

## Interview Preparation Pack (25 questions and answers): Operating Systems (for Lecture 6)

#	Question	Suggested Answer	What Interviewers Look For
1	Describe the different types of file system permissions and their significance	<p>File system permissions determine who can read, write, or execute a file or directory. The most common types are:</p> <ul style="list-style-type: none"> <li>• <b>Read (r):</b> Allows viewing file contents or listing directory contents.</li> <li>• <b>Write (w):</b> Allows modifying file contents or managing files within a directory.</li> <li>• <b>Execute (x):</b> Allows running a file as a program or accessing files within a directory.</li> </ul> <p>Permissions are assigned to three user categories:</p> <ul style="list-style-type: none"> <li>• <b>Owner:</b> The user who owns the file or directory, typically with the most control.</li> <li>• <b>Group:</b> A set of users sharing the same permissions, useful for collaboration.</li> <li>• <b>Others:</b> All other users not in the owner or group categories, usually with the most restrictive permissions.</li> </ul>	Conceptual clarity and understanding of OS fundamentals.
2	Discuss the pros and cons of using a distributed file system like HDFS.	Hadoop Distributed File System (HDFS) offers numerous advantages, including storing massive amounts of data, high fault tolerance, and streaming data access. However, it also has limitations, such as challenges with small files, inability to achieve low latency, and limited support for file modification and parallel writing.	Conceptual clarity, real-world relevance, and understanding of OS fundamentals.
3	How does the NTFS file system differ from FAT32?	<p>NTFS (New Technology File System) and FAT32 (File Allocation Table 32) are file systems used by Windows operating systems. Key differences include:</p> <ul style="list-style-type: none"> <li>• <b>File Size Limits:</b> NTFS supports very large files, up to 16 TB, whereas FAT32 has a 4 GB limit.</li> <li>• <b>Volume Size Limits:</b> NTFS can handle volumes up to 256 TB, while FAT32 is limited to 32 GB in Windows.</li> <li>• <b>Security:</b> NTFS provides file-level security with permissions and encryption, which FAT32 lacks.</li> </ul>	Conceptual clarity, real-world relevance, and understanding of OS fundamentals.

		<ul style="list-style-type: none"> <li>• <b>Reliability:</b> NTFS includes features like transaction logging and recovery, enhancing data integrity.</li> <li>• <b>Performance:</b> NTFS is generally faster and more efficient with large volumes and files.</li> <li>• <b>Compatibility:</b> FAT32 is more compatible with a wider range of operating systems and devices.</li> </ul>	
4	What is a B-tree and how is it used in file systems?	A B-tree is a self-balancing tree data structure that maintains sorted data and allows efficient insertion, deletion, and search operations. It is used in file systems and databases to store and access large amounts of data quickly. B-trees minimize disk I/O operations, enhancing performance. In file systems, B-trees index data like file names and metadata, ensuring the tree remains balanced for quick operations. B-trees are used in directory structures and file allocation tables, such as in the HFS+ file system in macOS	Conceptual clarity and understanding of OS fundamentals.
5	What is the difference between hardware RAID and Software RAID?	<p>The hardware-based RAID is independent from the host. A Hardware RAID device connects to the SCSI controller and presents the RAID arrays as a single SCSI drive. An external RAID system moves all RAID handling “intelligence” into a controller located in the external disk subsystem. The whole subsystem is connected to the host via a normal SCSI controller and appears to the host as a single disk.</p> <p>Software RAID is implemented under OS Kernel level. The Linux kernel contains an MD driver that allows the RAID solution to be completely hardware independent. The performance of a software-based array depends on the server CPU performance and load.</p>	Conceptual clarity, real-world relevance, and understanding of OS fundamentals.
6	What are the three main file allocation methods? Explain	<ul style="list-style-type: none"> <li>• <b>Contiguous Allocation:</b> Each file occupies a set of contiguous blocks on the disk.</li> <li>• <b>Linked Allocation:</b> Each file is a linked list of disk blocks, where each block contains a pointer to the next block.</li> <li>• <b>Indexed Allocation:</b> All pointers to scattered blocks are placed together in one location (the index block).</li> </ul>	Conceptual clarity, real-world relevance, and understanding of OS fundamentals.

