The Most Commonly Asked OS Interview Questions

Preparing for an interview about operating systems (OS) can be difficult. OS concepts are fundamental to understanding how software interacts with hardware. The function of operating systems is to manage hardware resources, develop a platform for executing applications and ensure efficient performance of the system. OS subjects are a common focus during technical interviews because they are fundamental to any software development or IT job.

This guide covers essential questions on process management, memory management, file systems as well as security. Additionally, example answers will be provided to help you prepare fully and ensure that you handle questions related to OS effectively during your interview.

Basic-level OS Interview Questions

1. What is an operating system?

Operating System is a computer program that manages both computer hardware devices and software resources. It acts as an intermediary between programs and users on one hand and the computer's hardware components on the other through providing typical services that computer programs use. Examples include Windows, macOS or Linux.

2. What are the main functions of an operating system?

The main functions of an operating system include:

Function	Description
Process Management	Manages the execution of processes, including multitasking and scheduling.
Memory Management	Allocates and manages the computer's memory resources.
File System Management	Controls the creation, deletion, and access of files and directories.
Device Management	Manages device communication via drivers.
Security and Access Control	Protects data and resources from unauthorised access.

3. What are the differences between multiprogramming, multitasking, and multiprocessing?

Feature	Multiprogramming	Multitasking	Multiprocessing
Definition	Multiple programs are loaded in memory, and the CPU is switched between them to increase efficiency.	Multiple tasks are performed concurrently by switching rapidly between them.	Multiple processors (CPUs) are used to perform tasks simultaneously.
Objective	Maximise CPU utilisation	Provide responsive interaction	Improve performance and reliability by parallel processing
Example	Running a batch job while a user program runs	Running a web browser and a text editor at the same time	Running multiple complex computations or server processes concurrently
Execution	Single CPU switches between programs	Single CPU switches between tasks frequently	Multiple CPUs execute different processes simultaneously
Complexity	Moderate	Moderate	High
Resource Usage	Efficient memory usage, high CPU utilisation	High CPU and memory usage	High CPU usage with efficient distribution of tasks

4. What is a context switch in the Operating System?

Context switch refers to a process whereby CPU state storage is done in such a way that multiple processes are enabled to share a single CPU resource. When context switching takes place, it enables the OS to jump from one task to another, ensuring efficiency in multitasking or execution of other critical functions within a PC system. During the context-switching process, information like registers or program counters would be saved into the process table before being loaded into the new state of the next running process.

5. What is a process and process table?

A process is a program in execution, including its code, data, and resources. It is an active entity with a unique Process ID (PID). The process table is a data structure maintained by the operating system that keeps track of all the processes. It contains information such as the process state, PID, program counter, and memory allocation.

Field	Description
Process ID (PID)	Unique identifier for the process
Process State	Current state (running, waiting)
Program Counter	Address of the next instruction
Memory Allocation	Memory used by the process

6. What is Thread?

A thread is the smallest unit of a process that can be scheduled and executed by the CPU. It is a path of execution within a process and shares the process's resources, such as memory and open files. Multiple threads can exist within a single process, enabling parallelism and efficient task management.

7. Explain the difference between a process and a thread.

Feature	Process	Thread
Definition	An independent program in execution.	A lightweight sub-process or task within a process.
Memory	Each process has its own memory space.	Threads share the same memory space within a process.
Resource Overhead	High, as each process has its own resources.	Low, as threads share resources of the process.
Communication	Inter-process communication (IPC) needed.	Direct communication through shared memory.
Creation Time	Longer, due to resource allocation.	Shorter, as resources are already allocated.

8. Describe the process of process creation and termination.

Process Creation:

- **Forking:** The operating system creates a new process by duplicating an existing one.
- **Initialization**: The new process is initialised with a unique Process ID (PID).
- **Resource Allocation:** Memory and other resources are allocated.
- **Execution:** The new process starts executing, either continuing from the point of the original process or beginning a new program.

Process Termination:

- **Normal Exit:** The process completes its execution and calls an exit system call.
- **Error Exit:** The process encounters an error and is terminated.
- **Forced Termination:** Another process or the OS forces the process to terminate using a kill command.
- **Resource Deallocation:** The OS reclaims the resources used by the process.

