Click to verify

Preparing for technical interviews can be a daunting task, especially when it comes to core subjects like Operating Systems (OS), Database Management Systems (DBMS), and Computer Networks (CN). These topics are essential for any computer science or IT professional and often form the foundation of many interview questions. To help you prepare effectively, this article provides a comprehensive collection of over 50+ key interview questions across OS, DBMS, and CN. These questions will help you refresh your knowledge, understand core concepts, and practice your problem-solving skills. From basic definitions to more complex scenarios, this guide covers the most important and frequently asked questions to ensure you're well-prepared for your next interview.Operating System Interview Questions1. What is an Operating System and why is it used?An Operating System (OS) is system software that manages computer hardware and software resources and provides common services for computer programs. It acts as an intermediary between users and the computer hardware. The purpose of an operating system is to provide an environment in which a user can execute programs conveniently and efficiently.Why it is used:Resource Management: Manages CPU, memory, disk space, and input/output devices.User Interface: Provides a user interface (UI), such as a command line or graphical user interface (GUI).Task Scheduling: Handles multitasking, allowing multiple programs to run simultaneously.Security: Ensures data protection and system security.File Management: Organizes and manages files and directories.2. What is Thrashing?Thrashing occurs when excessive paging occurs, causing the system to slow down significantly. It happens when the system spends more time swapping data between RAM and disk (virtual memory) than executing actual processes, often due to insufficient memory. This leads to a severe drop in performance as the system constantly tries to manage memory rather than perform useful tasks.3. What are the differences between process and thread?A process is an independent program in execution with its memory and resources, and a thread is the smallest segment of a process that shares memory and resources with other threads in the same process. Processes are heavier and have higher overhead, requiring inter-process communication (IPC) to interact. Threads are lightweight, faster to create, and can communicate easily within the same process since they share the same memory space.4. What is paging and why do we need it? What is demand paging?Paging is a memory management scheme that divides physical memory into fixed-sized blocks called pages and logical memory into blocks of the same size called page frames. It helps in efficiently utilizing memory by allowing non-contiguous memory allocation, thus avoiding fragmentation. We need paging to make better use of memory, allowing processes to use memory that may not be contiguous, and to provide each process with its own address space, ensuring isolation and security.Demand paging is a technique where a page is only loaded into memory when needed, rather than loading the entire process at once. This helps save memory and improves performance by loading only the pages that are actively being used, reducing the time and resources required to load a program.5. What are the different Scheduling Algorithms?Scheduling algorithms are methods used by the operating system to determine the order in which processes are executed by the CPU. These algorithms help manage process execution, ensuring efficient CPU utilization and process management.Types of Scheduling Algorithms:First-Come, First-Served (FCFS): The process that arrives first is executed first, without preemption.Shortest Job Next (SJN): The process with the smallest execution time is selected next.Round Robin (RR): Each process is assigned a fixed time slice, and execution is rotated among processes.Priority Scheduling: Processes are assigned a priority, and the one with the highest priority is executed first.Shortest Remaining Time First (SRTF): A preemptive version of SJN, where the process with the shortest remaining time is executed next.Multilevel Queue Scheduling: Processes are divided into queues based on priority or other factors, and each queue has its own scheduling algorithm.To know more about the topic refer to CPU Scheduling in Operating Systems6. What is the difference between logical and physical memory?Logical memory refers to the address space that a program uses, which is independent of the physical hardware. Its the memory the program sees and interacts with, typically managed by the operating system using virtual memory. Physical memory, on the other hand, refers to the actual RAM (hardware memory) installed on the system. Its where data is physically stored and accessed by the computers CPU.In summary, logical memory is the abstracted address space used by programs, while physical memory is the real hardware memory where data is stored.7. What is Belady's Anomaly?Belady's Anomaly is a situation where increasing the number of page frames in memory leads to more page faults, which goes against the usual expectation that more memory reduces page faults. This anomaly occurs in certain page replacement algorithms, like FIFO.To know more about the topic refer to Belady's Anomaly8. What is Banker's Algorithm?Banker's Algorithm is a deadlock avoidance algorithm used in operating systems to allocate resources to processes in a safe manner. It determines whether a process can be granted a resource request by checking if the system will remain in a "safe state" after the request is granted. A safe state is one where there is at least one sequence of processes that can complete without causing a deadlock.To know more about the topic refer to Bankers Algorithm in Operating System9. What is Fragmentation? What are the different types of Fragmentation?Fragmentation refers to the condition where memory is inefficiently utilized due to the allocation and deallocation of memory in varying sizes. Over time, this can lead to gaps or small unusable blocks of free memory, making it difficult to allocate large contiguous blocks of memory to processes, even though the total free memory may appear sufficient.Types of Fragmentation:External Fragmentation: Occurs when free memory is scattered in small, non-contiguous blocks, preventing large enough blocks for new processes despite having enough total free memory.Internal Fragmentation: Happens when allocated memory exceeds the memory requested by a process, leaving unused space within the allocated block.10. What is Spooling?Spooling (Simultaneous Peripheral Operations On-Line) is a process in which data is temporarily stored in a buffer or queue before being sent to an output device, like a printer or disk, for processing. This allows the CPU to continue executing other tasks while the peripheral device processes the data in the background. Essentially, spooling helps manage data efficiently by handling the queuing