7	What is a File Allocation Table (FAT)?	The FAT is a data structure used in many file systems to track the location of files and their associated blocks on the disk.	Conceptual clarity and understanding of OS fundamentals.
8	What is file buffering, and what are its advantages?	File buffering involves storing data temporarily in memory before writing it to disk. This can improve performance by reducing the number of disk accesses.	Conceptual clarity, real-world relevance, and understanding of OS fundamentals.
9	What is the difference between internal and external fragmentation?	<b>Internal Fragmentation:</b> Occurs when a file is allocated more space than it needs. <b>External Fragmentation:</b> Occurs when there are free spaces on the disk that are too small to accommodate a new file.	Conceptual clarity, real-world relevance, and understanding of OS fundamentals.
10	Explain the advantages and disadvantages of indexed allocation.	Advantages: Solves the problems of contiguous allocation (fragmentation, space limitations) and linked allocation (slow sequential access) by allowing direct access to any block without needing to traverse a linked list. Disadvantages: Requires an index block, which takes up additional space on the disk	Conceptual clarity and understanding of OS fundamentals.
11	Explain the advantages and disadvantages of contiguous allocation.	Supports both sequential and direct access. Direct access is fast because the address of the kth block can be calculated easily.	Conceptual clarity and understanding of OS fundamentals.
12	What are disk scheduling algorithms, and why are they necessary?	Disk scheduling algorithms are methods used to order I/O requests to a disk, aiming to reduce the average seek time (time it takes for the disk head to move to the desired track) and rotational latency (time it takes for the desired sector to rotate under the head). They're necessary because multiple processes can make I/O requests simultaneously, and these requests might be located at different tracks, potentially increasing seek time	Conceptual clarity, real-world relevance, and understanding of OS fundamentals.
13	Explain the First-Come-First-Served (FCFS) disk scheduling algorithm.	FCFS services requests in the order they arrive. While simple, it can lead to high seek times if requests are not clustered, especially if there are large variations in seek distances between requests.	Conceptual clarity, real-world relevance, and understanding of OS fundamentals.
14	What is the SCAN algorithm?	SCAN, also known as the Elevator algorithm, works by moving the head in one direction (e.g., from left to right) and servicing requests	Conceptual clarity, real-world relevance, and understanding of OS fundamentals.

		along the way until it reaches the end of the disk. Then, it reverses direction and services the remaining requests.	
15	Compare and contrast SCAN and C-SCAN/LOOK.	SCAN (and LOOK) move to the end of the disk even if there are no requests in that direction, potentially wasting seek time. C-SCAN and C-LOOK avoid this by jumping to the other end of the disk, providing more uniform servicing	Conceptual clarity, real-world relevance, and understanding of OS fundamentals.
16	Which algorithm is best for a system with a high number of I/O requests?	SSTF and LOOK are generally more efficient in handling a high number of requests, as they avoid unnecessary travel to the end of the disk if there are no requests in that direction.	Conceptual clarity, real-world relevance, and understanding of OS fundamentals.
17	How do disk scheduling algorithms affect disk access time?	By reordering requests, disk scheduling algorithms aim to minimize seek time, which is a major component of disk access time. They also contribute to faster response times and fairer service for all requests.	Conceptual clarity and understanding of OS fundamentals.
18	Describe Shortest Seek Time First (SSTF).	SSTF prioritizes the request with the shortest seek distance from the current head position. It can minimize seek time but might cause starvation if there are many requests in distant locations.	Conceptual clarity and understanding of OS fundamentals.
19	What are some common authentication methods used in OS?	Examples include username/password, multi-factor authentication (MFA) like two-factor authentication, and biometric authentication.	Conceptual clarity, real-world relevance, and understanding of OS fundamentals.
20	Explain the different types of user accounts and their privileges.	Superuser accounts (like root in Linux) have full access, while regular users have limited privileges.	Conceptual clarity, real-world relevance, and understanding of OS fundamentals.
21	How can OS authentication be made more secure?	Implementing stronger passwords, enabling MFA, and regularly updating the OS are important steps.	Conceptual clarity, real-world relevance, and understanding of OS fundamentals.
22	What is access control? What are the different types of access control?	Access control is the process of controlling who or what can access specific resources within the system, typically based on user identity or role. Common types include discretionary access control (DAC), mandatory access control (MAC), and role-based access control (RBAC).	Conceptual clarity, real-world relevance, and understanding of OS fundamentals.
23	What is a firewall? What is an intrusion detection system (IDS)?	A firewall is a network security device or software that monitors and controls network traffic based on a set of rules, preventing unauthorized access. An intrusion detection system (IDS) monitors	Conceptual clarity and understanding of OS fundamentals.

---

		network or system activity for malicious or suspicious behaviour, alerting administrators to potential threats.	
24	What is a buffer overflow? Explain	A buffer overflow occurs when a program writes more data into a buffer than it can hold, potentially overwriting other parts of memory and leading to security vulnerabilities	Conceptual clarity and understanding of OS fundamentals.
25	What are the differences between Hashing and Encryption?	Hashing is a one-way function that converts data into a fixed-length hash value, while encryption is the process of converting data into an unreadable format that can only be decrypted with a key	Conceptual clarity, real-world relevance, and understanding of OS fundamentals.