Process Schedulers in Operating System

Last Updated : 26 Jun, 2024

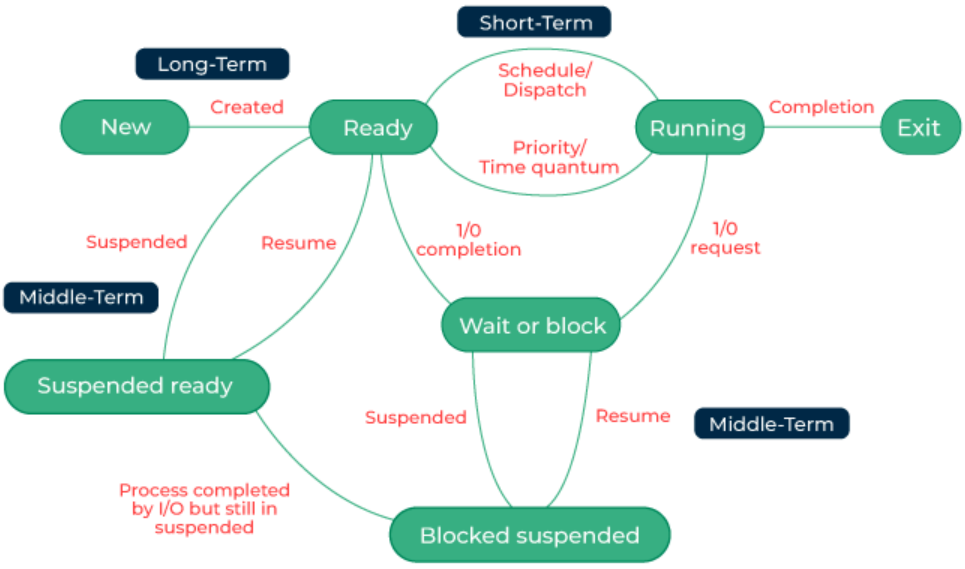
In computing, a process is the instance of a computer program that is being executed by one or many threads. Scheduling is important in many different computer environments. One of the most important areas of scheduling is which programs will work on the CPU. This task is handled by the Operating System (OS) of the computer and there are many different ways in which we can choose to configure programs.

Process schedulers are fundamental components of operating systems responsible for deciding the order in which processes are executed by the CPU. In simpler terms, they manage how the CPU allocates its time among multiple tasks or processes that are competing for its attention. In this article, we are going to discuss

What is Process Scheduling?

Process scheduling is the activity of the process manager that handles the removal of the running process from the CPU and the selection of another process based on a particular strategy.

Process scheduling is an essential part of a Multiprogramming operating system. Such operating systems allow more than one process to be loaded into the executable memory at a time and the loaded process shares the CPU using time multiplexing.



Process scheduler

Categories of Scheduling

Scheduling falls into one of two categories:

- **Non-Preemptive:** In this case, a process's resource cannot be taken before the process has finished running. When a running process finishes and transitions to a waiting state, resources are switched.
- **Preemptive:** In this case, the OS assigns resources to a process for a predetermined period. The process switches from running state to ready state or from waiting state to ready state during resource allocation. This switching happens because the CPU may give other processes priority and substitute the currently active process for the higher priority process.

Types of Process Schedulers

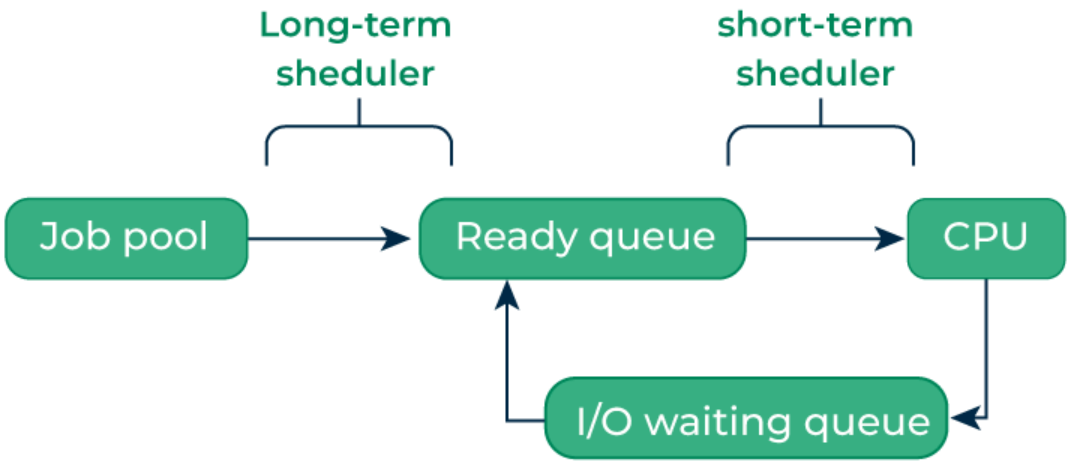
There are three types of process schedulers:

