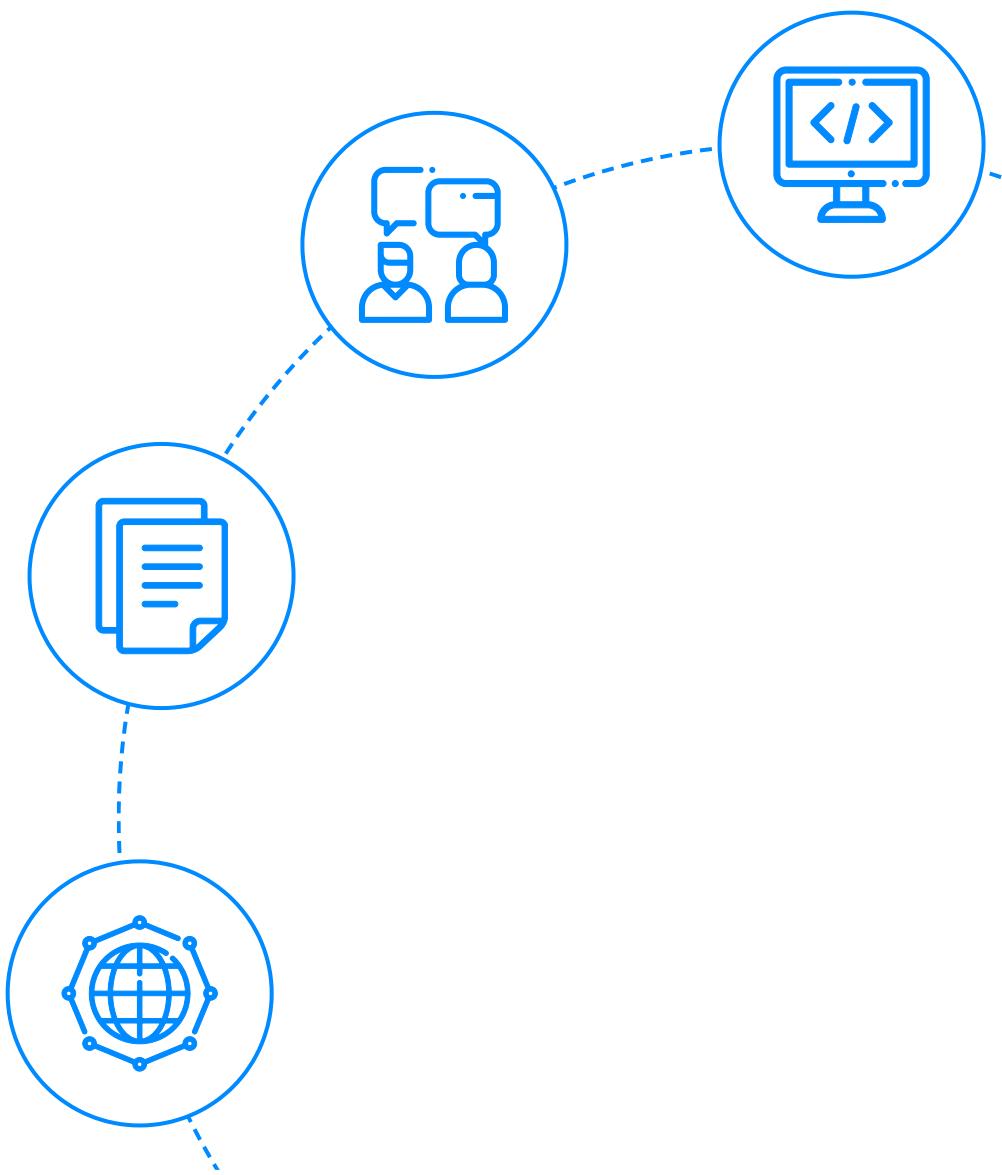




Operating System Interview Questions



To view the live version of the page, [click here](#).

Contents

Basic OS Interview Questions

1. Why is the operating system important?
2. What's the main purpose of an OS? What are the different types of OS?
3. What are the benefits of a multiprocessor system?
4. What is RAID structure in OS? What are the different levels of RAID configuration?
5. What is GUI?
6. What is a Pipe and when it is used?
7. What are the different kinds of operations that are possible on semaphore?
8. What is a bootstrap program in OS?
9. Explain demand paging?
10. What do you mean by RTOS?
11. What do you mean by process synchronization?
12. What is IPC? What are the different IPC mechanisms?
13. What is different between main memory and secondary memory.
14. What do you mean by overlays in OS?
15. Write top 10 examples of OS?

Intermediate OS Interview Questions

16. What is virtual memory?
17. What is thread in OS?
18. What is a process? What are the different states of a process?

Intermediate OS Interview Questions

(.....Continued)

19. What do you mean by FCFS?
20. What is Reentrancy?
21. What is a Scheduling Algorithm? Name different types of scheduling algorithms.
22. What is the difference between paging and segmentation?
23. What is thrashing in OS?
24. What is the main objective of multiprogramming?
25. What do you mean by asymmetric clustering?
26. What is the difference between multitasking and multiprocessing OS?
27. What do you mean by Sockets in OS?
28. Explain zombie process?
29. What do you mean by cascading termination?
30. What is starvation and aging in OS?

Advanced OS Interview Questions

31. What do you mean by Semaphore in OS? Why is it used?
32. What is Kernel and write its main functions?
33. What are different types of Kernel?
34. Write difference between micro kernel and monolithic kernel?
35. What is SMP (Symmetric Multiprocessing)?
36. What is a time-sharing system?

Advanced OS Interview Questions

(.....Continued)

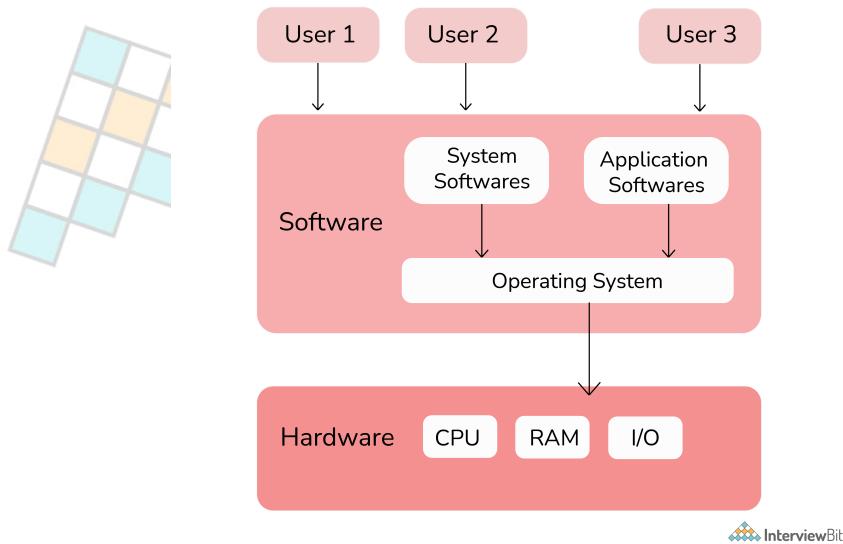
37. What is Context Switching?
38. What is difference between Kernel and OS?
39. What is difference between process and thread?
40. What are various sections of the process?
41. What is a deadlock in OS? What are the necessary conditions for a deadlock?
42. What do you mean by Belady's Anomaly?
43. What is spooling in OS?



