**INTRODUCTION**

Central Processing Unit (CPU) scheduling is a crucial aspect of operating systems, responsible for determining the order in which processes or tasks are executed by the CPU. It aims to optimize system performance by efficiently allocating CPU time among competing processes, minimizing waiting times, and maximizing overall throughput. Various scheduling algorithms, such as First-Come-First-Serve (FCFS), Shortest Job Next (SJN), and Round Robin, are employed to achieve these objectives and adapt to different workload scenarios.

CPU scheduling refers to the switching between processes that are being executed. It forms the basis of multiprogrammed systems. This switching ensures that CPU utilization is maximized so that the computer is more productive. There are two main types of CPU scheduling, preemptive and non-preemptive.



Figure 1 (Introduction image of CPU)

**Meaning of central processing unit scheduling**

Central Processing Unit (CPU) scheduling refers to the mechanism by which tasks or processes are scheduled to run on a computer's CPU. It involves the selection of the next process to execute from the ready queue, where processes wait for their turn to run. The goal of CPU scheduling is to optimize system performance by efficiently utilizing the CPU and minimizing wait times for processes.

1. CPU Scheduling is a**process that allows one process to use the CPU while another process is delayed** (in standby) due to unavailability of any resources such as I / O etc. thus making full use of the CPU. The purpose of CPU Scheduling is to make the system more efficient, faster, and fairer.
2. Whenever the CPU becomes idle, the operating system must select one of the processes in the line ready for launch. The selection process is done by a temporary (CPU) scheduler. The Scheduler selects between memory processes ready to launch and assigns the CPU to one of them.

**What is Process Scheduling?**

Process Scheduling is the process of the process manager handling the removal of an active process from the CPU and selecting another process based on a specific strategy.

Process Scheduling is an integral part of Multi-programming applications. Such operating systems allow more than one process to be loaded into usable memory at a time and the loaded shared CPU process uses repetition time.

**There are three types of process schedulers**

1. Long term or Job Scheduler
2. Short term or CPU Scheduler
3. Medium-term Scheduler

**Types Of CPU Scheduling**

Here are many types of central processing scheduling is given in a diagram.

1. Preemptive
2. Non-preemptive

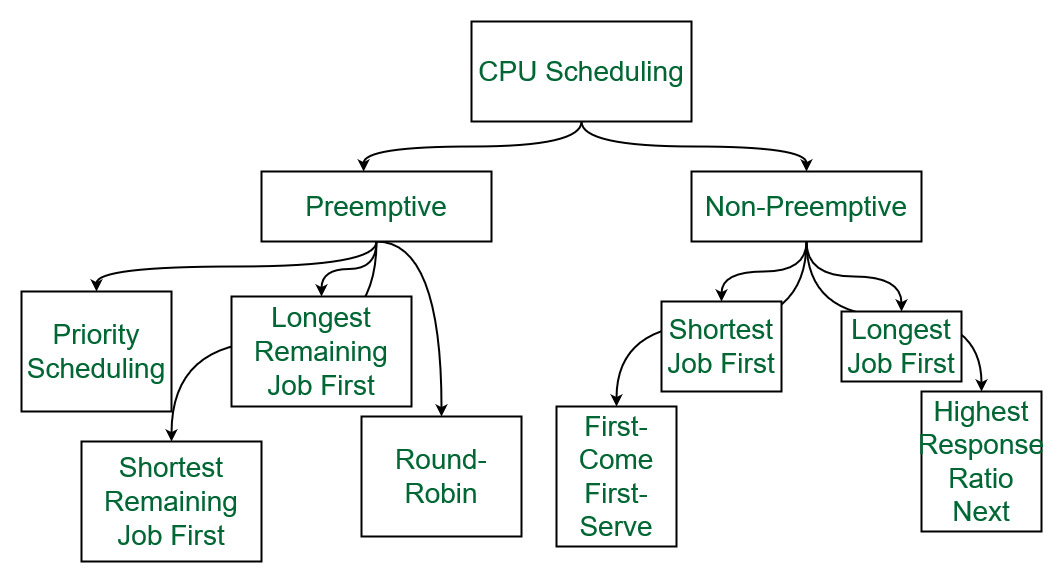


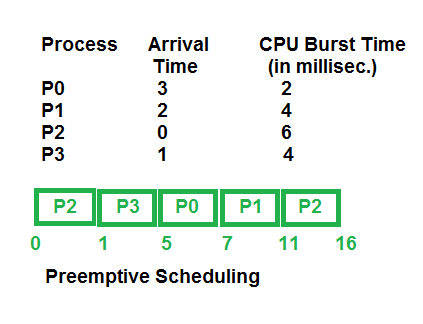
Figure 2 (Types of CPU scheduling)