1. Long Term or Job Scheduler

It brings the new process to the 'Ready State'. It controls the Degree of Multi-programming, i.e., the number of processes present in a ready state at any point in time. It is important that the long-term scheduler make a careful selection of both I/O and CPU-bound processes. I/O-bound tasks are which use much of their time in input and output operations while CPU-bound processes are which spend their time on the CPU. The job scheduler increases efficiency by maintaining a balance between the two. They operate at a high level and are typically used in batch-processing systems.

2. Short-Term or CPU Scheduler

It is responsible for selecting one process from the ready state for scheduling it on the running state. Note: Short-term scheduler only selects the process to schedule it doesn't load the process on running. Here is when all the scheduling algorithms are used. The CPU scheduler is responsible for ensuring no starvation due to high burst time processes.



Short Term Scheduler

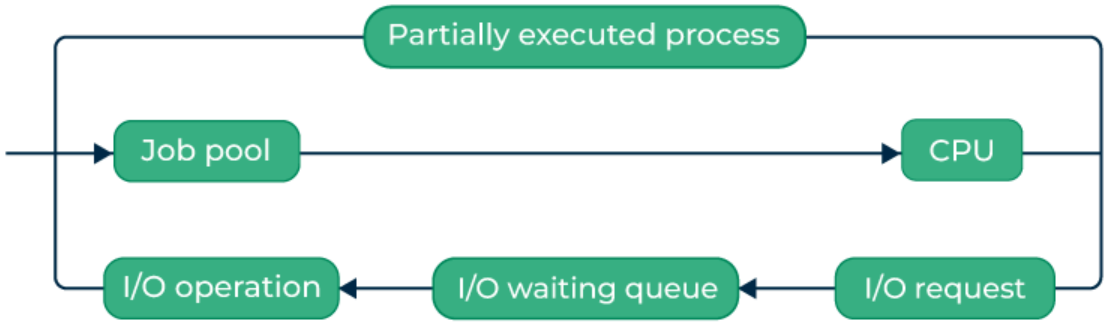
The dispatcher is responsible for loading the process selected by the Short-term scheduler on the CPU (Ready to Running State) Context

switching is done by the dispatcher only. A dispatcher does the following:

- Switching context.
- Switching to user mode.
- Jumping to the proper location in the newly loaded program.

3. Medium-Term Scheduler

It is responsible for suspending and resuming the process. It mainly does swapping (moving processes from main memory to disk and vice versa). Swapping may be necessary to improve the process mix or because a change in memory requirements has overcommitted available memory, requiring memory to be freed up. It is helpful in maintaining a perfect balance between the I/O bound and the CPU bound. It reduces the degree of multiprogramming.



Medium Term Scheduler

Some Other Schedulers

- **I/O Schedulers:** I/O schedulers are in charge of managing the execution of I/O operations such as reading and writing to discs or networks. They can use various algorithms to determine the order in

which I/O operations are executed, such as FCFS (First-Come, First-Served) or RR (Round Robin).

- **Real-Time Schedulers:** In real-time systems, real-time schedulers ensure that critical tasks are completed within a specified time frame. They can prioritize and schedule tasks using various algorithms such as EDF (Earliest Deadline First) or RM (Rate Monotonic).

Comparison Among Scheduler

Long Term Scheduler	Short Term Schedular	Medium Term Scheduler
It is a job scheduler	It is a CPU scheduler	It is a process-swapping scheduler.
Generally, Speed is lesser than short term scheduler	Speed is the fastest among all of them.	Speed lies in between both short and long-term schedulers.
It controls the degree of multiprogramming	It gives less control over how much multiprogramming is done.	It reduces the degree of multiprogramming.
It is barely present or nonexistent in the time-sharing system.	It is a minimal time-sharing system.	It is a component of systems for time sharing.
It can re-enter the process into memory, allowing for the continuation of execution.	It selects those processes which are ready to execute	It can re-introduce the process into memory and execution can be continued.