9. What is the difference between preemptive and non-preemptive scheduling?

Feature	Preemptive Scheduling	Non-Preemptive Scheduling
Definition	The OS can interrupt and switch tasks.	The OS waits for the running task to finish.
Control	The OS has control over task switching.	The running task has control over its completion.
Response Time	Lower, due to quick task switching.	Higher, as each task runs to completion.
Complexity	Higher, due to handling interruptions.	Lower, as tasks run without interruptions.
Example	Round Robin, Shortest Remaining Time First (SRTF)	First-Come, First-Served (FCFS), Shortest Job Next (SJN)

10. What are system calls, and how are they different from normal function calls?

System Calls:

- System calls are special functions provided by the OS that allow programs to request services from the kernel, such as file operations, process control, and communication.

- They run in kernel mode, providing access to hardware and critical system resources.

Normal Function Calls:

- Normal function calls are user-defined, or library functions that execute within the user program's context.
- They run in user mode and do not have direct access to hardware or kernel resources.

Feature	**System Calls**	**Normal Function Calls**
Access Level	Kernel mode	User mode
Purpose	Request services from the OS	Perform specific tasks within a program
Execution Overhead	Higher, due to context switching to kernel	Lower, executed within the same user space
Examples	open(), read(), write()	printf(), strlen(), sort()

11. Differentiate the kernel mode and user mode.

Feature	**Kernel Mode**	**User Mode**
Access Level	Full access to all system resources	Limited access to system resources
Privileges	High, can execute privileged instructions	Low, cannot execute privileged instructions
Purpose	Execute OS code and manage hardware	Execute application code
Stability	Less stable, errors can crash the system	More stable, errors are confined to the application
Example	System calls, device drivers	User applications, standard library calls

12. What is Thrashing?

Thrashing occurs when a system spends more time swapping pages in and out of memory than executing actual processes. This usually happens when there is insufficient RAM, causing excessive paging and severely degrading system performance.

13. What are the benefits of multithreaded programming?

Benefit	**Description**
Responsiveness	Improves application responsiveness by allowing background tasks to run concurrently.
Resource Sharing	Threads within a process share resources, which is more efficient than separate processes.
Scalability	Utilises multiprocessor architectures more effectively by running threads in parallel.
Simplified Program Structure	Easier to manage multiple tasks within a single application context.

14. What is Buffer?

Any temporary storage section that is used as an intermediate in withholding data during its movement from one location to another is known as a buffer. Buffers are used to manage data streams and improve data transfer efficiency between devices or processes.

15. What is Demand paging?

Demand paging refers to a memory management method in which only pages of data are loaded into memory only when they are needed, rather than preloading all pages. Hence, it helps reduce memory usage and improve system performance.

16. What are the different states of the process?

State	Description
New	The process is being created.
Ready	The process is waiting to be assigned to a CPU.
Running	The process is currently being executed by the CPU.
Waiting	The process is waiting for an event (e.g., I/O completion).
Terminated	The process has finished execution or has been terminated by the OS.
Suspended	The process is temporarily stopped, possibly swapped out to disk.

17. What are the 5 important concepts of OS?

- **Process Management:** Handling the creation, scheduling, and termination of processes.
- **Memory Management:** Managing the allocation and deallocation of memory space.
- **File System Management:** Organizing, storing, and accessing data on storage devices.
- **Security and Access Control:** Protecting data and resources from unauthorised access.
- **Device Management:** Controlling and coordinating hardware devices through drivers.

18. What are the different scheduling algorithms?

Scheduling Algorithm	Description
First-Come, First-Served (FCFS)	Processes are executed in the order they arrive.
Shortest Job Next (SJN)	Processes with the shortest execution time are scheduled next.
Priority Scheduling	Processes are scheduled based on priority levels.
Round Robin (RR)	Each process gets a fixed time slice in a cyclic order.
Multilevel Queue Scheduling	Processes are divided into multiple queues based on their priority and type.
Shortest Remaining Time First (SRTF)	A preemptive version of SJN, where the process with the least remaining time is scheduled.

19. Describe the objective of multiprogramming.

The objective of multiprogramming is to maximise CPU utilisation by running multiple processes simultaneously. This ensures that the CPU is always busy executing a process, improving system efficiency and throughput.

20. What problem do we face in computer systems without OS?

Without an OS, computer systems face several problems, including:

- **Resource Management:** No centralised management of CPU, memory, and I/O devices.
- **User Interface**: Lack of a user-friendly interface to interact with hardware.
- **Security:** No protection mechanisms to prevent unauthorised access.
- **Program Execution:** Difficulties in loading, executing, and managing multiple programs.

