithms

- **First-Come First- Serve (FCFS) Scheduling**

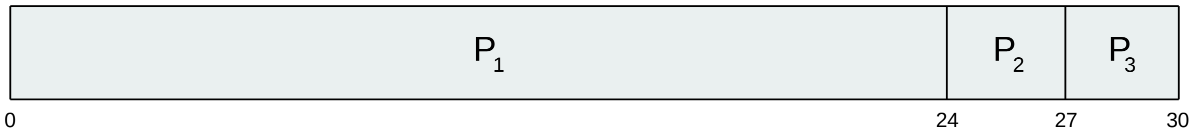
**First come first serve** (FCFS) scheduling algorithm simply schedules the jobs according to their arrival time. The job which comes first in the ready queue will get the CPU first. Also it is always non-preemptive in nature. In this type of algorithm, processes which requests the CPU first get the CPU allocation first. This is managed with a FIFO queue. As the process enters the ready queue, its PCB (Process Control Block) is linked with the tail of the queue and, when the CPU is free, it is allocated to the process at the head of the queue.

On the negative side, the average waiting time under the FCFS policy is often quite long. FCFS may suffer from the **convoy effect** if the burst time of the first job is the highest among all. If the CPU gets the processes of the higher burst time at the front end of the ready queue then the processes of lower burst time may get blocked which means they may never get the CPU if the job in the execution has a very high burst time. This is called **convoy effect.**

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

| Process | Burst Time |
|---|---|
| $P_1$ | 24 |
| $P_2$ | 3 |
| $P_3$ | 3 |

Suppose that the processes arrive in the order: $P_1$ , $P_2$ , $P_3$
The Gantt Chart for the schedule is:



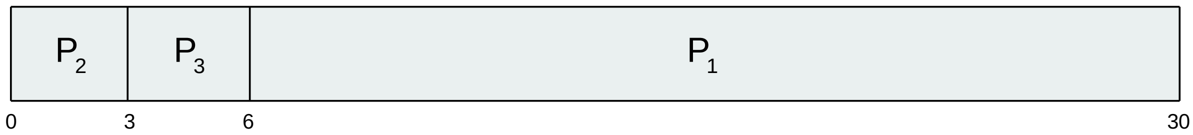

Waiting time for $P_1 = 0$; $P_2 = 24$; $P_3 = 27$ milliseconds

Average waiting time = (0 + 24 + 27)/3 = 17 milliseconds

Suppose that the processes arrive in the order:

$P_2$ , $P_3$ , $P_1$

The Gantt chart for the schedule is:




Waiting time for $P_1 = 6$; $P_2 = 0$; $P_3 = 3$ milliseconds

Average waiting time: (6 + 0 + 3)/3 = 3 milliseconds

- Much better than previous case
- **Convoy effect** - short process behind long process