Let's get Started

What do you mean by an operating system? What are its basic functions?

Operating System (OS) is basically a software program that manages and handles all resources of a computer such as hardware and software. The first OS was introduced in the early 1950s known as GMOs. An OS is responsible for managing, handling, and coordinating overall activities and sharing of computer resources. It acts as an intermediary among users of computer and computer hardware.



Functions of OS:

There are many functions of the OS. Some of the important functions of OS are given below:

- Memory and Processor Management
- Providing user interface to users
- File Management and Device Management
- Scheduling of resources and jobs
- Error Detection
- Security

Basic OS Interview Questions

1. Why is the operating system important?

OS is the most essential and vital part of a computer without which it is considered useless. It enables an interface or acts like a link for interaction between computer software that is installed on OS and users. It also helps to communicate with hardware and also maintains balance among hardware and CPU. It also provides services to users and a platform for programs to run on. It performs all common tasks applications require.

2. What's the main purpose of an OS? What are the different types of OS?

The main purpose of an OS is to execute user programs and make it easier for users to understand and interact with computers as well as run applications. It is specially designed to ensure that the computer system performs better by managing all computational activities. It also manages computer memory, processes, and operation of all hardware and software.

Types of OS:

- Batched OS (Example: Payroll System, Transactions Process, etc.)
- Multi-Programmed OS (Example: Windows O/S, UNIX O/S, etc.)
- Timesharing OS (Example: Multics, etc.)
- Distributed OS (LOCUS, etc.)
- Real-Time OS (PSOS, VRTX, etc.)

3. What are the benefits of a multiprocessor system?

A Multiprocessor system is a type of system that includes two or more CPUs. It involves the processing of different computer programs at the same time mostly by a computer system with two or more CPUs that are sharing single memory.

Benefits:

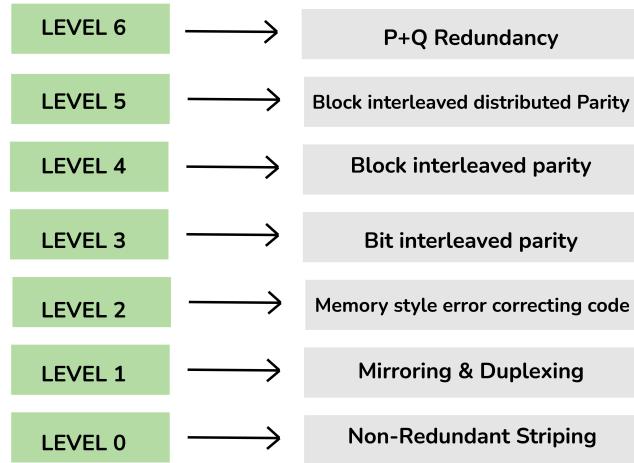
- Such systems are used widely nowadays to improve performance in systems that are running multiple programs concurrently.
- By increasing the number of processors, a greater number of tasks can be completed in unit time.
- One also gets a considerable increase in throughput and is cost-effective also as all processors share the same resources.
- It simply improves the reliability of the computer system.

4. What is RAID structure in OS? What are the different levels of RAID configuration?

RAID (Redundant Arrays of Independent Disks) is a method used to store data on Multiple hard disks therefore it is considered as data storage virtualization technology that combines multiple hard disks. It simply balances data protection, system performance, storage space, etc. It is used to improve the overall performance and reliability of data storage. It also increases the storage capacity of the system and its main purpose is to achieve data redundancy to reduce data loss.

Different levels of RAID

Nowadays, RAID is available in various schemes or RAID level as given below:



- RAID 0 - Non-redundant striping: This level is used to increase the performance of the server.
- RAID 1 - Mirroring and duplexing: This level is also known as disk mirroring and is considered the simplest way to implement fault tolerance.
- RAID 2 - Memory-style error-correcting codes: This level generally uses dedicated hamming code parity i.e., a liner form of error correction code.
- RAID 3 - Bit-interleaved Parity: This level requires a dedicated parity drive to store parity information.
- RAID 4 - Block-interleaved Parity: This level is similar to RAID 5 but the only difference is that this level confines all parity data to a single drive.
- RAID 5 - Block-interleaved distributed Parity: This level provides far better performance than disk mirroring and fault tolerance.
- RAID 6 - P+Q Redundancy: This level generally provides fault tolerance for two drive failures.

5. What is GUI?

GUI (Graphical User Interface) is basically a type of user interface that allows users to use graphics to interact with OS. GUI is created because it is more user-friendly, less complex, and easier to understand rather than a command-line interface. Its main goal is to increase efficiency and ease of use. Instead of having to memorize commands, users can just click on a button to simply execute the procedure. Examples of GUI include Microsoft Windows, macOS, Apple's iOS, etc.

6. What is a Pipe and when it is used?

The pipe is generally a connection among two or more processes that are interrelated to each other. It is a mechanism that is used for inter-process communication using message passing. One can easily send information such as the output of one program process to another program process using a pipe. It can be used when two processes want to communicate one-way i.e., inter-process communication (IPC).

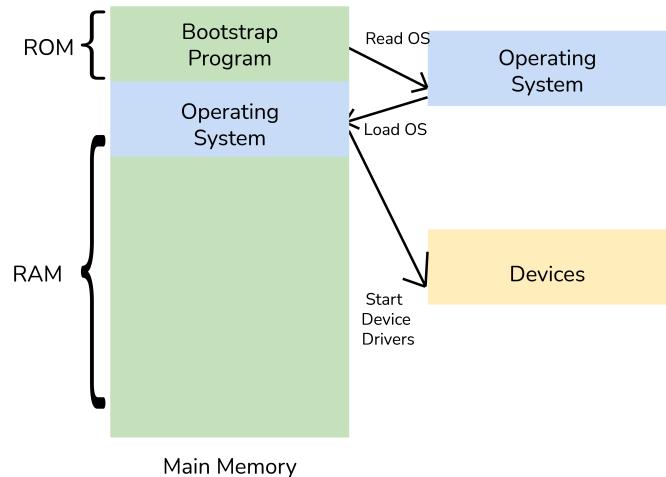
7. What are the different kinds of operations that are possible on semaphore?

