

# Commonly Asked Operating Systems Interview Questions | Set 1

- Difficulty Level : [Medium](#)
- Last Updated : 28 Jun, 2021

## What is a process and process table? What are different states of process

A *process* is an instance of program in execution. For example a Web Browser is a process, a shell (or command prompt) is a process.

The operating system is responsible for managing all the processes that are running on a computer and allocated each process a certain amount of time to use the processor. In addition, the operating system also allocates various other resources that processes will need such as computer memory or disks. To keep track of the state of all the processes, the operating system maintains a table known as the *process table*. Inside this table, every process is listed along with the resources the processes is using and the current state of the process.

*Processes can be in one of three states: running, ready, or waiting.* The running state means that the process has all the resources it need for execution and it has been given permission by the operating system to use the processor. Only one process can be in the running state at any given time. The remaining processes are either in a waiting state (i.e., waiting for some external event to occur such as user input or a disk access) or a ready state (i.e., waiting for permission to use the processor). In a real operating system, the waiting and ready states are implemented as queues which hold the processes in these states. The animation below shows a simple representation of the life cycle of a process

(Source: <http://courses.cs.vt.edu/csonline/OS/Lessons/Processes/index.html>)

## What is a Thread? What are the differences between process and thread?

A thread is a single sequence stream within in a process. Because threads have some of the properties of processes, they are sometimes called *lightweight processes*. Threads are popular way to improve application through parallelism. For example, in a browser, multiple tabs can be different threads. MS word uses multiple threads, one thread to format the text, other thread to process inputs, etc.

A thread has its own program counter (PC), a register set, and a stack space. Threads are not independent of one other like processes as a result threads shares with other threads their code section, data section and OS resources like open files and signals. See <http://www.personal.kent.edu/~rmuhamma/OpSystems/Myos/threads.htm> for more details.

## What are the benefits of multithreaded programming?

It makes the system more responsive and enables resource sharing. It leads to the use of multiprocess architecture. It is more economical and preferred.

## What are the different scheduling algorithms

*First-Come, First-Served (FCFS) Scheduling.*

*Shortest-Job-Next (SJN) Scheduling.*

*Priority Scheduling.*

*Shortest Remaining Time.*

*Round Robin(RR) Scheduling.*

*Multiple-Level Queues Scheduling.*

## What is deadlock?

Deadlock is a situation when two or more processes wait for each other to finish and none of them ever finish. Consider an example when two trains are coming toward each other on same track and there is only one track, none of the trains can move once they are in front of each other. Similar situation occurs in operating systems when there are two or more processes hold some resources and wait for resources held by other(s).

## What are the necessary conditions for deadlock?

*Mutual Exclusion:* There is a resource that cannot be shared.

*Hold and Wait:* A process is holding at least one resource and waiting for another resource which is with some other process.

*No Preemption:* The operating system is not allowed to take a resource back from a process until process gives it back.

*Circular Wait:* A set of processes are waiting for each other in circular form.

**What is Virtual Memory? How is it implemented?**

Virtual memory creates an illusion that each user has one or more contiguous address spaces, each beginning at address zero. The sizes of such virtual address spaces is generally very high. The idea of virtual memory is to use disk space to extend the RAM. Running processes don't need to care whether the memory is from RAM or disk. The illusion of such a large amount of memory is created by subdividing the virtual memory into smaller pieces, which can be loaded into physical memory whenever they are needed by a process.

**What is Thrashing?**

Thrashing is a situation when the performance of a computer degrades or collapses. Thrashing occurs when a system spends more time processing page faults than executing transactions. While processing page faults is necessary in order to appreciate the benefits of virtual memory, thrashing has a negative affect on the system. As the page fault rate increases, more transactions need processing from the paging device. The queue at the paging device increases, resulting in increased service time for a page fault (Source: <http://cs.gmu.edu/cne/modules/vm/blue/thrash.html>)

**What is Belady's Anomaly?**

Bélády's anomaly is an anomaly with some page replacement policies where increasing the number of page frames results in an increase in the number of page faults. It occurs with First in First Out page replacement is used. See [the wiki page](#) for an example and more details.

**Differences between mutex and semaphore?**

See <https://www.geeksforgeeks.org/mutex-vs-semaphore/>

- Practice [Quizzes](#) on Operating System topics
- [Last Minute Notes](#) – OS
- OS [articles](#)

**[Last Minute Notes – Operating Systems](#)**

We will soon be covering more Operating System questions. Please write comments if you find anything incorrect, or you want to share more information about the topic discussed above.

Attention reader! Don't stop learning now. Get hold of all the important CS Theory concepts for SDE interviews with the [CS Theory Course](#) at a student-friendly price and become industry ready.