

Operating Systems

| CPU Scheduling - Who Gets the CPU Next?

Prof. Dr. Mostafa Elhosseini

Professor of Smart Systems Engineering

<https://elhosseini.com/>

| Agenda

- The Roller Coaster - What is CPU Scheduling?
- The Challenge
- The Goal of CPU Scheduling
- What Does “CPU Burst” Mean?
- The Three Schedulers
- Preemptive vs Non-Preemptive
- CPU Scheduling Algorithms
- First-Come, First-Served (FCFS)



القطار السريع - The Roller Coaster



Welcome to the Amusement Park (حديقة الملاهي) Where the Operating System plays the role of the ride manager —deciding who rides next, who waits, and how long each ride lasts.

حديقة الملاهي الافتراضية



فيها لعبة شهيرة جدًا يحبها الجميع — الأفعوانية

لكن عندنا مشكلة: مئات الأشخاص (العمليات) ينتظرون، ويوجد قطار واحد فقط (المعالج).



القطار السريع - The Roller Coaster



The Roller Coaster

- Hundreds of excited visitors (   — processes)
- all waiting to enjoy one roller coaster  (the CPU).
- The line keeps growing, people keep arriving,
- and the manager must keep the ride busy — no empty seats, no chaos!

Key Points

 In a system with a single CPU core, only **one process can run at a time**.

 The other processes must **wait** until the CPU becomes free and can be rescheduled.

 The main goal of **multiprogramming** is to ensure that some process is always running, keeping the CPU fully utilized.



The Challenge



Keep the CPU (roller coaster) always busy.



Serve everyone fairly — no one should wait forever.



Choose who rides next for the best overall performance.



The Goal of CPU Scheduling



Its mission?

To keep everyone reasonably happy 😊

while keeping the CPU always busy 🧠 ⚡



The Goal of CPU Scheduling



Maximize CPU Utilization → no idle time!

- CPU Utilization is the percentage of time the CPU is actively working (executing processes) instead of being idle.
- It measures how busy the CPU is —the higher the utilization, the better the performance (up to a safe limit).
- $$\text{CPU Utilization} = \frac{\text{CPU Busy Time}}{\text{Total Time}} \times 100\%$$



The Goal of CPU Scheduling



Increase **Throughput** - (معدل الإنتاجية)

- **Throughput** 👉 The number of processes completed per unit of time.
- **Analogy:** Like counting how many people finish the roller coaster ride every hour.



The Goal of CPU Scheduling



Minimize Waiting & **Turnaround Time** - (زمن الإنجاز).



The total time taken for a process to complete —from the moment it enters the system until it finishes execution.

- Turnaround Time = Completion Time – Arrival Time
- It's like the total time a visitor spends in the amusement park —from entering the queue 🎫 to finishing the ride.



The Goal of CPU Scheduling



Improve Response Time (زمن الاستجابة)



The time from when a request is submitted until the system produces the first response.

- It's not the total time to finish — just the time until the first reaction appears.
- When you click a button on your computer , how long before you see a reaction on screen  ⚡ — that's the response time.



Goals

- Max CPU utilization
- High throughput
- Low waiting & turnaround time
- Quick response for users

What Does “CPU Burst” Mean?