21. What is FCFS?

First-Come, First-Served (FCFS) scheduling algorithm executes processes according to their arrival order in the ready queue. It has an easy validation process because it only needs proportionate waiting times, but in some scenarios, it results in a convoy effect, causing slow response due to long tasks with small ones behind them.

22. What is the RR scheduling algorithm?

Round Robin (RR) Scheduling Algorithm allocates a fixed time slot, (Time Quantum) for every process in the ready queue. The CPU cycles through processes giving each one a turn to execute for quantum duration. This ensures fairness, leading to better response time but may result in massive context switching overheads.

23. What is cache?

Caching is the process of placing often-used data in a temporary storage area to speed up the retrieval of data. This latency-reducing and performance-enhancing function can be found at different levels of computing.

24. What is the functionality of an Assembler?

An assembler converts assembly language code into machine code that is executable by a computer’s central processing unit (CPU). This makes it possible for humans to make use of mnemonics and symbols, which are then converted into the corresponding binary opcode and addresses, making low-level programming possible.

25. What does GUI mean?

A graphical user interface (GUI) allows people to interact with computers through windows, icons, buttons, menus as opposed to typing command lines. GUIs facilitate software accessibility and usability by using intuitive graphical controls that allow users to perform tasks easily.

26. What is a pipe? When Is It Used?

Pipes use unidirectional flow when passing information from one process to another in inter-process communication. Pipes are commonly used for connecting the output from one process to the input of another thus enabling sharing of data and task chaining in an efficient manner.

27. What are the goals of CPU scheduling?

Goal	Description
Fairness	Ensure all processes get a fair share of CPU time.
Efficiency	Maximise CPU utilisation and minimise idle time.
Response Time	Minimise the time from submission to the first response.
Turnaround Time	Minimise the total time from submission to completion of a process.
Throughput	Maximise the number of processes completed per unit time.
Predictability	Ensure consistent and predictable process performance.

28. Write the name of synchronisation techniques.

- Mutexes (Mutual Exclusion)
- Semaphores
- Monitors
- Spinlocks
- Barriers
- Condition Variables

29. Define the term Bit-Vector.

A pipe is a method of inter-process communication (IPC) that allows data to be passed from one process to another in a unidirectional flow. Pipes are commonly used to connect the output of one process to the input of another, enabling data sharing and task chaining in a streamlined manner.

30. Write the names of different operations on the file.

- Create
- Open
- Read
- Write
- Close
- Delete
- Append
- Rename

31. What is rotational latency?

Rotational latency is the delay experienced while waiting for the desired disk sector to rotate under the read/write head. It is a component of the total disk access time, along with seek time and data transfer time.

32. What is a File allocation table?

A File Allocation Table (FAT) is a file system structure used to keep track of the clusters (storage units) on a disk. It maps each file to its corresponding clusters, allowing the system to locate and manage files efficiently. FAT is used in various file systems, including FAT12, FAT16, and FAT32.

33. How to recover from a deadlock?

Deadlock Recovery Method	Description
Process Termination	Terminate one or more processes involved in the deadlock.
Resource Preemption	Temporarily take resources away from processes and relocate them.
Rollback	Roll back one or more processes to a safe state before the deadlock occurred.
Deadlock Detection and Resolution	Continuously check for deadlocks and resolve them using one of the above methods.

34. Definition of real-time systems

Real-time systems refer to computer systems that must respond quickly and precisely at times dictated by external events. Such systems are meant for processing data within hard constraints with regard to timing, they are popularly used in areas like embedded systems, industrial control, mission-critical tasks etc.

35. What is seek time?

Seek time is the duration that a hard disk drive (HDD) takes to find and move a read or write head from one track to another where necessary data has been stored. It contributes significantly to the performance of hard disk access as its value forms a part of the total disk access time.

Intermediate-level OS Interview Questions

36. What are the different RAID levels?

RAID Level	Description
RAID 0	Data striping without redundancy, improves performance but no fault tolerance.
RAID 1	Data mirroring, duplicates data on two disks for fault tolerance.
RAID 5	Data striping with distributed parity, provides fault tolerance and efficient storage.
RAID 6	Similar to RAID 5 but with an extra parity block, allows for up to two disk failures.
RAID 10	Combines RAID 1 and RAID 0, mirroring and striping for high performance and fault tolerance.
RAID 50	Combines RAID 5 and RAID 0, providing fault tolerance and high performance.
RAID 60	Combines RAID 6 and RAID 0, offering enhanced fault tolerance and performance.