## ****Preemptive Scheduling****

Preemptive scheduling is used when a process switches from the running state to the ready state or from the waiting state to the ready state. The resources (mainly CPU cycles) are allocated to the process for a limited amount of time and then taken away, and the process is again placed back in the ready queue if that process still has CPU burst time remaining

**Algorithms based on preemptive scheduling are**

1. [Round Robin (RR)](https://www.geeksforgeeks.org/program-round-robin-scheduling-set-1/)
2. [Shortest Remaining Time First (SRTF)](https://www.geeksforgeeks.org/program-shortest-job-first-scheduling-set-2srtf-make-changesdoneplease-review/)
3. [Priority (preemptive version)](https://www.geeksforgeeks.org/program-for-preemptive-priority-cpu-scheduling/) etc.



## 

Preemptive scheduling has a number of advantages and disadvantages. The following are non-preemptive scheduling’s benefits and drawbacks:

**Advantages**

1. Because a process may not monopolize the processor, it is a more reliable method.
2. Each occurrence prevents the completion of ongoing tasks.
3. The average response time is improved.
4. Utilizing this method in a multi-programming environment is more advantageous.
5. The operating system makes sure that every process using the CPU is using the same amount of CPU time.

**Disadvantages**

1. Limited computational resources must be used.
2. Suspending the running process, change the context, and dispatch the new incoming process all take more time.
3. The low-priority process would have to wait if multiple high-priority processes arrived at the same time.

## ****Non-Preemptive Scheduling****

Non-preemptive Scheduling is used when a process terminates, or a process switches from running to the waiting state. In this scheduling, once the resources (CPU cycles) are allocated to a process, the process holds the CPU till it gets terminated or reaches a waiting state. In the case of non-preemptive scheduling does not interrupt a process running CPU in the middle of the execution. Instead, it waits till the process completes its CPU burst time, and then it can allocate the CPU to another process.

**Algorithms based on non-preemptive scheduling are:**

1. [Shortest Job First (SJF basically non preemptive)](https://www.geeksforgeeks.org/program-shortest-job-first-sjf-scheduling-set-1-non-preemptive/)
2. [Priority (non-preemptive version)](https://www.geeksforgeeks.org/operating-system-priority-scheduling-different-arrival-time-set-2/)

# **Priority Scheduling Algorithm: Preemptive, Non-Preemptive**

## What is Priority Scheduling?

**Priority Scheduling** is a method of scheduling processes that is based on priority. In this algorithm, the scheduler selects the tasks to work as per the priority.

The processes with higher priority should be carried out first, whereas jobs with equal priorities are carried out on a round-robin or FCFS basis. Priority depends upon memory requirements, time requirements, etc.

## Characteristics of Priority Scheduling:

* A CPU algorithm that schedules processes based on priority.
* It used in Operating systems for performing batch processes.
* If two jobs having the same priority are READY, it works on a [FIRST COME, FIRST SERVED](https://www.guru99.com/fcfs-scheduling.html) basis.
* In priority scheduling, a number is assigned to each process that indicates its priority level.
* Lower the number, higher is the priority.
* In this type of scheduling algorithm, if a newer process arrives, that is having a higher priority than the currently running process, then the currently running process is preempted.

## Example of Priority Scheduling:

Consider following five processes P1 to P5. Each process has its unique priority, burst time, and arrival time.

| Process | Priority | Burst time | Arrival time |
| --- | --- | --- | --- |
| P1 | 1 | 4 | 0 |
| P2 | 2 | 3 | 0 |
| P3 | 1 | 7 | 6 |
| P4 | 3 | 4 | 11 |
| P5 | 2 | 2 | 12 |