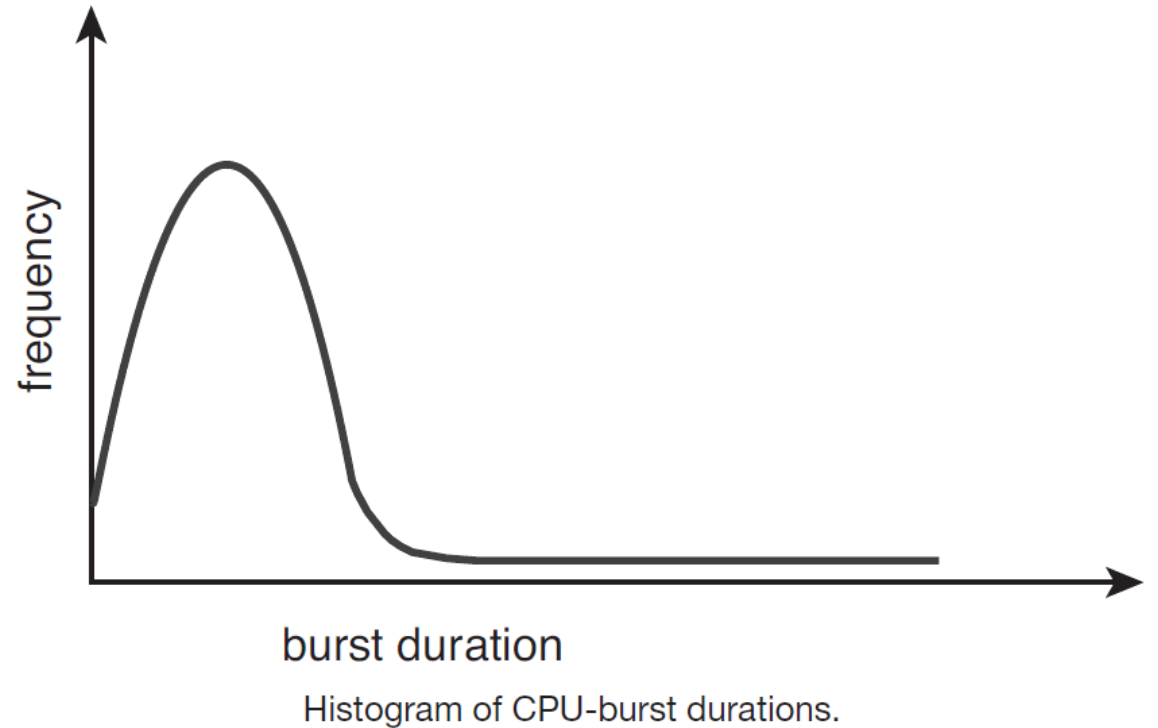
Origin of the Term

- The word “Burst” literally means explosion or sudden surge 
- In the 1950s–1960s, OS researchers studied how processes use the CPU 
- They discovered that CPU usage isn’t constant — it comes in waves 

🌟 What Does “CPU Burst” Mean?

📊 Observed Pattern

- The CPU activity graph looked like sharp peaks (spikes) 🏔️
- Each spike represented a quick, intense period of computation ⚡
- Between bursts, processes often waited for I/O ⌚



It's a Game of Balance

Between:

-  Fairness → every process gets a chance.
-  Efficiency → the CPU never rests.
-  Speed → users feel the system is snappy and alive.

The Three Schedulers

- ❶ Long-Term Scheduler → decides who enters the system..
 - ❷ Medium-Term Scheduler → suspends / resumes processes.
 - ❸ Short-Term Scheduler → picks next process for the CPU.
-  The short-term scheduler works every few milliseconds!



Preemptive vs Non-Preemptive



Non-Preemptive تنفيذ مستمر بلا مقاطعة

process keeps CPU till done / blocked.



Preemptive يسمح بالقطع: OS can pause & switch anytime.



Example:

Non-Preemptive = long speech nobody can interrupt.

Preemptive = talk show with a time limit per speaker.



CPU Scheduling Algorithms



Purpose: To decide which process gets the CPU next—in a way that balances efficiency, fairness, and responsiveness ⚡ 💻

- 1 First-Come, First-Served (FCFS)
- 2 Shortest Job First (SJF)
- 3 Shortest Remaining Time First (SRTF)
- 4 Priority Scheduling
- 5 Round Robin (RR)
- 6 Multilevel Queue Scheduling (MLQ)
- 7 Multilevel Feedback Queue (MLFQ)

First-Come, First-Served (FCFS)

 Simplest algorithm

- The process that arrives first runs first.
- Like standing in line at a bank — whoever came first gets served first.

 Problem: Convoy effect **تأثير القافلة** — short jobs wait behind long ones - slow jobs can block the queue



Best for batch systems with no need for interactivity.

1 First-Come, First-Served (FCFS)

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

<u>Process</u>	<u>Burst Time</u>
P_1	24
P_2	3
P_3	3

- If the processes arrive in the order P_1, P_2, P_3 , and are served in FCFS order
- Draw the Gantt Chart - Average Waiting Time - Average Turnaround Time

1 First-Come, First-Served (FCFS)

<u>Process</u>	<u>Burst Time</u>
P_1	24
P_2	3
P_3	3

1 FCFS – Example 02

Process	Arrival Time	Burst Time
P1	0	7
P2	2	4
P3	4	1
P4	5	4