37. Explain Banker’s algorithm.

Banker’s algorithm is a deadlock-avoidance algorithm. It allocates resources based on resource needs of each process, then determines if the state is safe in terms of deadlock avoidance. If the system remains in safe state after allocation, the resources are allocated; otherwise make it wait.

38. What are the Benefits of Dynamic Loading Concerning Better Memory Space Utilisation?

Dynamic loading loads a program’s parts into memory only when they are needed, rather than loading the entire program at once. This reduces the amount of RAM consumed by any specific program at any given moment and allows more simultaneous loading of programs.

39. Explain the main difference between logical and physical address space.

Feature	Logical Address Space	Physical Address Space
Definition	The address generated by the CPU during program execution.	The actual address in the memory unit.
View	User view of the memory (virtual address).	Hardware view of the memory.
Access	Managed by the OS using memory management techniques.	Directly accessed by the memory hardware.

40. What are overlays?

Overlays are mechanisms used to work around physical memory limitations within systems by loading only relevant instructions and data segments into memory. Whenever some different portion of this program may be needed, the current overlay is replaced with the required one. By using this technique, large-sized applications can run on low-memory systems without compromising their functionality at all.

41. What is fragmentation?

Fragmentation occurs when memory is allocated and deallocated in a way that leaves small, unusable gaps. It comes in two types:

Type	Description
Internal Fragmentation	Unused memory within allocated regions, due to the allocated size being larger than needed.
External Fragmentation	Unused memory between allocated regions, due to varying sizes of memory allocation and deallocation.

42. What is the basic function of paging?

It is a memory management technique that divides a process’s memory into pages that have fixed sizes and physical memory into frames with similar sizes. The functionality of paging provides a way to connect virtual addresses to physical addresses, thereby making it possible for computer systems to use memory in an efficient manner without the need for contiguous allocation.

43. How does swapping result in better memory management?

Swapping improves memory management by temporarily moving inactive processes from the main memory to a storage device (swap space). This frees up RAM for active processes, allowing the system to handle more processes concurrently and improve overall performance. When the swapped-out process is needed again, it is swapped back into memory.

44. What is the Direct Access Method?

The Direct Access Method allows data to be read or written directly to a specific location on a storage device without sequentially accessing preceding data. This method is commonly used in disk drives where each block or sector has a unique address, enabling quick access to data and improving performance for large data sets.

45. When does thrashing occur?

Thrashing occurs when the computer system spends too much time paging (swapping data in and out of memory) than executing any actual processes. This happens due to insufficient RAM, which can cause excessive paging and significantly degrade the system's performance.

46. What are interrupts?

Interrupts mean signals that are sent to CPU by software or hardware, indicating an event which needs immediate attention. When an interrupt takes place, the CPU temporarily stops executing current instructions and transfers control to an interrupt handler to address the event. After handling the interrupt, the CPU resumes its normal execution.

47. What is Preemptive Multitasking?

Preemptive Multitasking is one scheduling method for a central processing unit. In this method, the operating system allocates CPU time to processes and may forcibly remove a process from the CPU when it is allocating another process's time. This ensures that all processes get an equal share of CPU time and improves this system's response by not allowing one process to hold the CPU fully.

48. What are the advantages of semaphores?

Advantage	Description
Synchronisation	Semaphores help synchronise access to shared resources, preventing race conditions.
Mutual Exclusion	Ensures that only one process accesses a critical section at a time.
Simplicity	Provides a simple mechanism for managing concurrent processes.
Flexibility	Can be used for both signalling and resource counting, making them versatile.
Efficiency	Reduces the need for busy waiting, as processes can be blocked until the semaphore is available.

49. IPC is what?

Inter-process Communication (IPC) is a mechanism through which several processes interact with each other along with coordinating their activities. These include message passing, shared memory, semaphores, pipes etc., thus enabling sharing of data as well as synchronisation among various processes.

50. What is a Batch Operating System?

A Batch Operating System processes batches of jobs with similar requirements together, without user interaction during execution. Jobs are collected, grouped, and processed sequentially, improving resource utilisation and throughput.

51. What are starvation and ageing in the OS?

Concept	Description
Starvation	Occurs when a process waits indefinitely for resources due to continuous resource allocation to other processes.
Aging	A technique used to prevent starvation by gradually increasing the priority of waiting processes over time.