There are basically two atomic operations that are possible:

- Wait()
- Signal()

8. What is a bootstrap program in OS?

It is generally a program that initializes OS during startup i.e., first code that is executed whenever computer system startups. OS is loaded through a bootstrapping process or program commonly known as booting. Overall OS only depends on the bootstrap program to perform and work correctly. It is fully stored in boot blocks at a fixed location on the disk. It also locates the kernel and loads it into the main memory after which the program starts its execution.



9. Explain demand paging?

Demand paging is a method that loads pages into memory on demand. This method is mostly used in virtual memory. In this, a page is only brought into memory when a location on that particular page is referenced during execution. The following steps are generally followed:

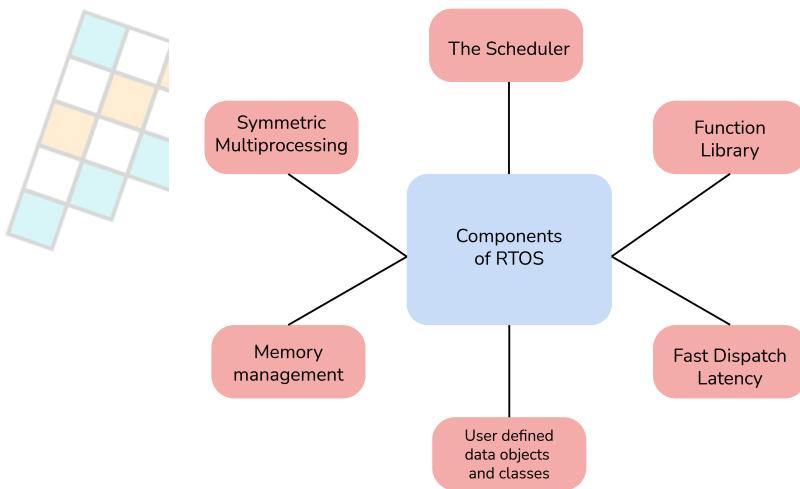
- Attempt to access the page.
- If the page is valid (in memory) then continue processing instructions as normal.
- If a page is invalid then a **page-fault trap** occurs.
- Check if the memory reference is a valid reference to a location on secondary memory. If not, the process is terminated (**illegal memory access**). Otherwise, we have to **page in** the required page.
- Schedule disk operation to read the desired page into main memory.
- Restart the instruction that was interrupted by the operating system trap.

10. What do you mean by RTOS?

Real Time Operating System (RTOS) is an operating system that is used for real-time applications i.e., for those applications where data processing should be done in a fixed and small measure of time. It performs much better on tasks that are needed to be executed within a short time. It also takes care of execution, monitoring, and all-controlling processes. It also occupies less memory and consumes fewer resources.

Types of RTOS:

- Hard Real-Time
- Firm Real-Time
- Soft Real-Time



RTOS is used in Air traffic control systems, Anti-lock Brake Systems, and Heart pacemakers.

11. What do you mean by process synchronization?

Process synchronization is basically a way to coordinate processes that use shared resources or data. It is very much essential to ensure synchronized execution of cooperating processes so that will maintain data consistency. Its main purpose is to share resources without any interference using mutual exclusion. There are two types of process synchronization:

- Independent Process
- Cooperative Process

12. What is IPC? What are the different IPC mechanisms?

IPC (Interprocess Communication) is a mechanism that requires the use of resources like a memory that is shared between processes or threads. With IPC, OS allows different processes to communicate with each other. It is simply used for exchanging data between multiple threads in one or more programs or processes. In this mechanism, different processes can communicate with each other with the approval of the OS.

Different IPC Mechanisms:

- Pipes
- Message Queuing
- Semaphores
- Socket
- Shared Memory
- Signals

13. What is different between main memory and secondary memory.

Main memory: Main memory in a computer is RAM (Random Access Memory). It is also known as primary memory or read-write memory or internal memory. The programs and data that the CPU requires during the execution of a program are stored in this memory.

Secondary memory: Secondary memory in a computer are storage devices that can store data and programs. It is also known as external memory or additional memory or backup memory or auxiliary memory. Such storage devices are capable of storing high-volume data. Storage devices can be hard drives, USB flash drives, CDs, etc.

Primary Memory	Secondary Memory
Data can be directly accessed by the processing unit.	Firstly, data is transferred to primary memory and after then routed to the processing unit.
It can be both volatile and non-volatile in nature.	It is non-volatile in nature.
It is more costly than secondary memory.	It is more cost-effective or less costly than primary memory.
It is temporary because data is stored temporarily.	It is permanent because data is stored permanently.
In this memory, data can be lost whenever there is a power failure.	In this memory, data is stored permanently and therefore cannot be lost even in case of power failure.
It is much faster than secondary memory and saves data that is currently used by the computer.	It is slower as compared to primary memory and saves different kinds of data in different formats.
It can be accessed by data.	It can be accessed by I/O channels.

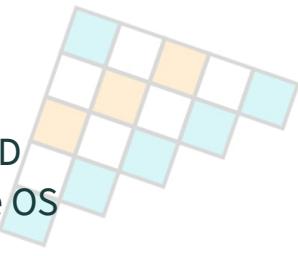
14. What do you mean by overlays in OS?

Overlays is basically a programming method that divides processes into pieces so that instructions that are important and need can be saved in memory. It does not need any type of support from the OS. It can run programs that are bigger in size than physical memory by only keeping only important data and instructions that can be needed at any given time.

15. Write top 10 examples of OS?

Some of the top OS's that are used mostly are given below:

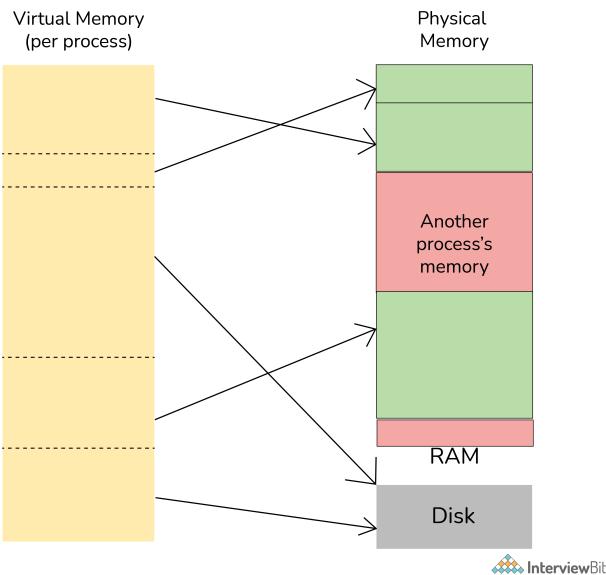
- MS-Windows
- Ubuntu
- Mac OS
- Fedora
- Solaris
- Free BSD
- Chrome OS
- CentOS
- Debian
- Android



Intermediate OS Interview Questions

16. What is virtual memory?

It is a memory management technique feature of OS that creates the illusion to users of a very large (main) memory. It is simply space where a greater number of programs can be stored by themselves in the form of pages. It enables us to increase the use of physical memory by using a disk and also allows us to have memory protection. It can be managed in two common ways by OS i.e., paging and segmentation. It acts as temporary storage that can be used along with RAM for computer processes.