## Advantages of priority scheduling:

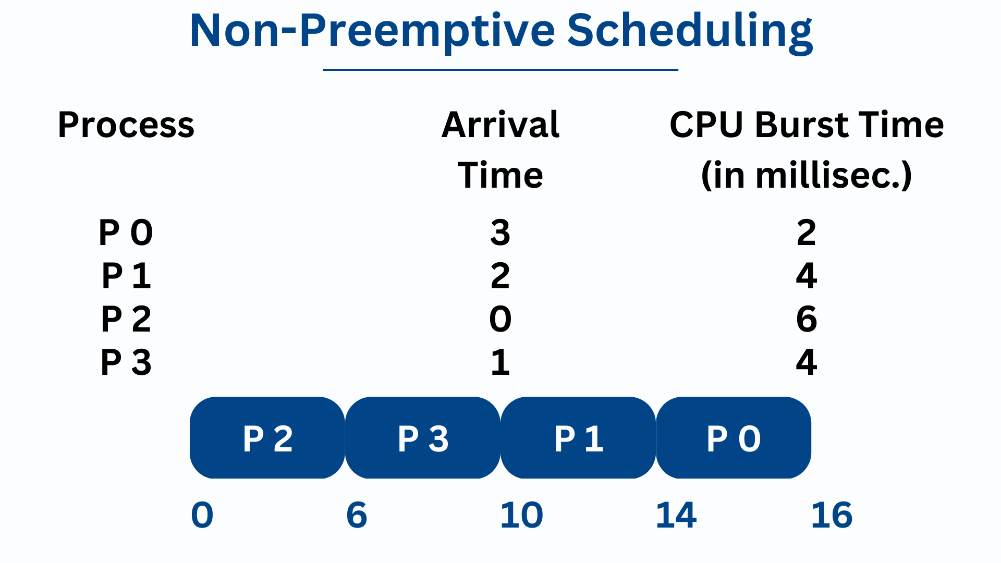
Here, are benefits/pros of using priority scheduling method:

* Easy to use scheduling method
* Processes are executed on the basis of priority so high priority does not need to wait for long which saves time
* This method provides a good mechanism where the relative important of each process may be precisely defined.
* Suitable for applications with fluctuating time and resource requirements.

## Disadvantages of priority scheduling:

Here, are cons/drawbacks of priority scheduling

* If the system eventually crashes, all low priority processes get lost.
* If high priority processes take lots of CPU time, then the lower priority processes may starve and will be postponed for an indefinite time.
* This scheduling algorithm may leave some low priority processes waiting indefinitely.
* A process will be blocked when it is ready to run but has to wait for the CPU because some other process is running currently.
* If a new higher priority process keeps on coming in the ready queue, then the process which is in the waiting state may need to wait for a long duration of time.



# **Round Robin Scheduling Algorithm**

## What is Round-Robin Scheduling?

The name of this algorithm comes from the round-robin principle, where each person gets an equal share of something in turns. It is the oldest, simplest scheduling algorithm, which is mostly used for multitasking.

In Round-robin scheduling, each ready task runs turn by turn only in a cyclic queue for a limited time slice. This algorithm also offers starvation free execution of processes.

## Characteristics of Round-Robin Scheduling

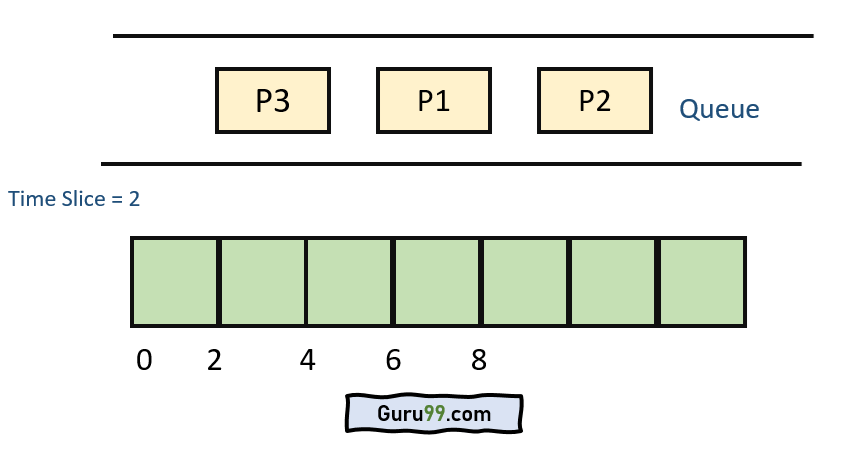
Here are the important characteristics of Round-Robin Scheduling:

* Round robin is a pre-emptive algorithm
* The CPU is shifted to the next process after fixed interval time, which is called time quantum/time slice.
* The process that is preempted is added to the end of the queue.
* Round robin is a hybrid model which is clock-driven

## Example of Round-robin Scheduling

Consider this following three processes

| **Process Queue** | **Burst time** |
| --- | --- |
| P1 | 4 |
| P2 | 3 |
| P3 | 5 |



**Step 1)**

  The execution begins with process P1, which has burst time 4. Here, every process executes for 2 seconds. P2 and P3 are still in the waiting queue.