52. What is PCB?

A Process Control Block (PCB) is a data structure used by the operating system to store information about a process. The PCB contains details such as:

Field	Description
Process ID (PID)	Unique identifier for the process
Process State	Current state of the process (e.g., running, waiting)
Program Counter	Address of the next instruction to execute
CPU Registers	Current values of the CPU registers
Memory Management	Information about memory allocation

53. When is a system in a safe state?

A system is in a safe state if there exists a sequence of processes such that each process can be allocated the necessary resources and complete its execution without causing a deadlock. In a safe state, the system can avoid deadlock by careful resource allocation.

54. What is Cycle Stealing?

Cycle stealing entails Direct Memory Access (DMA) controllers transferring data between memory and I/O devices directly by ‘stealing’ CPU cycles. During cycle stealing, the DMA controller gains access to memory while momentarily halting the operation of the Central Processing Unit, thereby reducing its workloads in many cases related to data transfer operations.

55. What are Trap and Trapdoor?

Concept	Description
Trap	A trap is a software-generated interrupt caused by an error or specific condition in a program, triggering a switch to the OS to handle the event.
Trapdoor	A trapdoor (or backdoor) is a hidden method of bypassing normal authentication or security controls, often used maliciously to gain unauthorised access to a system.

56. What does the dispatcher imply?

Dispatcher is a component of the CPU scheduler that transfers control over CPU selection made by the scheduler itself. The dispatcher performs context switching, i.e., changing the CPU from one process to another selected process and ensuring the selected process starts execution.

57. What is the Locality of reference?

Locality of reference refers to the tendency of a program to access a relatively small set of memory locations repeatedly over a short period. It is divided into two types:

Type	Description
Temporal Locality	Recently accessed memory locations are likely to be accessed again soon.
Spatial Locality	Memory locations near recently accessed locations are likely to be accessed soon.

58. How do we calculate performance in virtual memory?

Performance in virtual memory is calculated by evaluating the following factors:

Factor	Description
Page Fault Rate	The frequency of page faults (lower is better)
Effective Access Time (EAT)	Calculated as: EAT = (1 – Page Fault Rate) * Memory Access Time + Page Fault Rate * Page Fault Handling Time
TLB Hit Rate	The frequency of successful lookups in the Translation Lookaside Buffer (TLB) (higher is better)

59. What is reentrancy?

Reentrancy refers to the ability of a program or function to be safely interrupted and called again (“re-entered”) before its previous executions are complete. Reentrant code does not rely on shared data and uses local variables, allowing multiple instances to execute concurrently without causing conflicts.

60. Write the top 10 examples of OS.

- Operating System
- Windows 10
- Windows 11
- macOS
- Linux (Ubuntu)
- Linux (Fedora)
- Linux (Debian)
- Android
- iOS
- FreeBSD
- Chrome OS

Advanced-level OS Interview Questions

61. What are the different types of Kernel?

Kernel Type	Description
Monolithic Kernel	A single large kernel that handles all OS services in one address space.
Microkernel	A minimal kernel that runs basic services like communication and I/O in kernel space, with other services in user space.
Hybrid Kernel	Combines features of monolithic and microkernels to improve performance and modularity.
Exokernel	Provides minimal abstractions and allows applications to manage hardware resources directly.
Nanokernel	An extremely lightweight kernel that performs very few basic tasks, mainly used in embedded systems.

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

Semaphore is a synchronisation primitive used to regulate access of multiple processes to a common resource in a concurrent system. It is an integer variable that can be incremented or decremented due to a request from the process. Semaphores are majorly used to combat critical section problems and race conditions, ensuring secure and orderly access to shared resources.

63. What are the main functions of Kernel?

Function	Description
Process Management	Manages process creation, scheduling, and termination.
Memory Management	Controls memory allocation and deallocation.
Device Management	Manages communication between hardware devices and the system.
File System Management	Handles file creation, deletion, and access control.

Security and Access Control	Enforces security policies and controls access to system resources.
System Calls Management	Provides an interface for user applications to interact with the hardware.

64. Write the difference between a microkernel and a monolithic kernel.

Feature	Microkernel	Monolithic Kernel
Structure	Minimal core functions in kernel space, other services in user space.	All OS services run in kernel space.
Size	Smaller	Larger
Performance	It can be slower due to user-space communication overhead.	Generally faster due to direct service handling in kernel space.
Reliability	More reliable and secure due to isolated services.	Less reliable as a bug in one service can crash the entire system.
Extensibility	Easier to extend and modify without affecting the entire system.	Harder to modify and extend due to tightly coupled services.