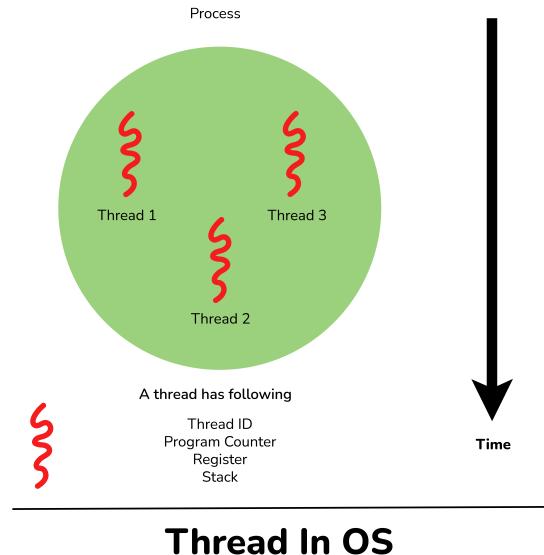


17. What is thread in OS?

Thread is a path of execution that is composed of a program counter, thread id, stack, and set of registers within the process. It is a basic unit of CPU utilization that makes communication more effective and efficient, enables utilization of multiprocessor architectures to a greater scale and greater efficiency, and reduces the time required in context switching. It simply provides a way to improve and increase the performance of applications through parallelism. Threads are sometimes called **lightweight processes** because they have their own stack but can access shared data.

Multiple threads running in a process share: Address space, Heap, Static data, Code segments, File descriptors, Global variables, Child processes, Pending alarms, Signals, and signal handlers.

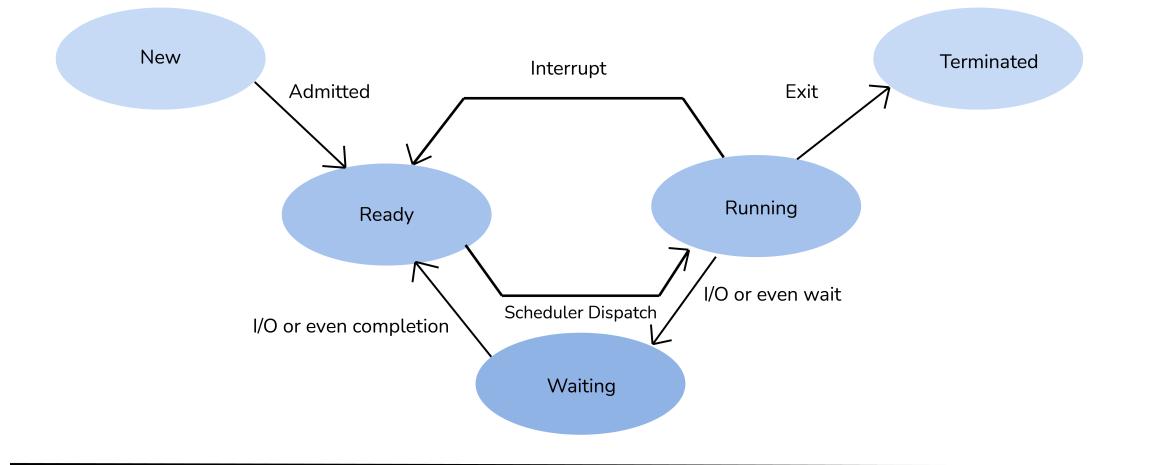
Each thread has its own: Program counter, Registers, Stack, and State.



18. What is a process? What are the different states of a process?

The process is basically a program that is currently under execution. The main function of an OS is to manage and handle all of these processes. When a program is loaded into the memory and it becomes a process, it can be divided into four sections—stack, heap, text, and data. There are two types of processes:

1. Operating System Processes
2. User Processes



Process State

States of Process:

Different states of the process through which process goes are given below:

- **New State:** In this state, a process is just created.
- **Running:** In this state, the CPU starts working on the process's instructions.
- **Waiting:** In this state, the process cannot run because it just waits for some event to occur
- **Ready:** In this state, the process has all resources available that are required to run but it waits to get assigned to a processor because CPUs are not working currently on instructions passed by the process.
- **Terminate:** In this state, the process is completed i.e., the process has finished execution.

19. What do you mean by FCFS?

FCFS (First Come First Serve) is a type of OS scheduling algorithm that executes processes in the same order in which processes arrive. In simple words, the process that arrives first will be executed first. It is non-preemptive in nature. FCFS scheduling may cause the problem of starvation if the burst time of the first process is the longest among all the jobs. Burst time here means the time that is required in milliseconds by the process for its execution. It is also considered the easiest and simplest OS scheduling algorithm as compared to others. Implementation of FCFS is generally managed with help of the FIFO (First In First Out) queue.

20. What is Reentrancy?

Reentrant is simply a function in which various clients can use and shares a single copy of a program during a similar period. This concept is generally associated with OS code and does not deal with concurrency. It has two major functions:

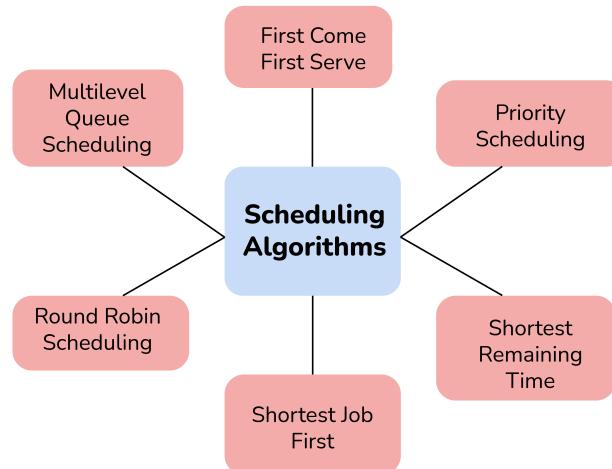
- Program code cannot change or modify itself.
- Local data for every client process needs to be stored in different disks.

21. What is a Scheduling Algorithm? Name different types of scheduling algorithms.

A scheduling algorithm is a process that is used to improve efficiency by utilizing maximum CPU and providing minimum waiting time to tasks. It simply deals with the problem of deciding which of outstanding requests is to be allocated resources. Its main aim is to reduce resource starvation and to ensure fairness amongst parties that are utilizing the resources. In simple words, it is used to allocate resources among various competing tasks.

Types of Scheduling Algorithm

There are different types of scheduling algorithms as given below:



22. What is the difference between paging and segmentation?

Paging: It is generally a memory management technique that allows OS to retrieve processes from secondary storage into main memory. It is a non-contiguous allocation technique that divides each process in the form of pages.

