

First Come First Served (FCFS)

Table of Contents

1. What is FCFS?	1
2. Key Characteristics	1
3. Example Scenario	1
3.1. Waiting Times example 1:	2
3.2. Waiting Times example 2:	2
4. Why Long Waiting Times?	2
5. Non-Preemptive vs Preemptive	2
6. Implementation	3
7. Live Demo	3
8. Metrics	3
8.1. Key Scheduling Metrics	3
8.2. Timing Metrics	3
8.3. Timing Metrics	4
8.4. Summary	4
9. Sources	4

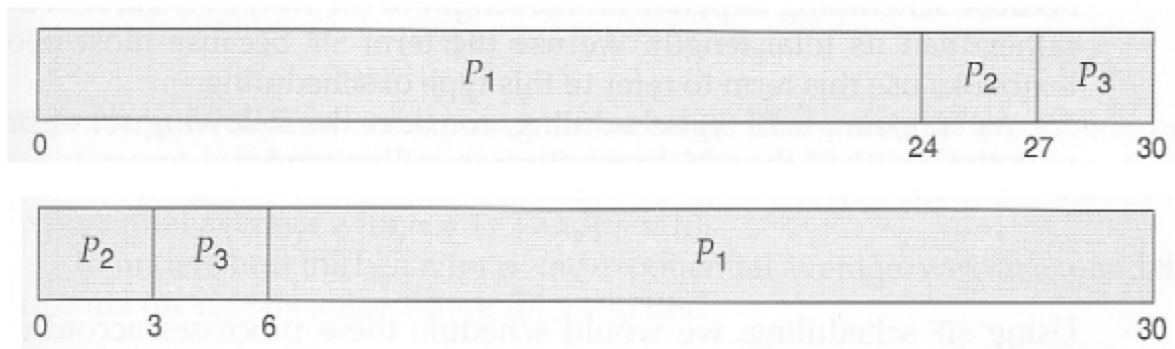
1. What is FCFS?

- **Definition:** Processes are allocated to the CPU in the order they arrive (FIFO).
- **Non-preemptive:** Once started, a process runs **to completion**.
- **Key Metric:** Often results in **long average waiting times**.

2. Key Characteristics

- **Simplicity:** Easy to understand and implement.
- **Fairness:** No prioritization — "first arrived, first served".
- **Drawback:**
 - No consideration for process length.
 - One long process can delay all others (**Convoy Effect**).

3. Example Scenario



3.1. Waiting Times example 1:

- P_1 : 0ms
- P_2 : 24ms
- P_3 : 27ms

Average: 17ms

3.2. Waiting Times example 2:

- P_1 : 0ms
- P_2 : 3ms
- P_3 : 6ms

Average: 3ms

4. Why Long Waiting Times?

No Sorting: Processes aren't ordered by burst time.

Shortest Job First (SJF) fixes that

5. Non-Preemptive vs Preemptive

Preemption	Non-Preemption
Processes can be interrupted mid-execution	Processes run to completion once started
Better for time-sharing systems	Suitable for batch processing systems
Higher overhead (frequent context switches)	Lower overhead (fewer interruptions)
Provides better responsiveness	May cause convoy effect
Examples: Round Robin, SRTF	Examples: FCFS, Non-preemptive SJF

6. Implementation

Using a FIFO Queue:

```
private final Queue<MyProcess> queue = new ArrayBlockingQueue<>( 10 );

public void schedule( MyProcess myProcess ) {
    queue.add( myProcess );
}

public void runCpu() {
    while ( !queue.isEmpty() ) {
        var process = queue.poll();
        // non-preemptive
        process.run();
    }
}
```

7. Live Demo

8. Metrics

8.1. Key Scheduling Metrics

Burst Time:

- Total CPU execution time of process
- Formula: End time - Start time

Arrival Time:

- When process enters ready queue
- Example: P1=0ms, P2=1ms, P3=2ms

8.2. Timing Metrics

Exit Time:

- When process completes execution

Response Time:

- First CPU allocation - Arrival time

Example (FCFS):

- P2: $8\text{ms} - 1\text{ms} = 7\text{ms}$
- P3: $15\text{ms} - 2\text{ms} = 13\text{ms}$

8.3. Timing Metrics

Waiting Time:

- Total ready queue time

Turnaround Time:

- Completion to arrival duration

8.4. Summary

1. Burst Time = Actual CPU usage
2. Arrival/Exit = Process lifecycle markers
3. Response measures initial delay
4. Turnaround reflects total processing duration

9. Sources

- <https://afteracademy.com/blog/what-is-burst-arrival-exit-response-waiting-turnaround-time-and-throughput/>
- <https://www.geeksforgeeks.org/difference-between-arrival-time-and-burst-time-in-cpu-scheduling/>
- <https://www.youtube.com/watch?v=VSMAjMfJ6KQ&t=54s>
- Moodle PDF