65. What does SMP mean (Symmetric Multiprocessing)?

Symmetric multiprocessing (SMP) can be defined as an architecture in which one common memory exists with multiple processors operating under one operating system instance. For each processor, it has equal access towards memory as well as I/O devices such that multitasking applications have parallel processing, resulting in better performance.

66. What is the time-sharing system?

Time-sharing allows several users or tasks to share the same system resources at once by switching between them quickly. Each user or task is given a short amount of time for their execution which gives the impression of having many processes running at the same time thus increasing the general performance and responsiveness of the systems.

67. What are the benefits and disadvantages of a Batch Operating System?

Aspect	Benefits	Disadvantages
Efficiency	Maximises resource utilisation by executing batches of jobs.	Lack of interaction and flexibility for real-time tasks.
Automation	Reduces manual intervention and automates repetitive tasks.	Difficult to debug and handle errors during job execution.
Throughput	High throughput due to streamlined job processing.	Long turnaround time for individual jobs.
Cost	Lower operational costs due to efficient resource usage.	Inflexibility in handling diverse job requirements.

68. What is a bootstrap program in the OS?

A bootstrap program is the initial code executed when a computer is powered on or restarted. It initialises the hardware, sets up the environment for the operating system to run, and loads the OS kernel into memory. The bootstrap program is typically stored in the computer's firmware (e.g., BIOS or UEFI).

69. What are the different IPC mechanisms?

IPC Mechanism	Description
Pipes	Allows communication between processes through a unidirectional data stream.
Message Queues	Enables processes to exchange messages in a queue.
Shared Memory	Allows multiple processes to access a common memory space.
Semaphores	Synchronises access to shared resources using signalling mechanisms.
Sockets	Facilitates communication between processes over a network.
Signals	Uses asynchronous notifications to communicate events to processes.

70. Which thing means deadlock in OS?

Deadlock takes place when there exist some processes that cannot go forward because every process waits for another process among this set holding a resource they need. Consequently, no single process can complete its execution, leading to the halting of the entire system. Deadlocks mainly involve four conditions: mutual exclusion, hold and wait, no preemption, and circular wait.

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

Belady's Anomaly is a phenomenon in some page replacement algorithms where increasing the number of page frames results in an increase in the number of page faults. It occurs in algorithms like FIFO (First-In, First-Out), where more memory can paradoxically lead to worse performance due to suboptimal page replacement decisions.

72. What is spooling in the operating system?

Spooling (simultaneous peripheral operations online) is a technique where data is held temporarily in a buffer so that it can be used and executed by another device or program later on. This method is often employed for print management purposes whereby documents are spooled to disk or memory and lined up for printing, freeing up CPU cycles so that it can perform other tasks.

73. Where is the Batch Operating System used in Real Life?

Batch Operating Systems are used in environments where large volumes of similar jobs are processed without user interaction. Real-life applications include:

- **Payroll Systems:** Processing employee salaries and benefits.
- **Banking Systems:** Handling end-of-day transactions and updates.
- **Scientific Computation:** Running extensive calculations and simulations.
- **Data Processing:** Large-scale data analysis and reporting.

74. What are Monitors in the Context of Operating Systems?

Monitors are high-level synchronisation constructs used to control access to shared resources in concurrent programming. They combine mutual exclusion (mutex) and condition variables to manage the safe execution of code blocks by allowing only one process or thread to execute within the monitor at a time.

75. What is a zombie process?

A zombie process is a process that has completed execution but still has an entry in the process table. This occurs when the process's parent has not read its exit status using the wait() system call. Zombie processes occupy a slot in the process table but do not consume system resources.

Conclusion

Understanding operating system concepts is crucial for performing well in technical interviews and excelling in software development and IT roles. This guide has covered key topics such as process management, memory management, file systems, and security, providing you with a solid foundation to tackle OS-related questions. By mastering these areas, you can enhance your problem-solving skills and demonstrate your technical expertise to potential employers.

As you prepare for your interview, remember to practise answering questions clearly and concisely, and consider how these concepts apply to real-world scenarios. With thorough preparation and a deep understanding of operating systems, you will be well-equipped to handle challenging interview questions and advance your career in the tech industry.