Segmentation: It is generally a memory management technique that divides processes into modules and parts of different sizes. These parts and modules are known as segments that can be allocated to process.

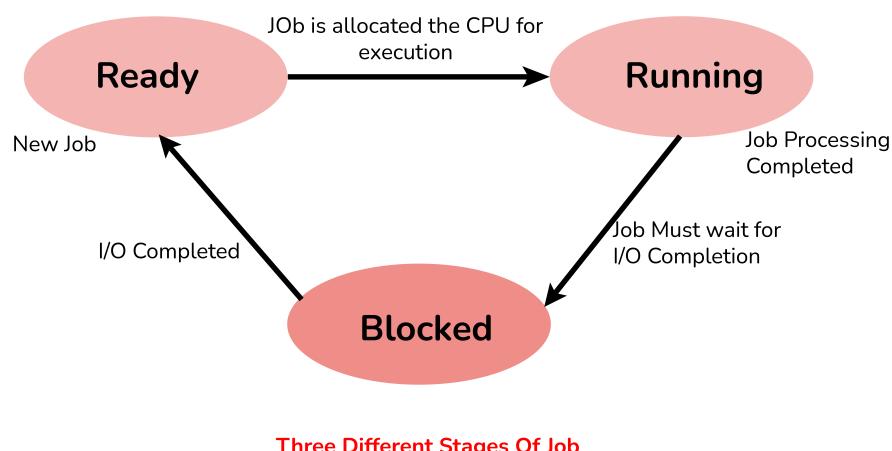
Paging	Segmentation
It is invisible to a programmer.	It is visible to a programmer.
In this, the size of pages is fixed.	In this, the size of segments is not fixed.
Procedures and data cannot be separated in paging.	Procedures and data can be separated in segmentation.
It allows a cumulative total of virtual address spaces to cross physical main memory.	It allows all programs, data, and codes to break up into independent address spaces.
It is mostly available on CPUs and MMU chips.	It is mostly available on Windows servers that may support backward compatibility, while Linux has limited support.
It is faster for memory access as compared to segmentation.	It is slower as compared to paging.
In this, OS needs to maintain a free frame.	In this, OS needs to maintain a list of holes in the main memory.
In paging, the type of fragmentation is internal.	In segmentation, the type of fragmentation is external.
The size of the page is determined by available memory.	The size of the page is determined by the user.

23. What is thrashing in OS?

It is generally a situation where the CPU performs less productive work and more swapping or paging work. It spends more time swapping or paging activities rather than its execution. By evaluating the level of CPU utilization, a system can detect thrashing. It occurs when the process does not have enough pages due to which the page-fault rate is increased. It inhibits much application-level processing that causes computer performance to degrade or collapse.

24. What is the main objective of multiprogramming?

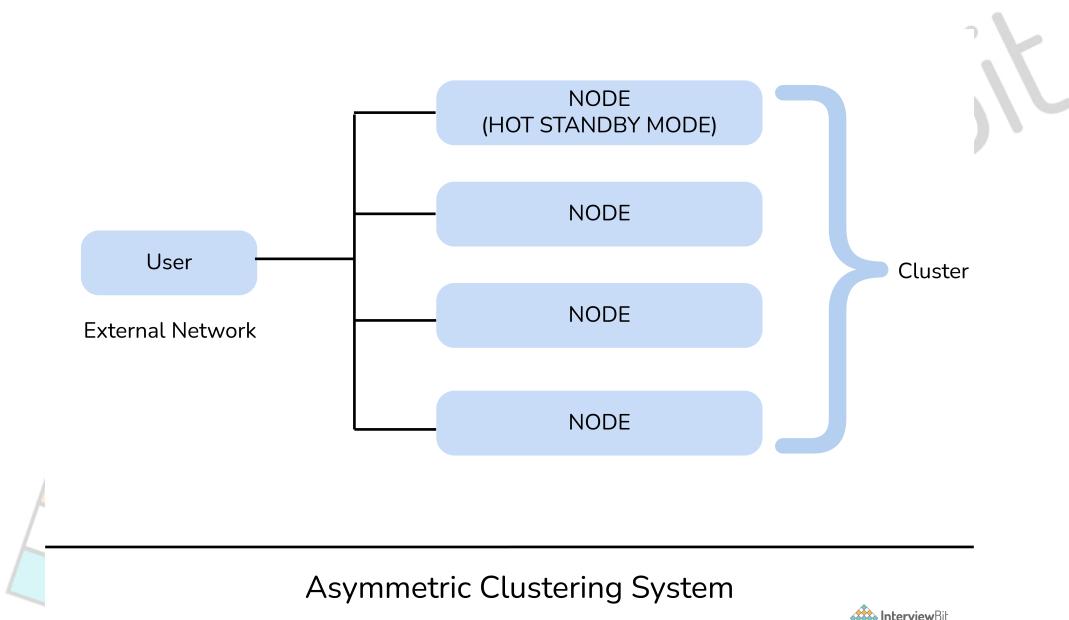
It refers to the ability to execute or perform more than one program on a single processor machine. This technique was introduced to overcome the problem of underutilization of CPU and main memory. In simple words, it is the coordination of execution of various programs simultaneously on a single processor (CPU). The main objective of multiprogramming is to have at least some processes running at all times. It simply improves the utilization of the CPU as it organizes many jobs where the CPU always has one to execute.



Multiprogramming System

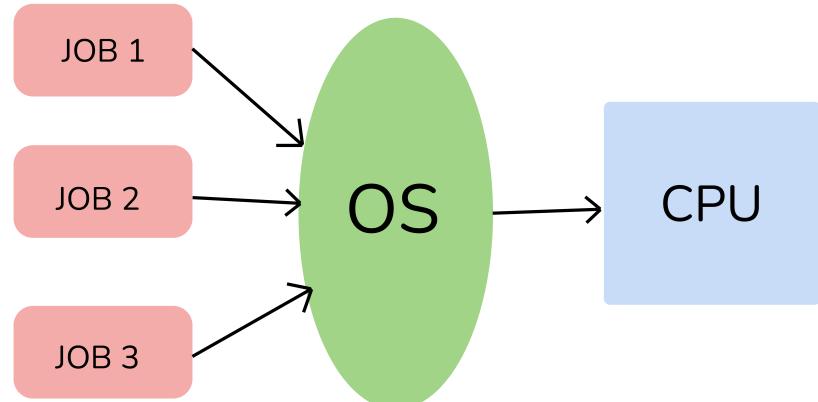
25. What do you mean by asymmetric clustering?

Asymmetric Clustering is generally a system in which one of the nodes among all nodes is in hot standby mode whereas the rest of all nodes run different applications. It simply uses whole or entire hardware resources therefore it is considered a more reliable system as compared to others.