Two-State Process Model Short-Term

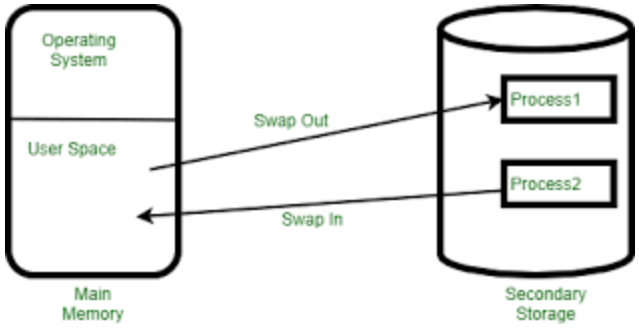
The terms “running” and “non-running” states are used to describe the two-state process model.

- **Running:** A newly created process joins the system in a running state when it is created.
- **Not Running:** Processes that are not currently running are kept in a queue and await execution. A pointer to a specific process is contained in each entry in the queue. Linked lists are used to implement the queue system. This is how the dispatcher is used. When a process is stopped, it is moved to the back of the waiting queue. The process is discarded depending on whether it succeeded or failed. The dispatcher then chooses a process to run from the queue in either scenario.

Context Switching

In order for a process execution to be continued from the same point at a later time, context switching is a mechanism to store and restore the state or context of a CPU in the Process Control block. A context switcher makes it possible for multiple processes to share a single CPU using this method. A multitasking operating system must include context switching among its features.

The state of the currently running process is saved into the process control block when the scheduler switches the CPU from executing one process to another. The state used to set the computer, registers, etc. for the process that will run next is then loaded from its own PCB. After that, the second can start processing.



Context Switching

In order for a process execution to be continued from the same point at a later time, context switching is a mechanism to store and restore the state or context of a CPU in the Process Control block. A context switcher makes it possible for multiple processes to share a single CPU using this method. A multitasking operating system must include context switching among its features.

- Program Counter
- Scheduling information
- The base and limit register value
- Currently used register
- Changed State
- I/O State information
- Accounting information

Conclusion

In conclusion, process schedulers are essential parts of operating systems that manage how the CPU handles multiple tasks or processes. They ensure that processes are executed efficiently, making the best use of CPU resources and maintaining system responsiveness. By choosing the right process to run at the right time, schedulers help optimize overall system performance, improve user experience, and ensure fair access to CPU resources among competing processes.

Frequently Asked Questions on Process Scheduling – FAQs

What is CPU scheduling in OS?

In an operating system, CPU scheduling refers to a technique that permits one process to utilize the CPU while keeping the other programs waiting or put on hold.

What is Inter-Process Communication (IPC)?

IPC is an operating system technique that facilitates data sharing, synchronization, and communication between processes.

What is PCB in OS?

The operating system uses a data structure called a Process Control Block (PCB) to store and handle process-related data.

Previous Article

Operations on Processes

Next Article

Inter Process Communication (IPC)

Similar Reads

Two State Process Model in Operating System

Pre-requisite: States of a Process in Operating Systems The process in an operating system passes from different states starting from its formation t...

3 min read

Three State Process Model in Operating System

Pre-requisites: States of a Process in Operating Systems, Two-State Process Model in Operating SystemIn this article, we'll be discussing a...

4 min read

Cooperating Process in Operating System

Pre-requisites: Process Synchronization In an operating system, everything is around the process. How the process goes through several different...

2 min read

Six-State Process Model in Operating System

In this article, we are going to discuss the Six-State Process Model in Operating Systems and we will also understand what was the need for...

4 min read

Lottery Process Scheduling in Operating System

Prerequisite - CPU Scheduling, Process Management Lottery Scheduling is a type of process scheduling, somewhat different from other Scheduling....

4 min read

View More Articles

Article Tags : Android Linked List Operating Systems C++-Misc C++ +10 More

Practice Tags : Functions Linked List