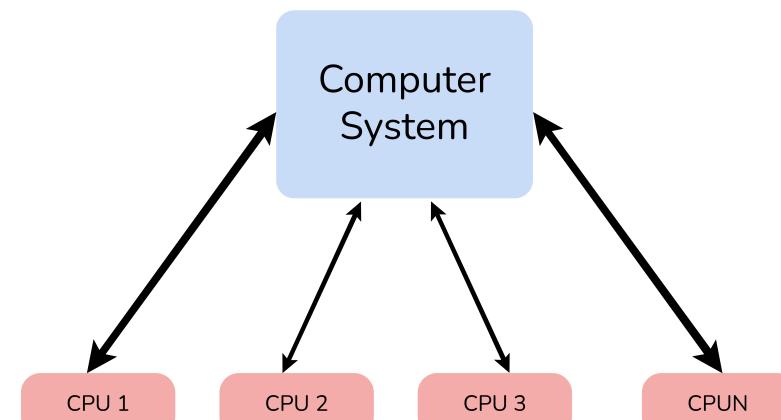
26. What is the difference between multitasking and multiprocessing OS?

Multitasking: It is a system that allows more efficient use of computer hardware. This system works on more than one task at one time by rapidly switching between various tasks. These systems are also known as time-sharing systems.



Multitasking

Multiprocessing: It is a system that allows multiple or various processors in a computer to process two or more different portions of the same program simultaneously. It is used to complete more work in a shorter period of time.



Multiprocessing

Multitasking	Multiprocessing
It performs more than one task at a time using a single processor.	It performs more than one task at a time using multiple processors.
In this, the number of CPUs is only one.	In this, the number of CPUs is more than one.
It is more economical.	It is less economical.
It is less efficient than multiprocessing.	It is more efficient than multitasking.
It allows fast switching among various tasks.	It allows smooth processing of multiple tasks at once.
It requires more time to execute tasks as compared to multiprocessing.	It requires less time for job processing as compared to multitasking.

27. What do you mean by Sockets in OS?

The socket in OS is generally referred to as an endpoint for IPC (Interprocess Communication). Here, the endpoint is referred to as a combination of an IP address and port number. Sockets are used to make it easy for software developers to create network-enabled programs. It also allows communication or exchange of information between two different processes on the same or different machines. It is mostly used in client-server-based systems.

Types of Sockets

There are basically four types of sockets as given below:

- Stream Sockets
- Datagram Sockets
- Sequenced Packet Sockets
- Raw Sockets

28. Explain zombie process?

Zombie process, referred to as a defunct process, is basically a process that is terminated or completed but the whole process control block is not cleaned up from the main memory because it still has an entry in the process table to report to its parent process. It does not consume any of the resources and is dead, but it still exists. It also shows that resources are held by process and are not free.

29. What do you mean by cascading termination?

Cascading termination is a process termination in which if the parent process is exiting or terminating then the children process will also get terminated. It does not allow the child to continue processing as its parent process terminates. It is generally initiated by OS.

30. What is starvation and aging in OS?

Starvation: It is generally a problem that usually occurs when a process has not been able to get the required resources it needs for progress with its execution for a long period of time. In this condition, low priority processes get blocked and only high priority processes proceed towards completion because of which low priority processes suffer from lack of resources.

Aging: It is a technique that is used to overcome the situation or problem of starvation. It simply increases the priority of processes that wait in the system for resources for a long period of time. It is considered the best technique to resolve the problem of starvation as it adds an aging factor to the priority of each and every request by various processes for resources. It also ensures that low-level queue jobs or processes complete their execution.

Advanced OS Interview Questions

31. What do you mean by Semaphore in OS? Why is it used?

Semaphore is a signaling mechanism. It only holds one positive integer value. It is simply used to solve the problem or issue of critical sections in the synchronization process by using two atomic operations i.e., wait() and signal().

Types of Semaphore

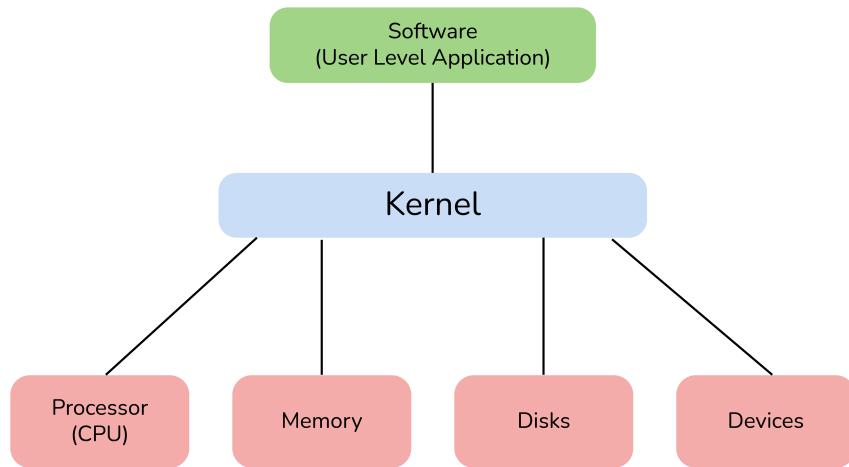
There are usually two types of semaphores as given below:

- Binary Semaphore
- Counting Semaphore

Binary Semaphore	Mutex
It allows various process threads to get the finite instance of the resource until resources are available.	It allows various process threads to get single shared resource only at a time.
Its functions are based upon signaling mechanisms.	Its functions are based upon a locking mechanism.
Binary semaphores are much faster as compared to Mutex.	Mutex is slower as compared to binary semaphores.
It is basically an integer.	It is basically an object.

32. What is Kernel and write its main functions?

The kernel is basically a computer program usually considered as a central component or module of OS. It is responsible for handling, managing, and controlling all operations of computer systems and hardware. Whenever the system starts, the kernel is loaded first and remains in the main memory. It also acts as an interface between user applications and hardware.



Functions of Kernel:

- It is responsible for managing all computer resources such as CPU, memory, files, processes, etc.
- It facilitates or initiates the interaction between components of hardware and software.
- It manages RAM memory so that all running processes and programs can work effectively and efficiently.
- It also controls and manages all primary tasks of the OS as well as manages access and use of various peripherals connected to the computer.
- It schedules the work done by the CPU so that the work of each user is executed as efficiently as possible.

33. What are different types of Kernel?

There are basically five types of Kernels as given below:

- Monolithic Kernel
- MicroKernel
- Hybrid Kernel
- Nano Kernel
- Exo Kernel

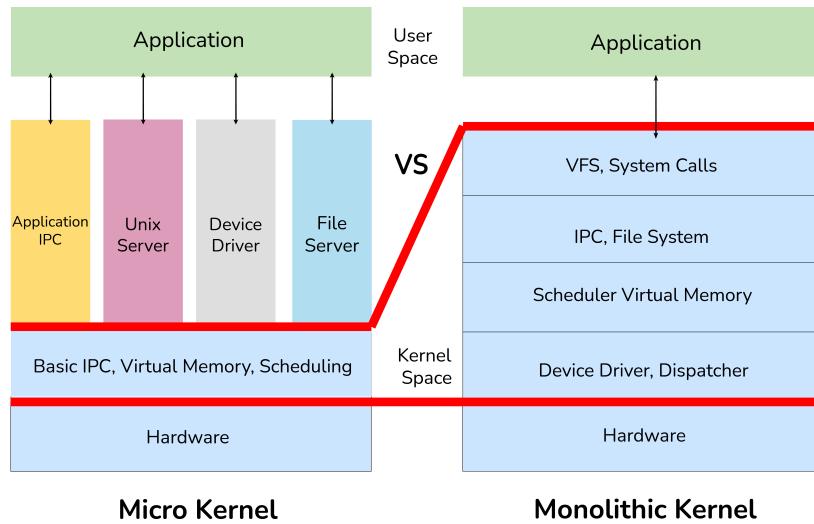
34. Write difference between micro kernel and monolithic kernel?

MicroKernel: It is a minimal OS that executes only important functions of OS. It only contains a near-minimum number of features and functions that are required to implement OS.

Example: QNX, Mac OS X, K42, etc.

Monolithic Kernel: It is an OS architecture that supports all basic features of computer components such as resource management, memory, file, etc.

Example: Solaris, DOS, OpenVMS, Linux, etc.



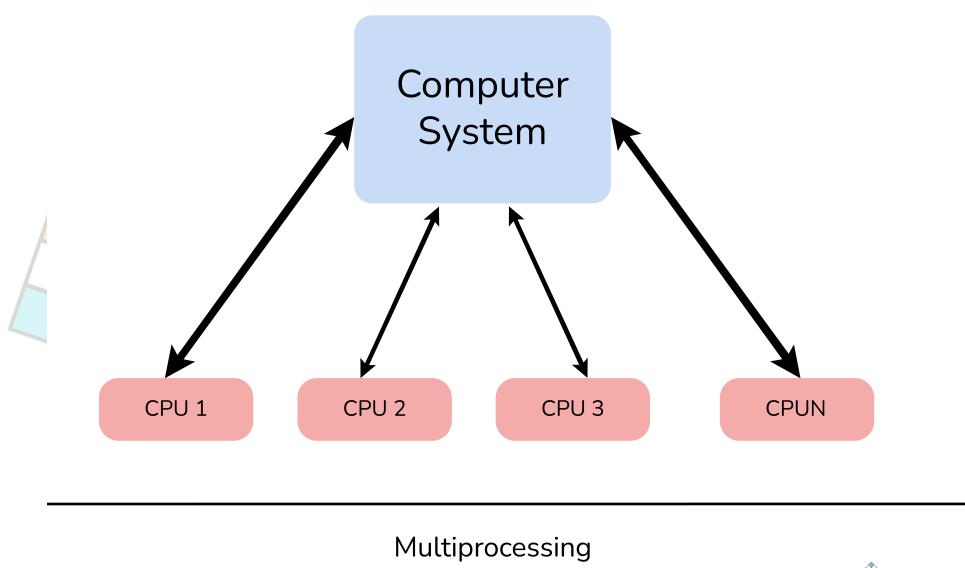
MicroKernel	Monolithic Kernel
In this software or program, kernel services and user services are present in different address spaces.	In this software or program, kernel services and user services are usually present in the same address space.
It is smaller in size as compared to the monolithic kernel.	It is larger in size as compared to a microkernel.
It is easily extendible as compared to a monolithic kernel.	It is hard to as extend as compared to a microkernel.
If a service crashes, it does affect on working of the microkernel.	If a service crashes, the whole system crashes in a monolithic kernel.
It uses message queues to achieve inter-process communication.	It uses signals and sockets to achieve inter-process communication.

35. What is SMP (Symmetric Multiprocessing)?

SMP is generally referred to as computer architecture in which the processing of programs is done by multiple processors that share a common OS and memory. SMP is very much required if you want to take advantage of multiprocessor hardware. It simply enables any processor to work on any of the tasks no matter where data or resources for that particular task are located in memory. These systems are more reliable than single-processor systems.

36. What is a time-sharing system?

It is a system that allows more than one user to access the resources of a particular system in many locations. In simple words, it performs multiple tasks on a single processor or CPU. As the name suggests, it means to share time into multiple slots in several processes. It also allows different users from different locations to use a particular computer system at the same time therefore it is considered one of the important types of OS.



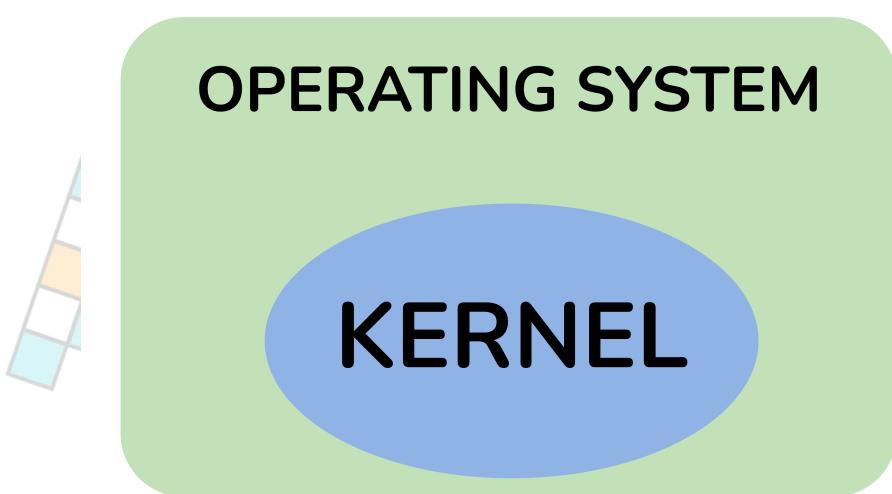
37. What is Context Switching?

Context switching is basically a process of saving the context of one process and loading the context of another process. It is one of the cost-effective and time-saving measures executed by CPU because it allows multiple processes to share a single CPU. Therefore, it is considered an important part of a modern OS. This technique is used by OS to switch a process from one state to another i.e., from running state to ready state. It also allows a single CPU to handle and control various different processes or threads without even the need for additional resources.

38. What is difference between Kernel and OS?

Kernel: Kernel is a system program that controls all programs running on the computer. The kernel is basically a bridge between the software and hardware of the system.

Operating System: Operating system is a system program that runs on the computer to provide an interface to the computer user so that they can easily operate on the computer.

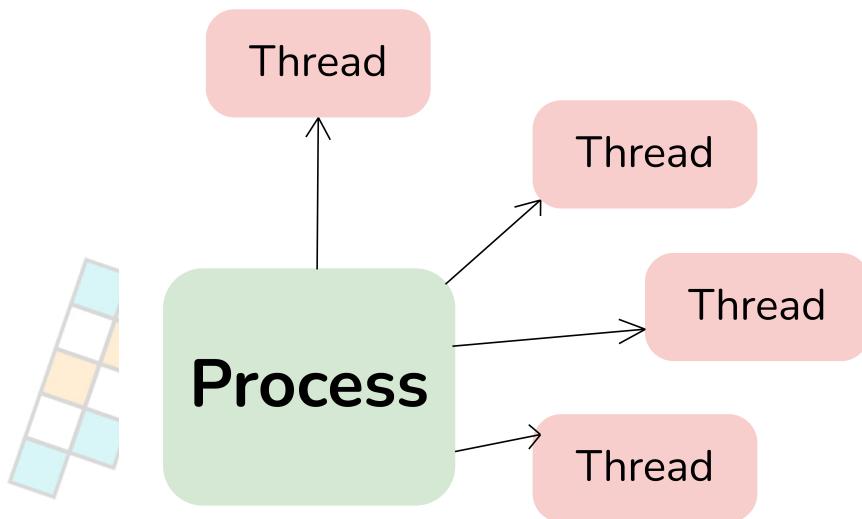


Kernel	OS
It is considered a central component of OS	It is considered system software.
It is generally responsible for converting user commands into machine-level commands.	It is generally responsible for managing the resources of system.
It simply acts as an interface between hardware and applications.	It simply acts as an interface between hardware and user.
It also performs functions like process management, file management, device management, I/O communication, etc.	It also performs functions like providing security to data and files in the system, providing access controls to users, maintaining the system privacy, etc.
Its type includes Microkernel, Monolithic kernel, etc.	Its type includes Single and Multiprogramming batch systems, Distributed OS, Real-time OS.

39. What is difference between process and thread?

Process: It is basically a program that is currently under execution by one or more threads. It is a very important part of the modern-day OS.

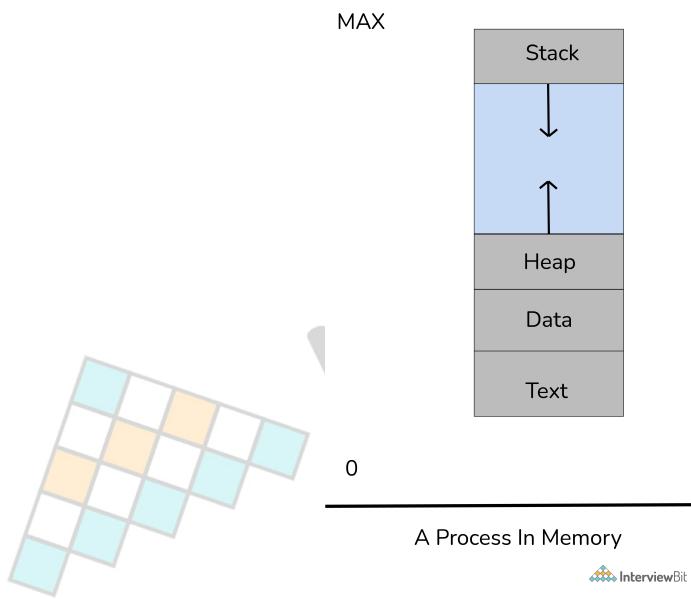
Thread: It is a path of execution that is composed of the program counter, thread id, stack, and set of registers within the process.



Process	Thread
It is a computer program that is under execution.	It is the component or entity of the process that is the smallest execution unit.
These are heavy-weight operators.	These are lightweight operators.
It has its own memory space.	It uses the memory of the process they belong to.
It is more difficult to create a process as compared to creating a thread.	It is easier to create a thread as compared to creating a process.
It requires more resources as compared to thread.	It requires fewer resources as compared to processes.
It takes more time to create and terminate a process as compared to a thread.	It takes less time to create and terminate a thread as compared to a process.
It usually run-in separate memory space.	It usually run-in shared memory space.
It does not share data.	It shares data with each other.
It can be divided into multiple threads.	It can't be further subdivided.

40. What are various sections of the process?

There are basically four sections in the process as given below:



- **Stack:** It is used for local variables and returns addresses.
- **Heap:** It is used for dynamic memory allocation.
- **Data:** It stores global and static variables.
- **Code or text:** It comprises compiled program code.

41. What is a deadlock in OS? What are the necessary conditions for a deadlock?

Deadlock is generally a situation where a set of processes are blocked as each process is holding resources and waits to acquire resources held by another process. In this situation, two or more processes simply try to execute simultaneously and wait for each to finish their execution because they are dependent on each other. We can see a hand problem in our system whenever a deadlock occurs in a program. It is one of the common problems you can see in multiprocessing.

Necessary Conditions for Deadlock

There are basically four necessary conditions for deadlock as given below:

- Mutual Exclusion
- Hold and Wait
- No Pre-emption
- Circular Wait or Resource Wait

42. What do you mean by Belady's Anomaly?

In the Operating System, process data is loaded in fixed-sized chunks and each chunk is referred to as a page. The processor loads these pages in the fixed-sized chunks of memory called frames. Belady's Anomaly is a phenomenon in which if we increase the number of frames in memory, then the number of page faults also increases. It is generally experienced when we use FIFO (First in First out) page replacement algorithm.

43. What is spooling in OS?

Spooling simply stands for Simultaneous peripheral operations online. It is referred to as putting data of various I/O jobs in a buffer. Here, buffer means a special area in memory or hard disk that can be accessible to an I/O device. It is used for mediation between a computer application and a slow peripheral. It is very useful and important because devices access or acquire data at different rates. This operation also uses disk as a very large buffer and is capable of overlapping I/O operations for one task with processor operations for another task.

Links to More Interview Questions

[C Interview Questions](#)

[Web Api Interview Questions](#)

[Cpp Interview Questions](#)

[Machine Learning Interview Questions](#)

[Css Interview Questions](#)

[Django Interview Questions](#)

[Operating System Interview Questions](#)

[Git Interview Questions](#)

[Dbms Interview Questions](#)

[Pl Sql Interview Questions](#)

[Ansible Interview Questions](#)

[Php Interview Questions](#)

[Hibernate Interview Questions](#)

[Oops Interview Questions](#)

[Docker Interview Questions](#)

[Laravel Interview Questions](#)

[Dot Net Interview Questions](#)

[React Native Interview Questions](#)

[Java 8 Interview Questions](#)

[Spring Boot Interview Questions](#)

[Tableau Interview Questions](#)

[Java Interview Questions](#)

[C Sharp Interview Questions](#)

[Node Js Interview Questions](#)

[Devops Interview Questions](#)

[Mysql Interview Questions](#)

[Asp Net Interview Questions](#)

[Kubernetes Interview Questions](#)

[Aws Interview Questions](#)

[Mongodb Interview Questions](#)

[Power Bi Interview Questions](#)

[Linux Interview Questions](#)

[Jenkins Interview Questions](#)