and scheduling of output operations, ensuring that multiple processes can be handled without conflict.11. What are the Classic Synchronization Problems?Classic synchronization problems are common issues in concurrent programming where multiple processes or threads need to access shared resources without causing data inconsistencies or conflicts. These problems highlight the challenges of synchronizing access to shared data and ensuring correct behavior in multi-threaded or multi-process systems.Types of Classic Synchronization Problems:The Producer-Consumer Problem: A producer generates data and a consumer consumes it. The challenge is to ensure the producer doesn't overwrite data before the consumer processes it, and vice versa.The Reader-Writer Problem: Multiple readers can access the resource simultaneously, but writers need exclusive access. The challenge is to balance access between readers and writers without causing data inconsistencies.The Dining Philosophers Problem: A scenario where multiple philosophers need two forks to eat. The challenge is to avoid deadlock and ensure no philosopher starves due to resource contention.The Sleeping Barber Problem: A barber serves customers in a barber shop. The challenge is to synchronize the barber and customers in such a way that no customer is left waiting indefinitely and the barber doesnt sit idle if there are customers.To know more about the topic refer to Classical Problems of Synchronization with Semaphore Solution12. What is the difference between Multitasking and Multiprocessing?Multitasking and multiprocessing are both techniques used to execute multiple tasks simultaneously, but they differ in how they utilize system resources.Multitasking refers to the ability of an operating system to manage multiple tasks (processes or threads) at once. It gives the illusion of parallelism by rapidly switching between tasks using a single processor. There are two types: preemptive multitasking (where the OS controls the switching) and cooperative multitasking (where tasks yield control voluntarily).Multiprocessing, on the other hand, involves using two or more processors (or cores) to execute multiple tasks simultaneously. Each processor can run a different task, achieving true parallelism and significantly improving performance for compute-heavy applications.13. What is Starvation and Aging in OS?Starvation happens when a process keeps getting ignored and never gets a chance to run because other processes are always prioritized.Aging is a method used to prevent starvation, where the waiting process's priority is slowly increased over time, making sure it eventually gets a chance to run.14. What is Semaphore and Mutex?A semaphore is a synchronization tool used to control access to shared resources by multiple processes. It has a counter that represents the number of available resources, and processes can signal or wait based on that value. A mutex (mutual exclusion) is a lock that ensures only one thread or process can access a resource at a time. Unlike semaphores, mutexes have ownership, meaning only the thread that locked the mutex can unlock it. The main difference is that semaphores can manage multiple resources, while mutexes provide exclusive access to a single resource.15. What is RAID? What are the different RAID levels?RAID (Redundant Array of Independent Disks) is a technology that combines multiple physical disk drives into a single logical unit to improve performance, data redundancy, or both. RAID is used to ensure better data reliability, faster data access, and improved storage capacity. RAID has different levels:RAID-0 (Stripping)RAID-1 (Mirroring)RAID-2 (Bit-Level Stripping with Dedicated Parity)RAID-3 (Byte-Level Stripping with Dedicated Parity)RAID-4 (Block-Level Stripping with Dedicated Parity)RAID-5 (Block-Level Stripping with Distributed Parity)RAID-6 (Block-Level Stripping with two Parity Bits)To know more about the topic refer to RAID 16. Explain Address Binding. What are the different types of Address Binding?Address binding is the process of mapping a logical address (used by a program) to a physical address (in memory) during the execution of a program. This is necessary for the system to access data and instructions in the correct locations in physical memory.Types of Address Binding:Compile-Time Binding: The address is determined and fixed when the program is compiled. The program must run at the same memory location each time.Load-Time Binding: The address is determined when the program is loaded into memory, allowing the program to be loaded at different memory locations each time it runs.Execution-Time Binding: The address is determined during the program's execution, allowing for dynamic address changes, such as in the case of virtual memory.17. What is cache?A cache is a small, high-speed memory storage used to temporarily store frequently accessed data or instructions. It helps improve the performance of a system by reducing the time it takes to retrieve data from slower memory, like RAM or disk. Caches are typically used in CPUs, disk storage, and web browsers to speed up data access.18. What are the different kinds of operations possible on semaphore?The process of using Semaphores provides two operations:wait (P):The wait operation decrements the value of the semaphore.signal (V):The signal operation increments the value of the semaphore.19. Explain zombie process?Azombie processis a process that has completed its execution but still remains in the process table because its parent process has not yet read its exit status. It is called a zombie because it is no longer active or running, but it still exists as a placeholder in the system. Zombie processes do not consume system resources like CPU or memory, but they can clutter the process table if not handled properly, especially if many zombie processes accumulate.20. What is a process? What are the different states of a process?Process is basically a program under execution. When a program is loaded into memory, it becomes a process and is divided into four sections: stack, heap, text, and data. States of Process:States of a ProcessDifferent 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 processs instructions.Waiting:In this state, the process cannot run because it just waits for some event to occurReady: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.21. What is a deadlock in OS? What are the necessary conditions for a deadlock?Deadlock is a situation in computing where two or more processes are blocked because each process is waiting for the resources held by other processes, creating a cycle of dependencies with no way out. In this state, the processes are stuck and cannot continue their execution.There are basically four necessary conditions for deadlock as given below:Mutual ExclusionHold and WaitNo Pre-emptionCircular Wait or Resource Wait22. 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 computerDatabase Management System Interview Questions23. What is DBMS? What are the advantages of DBMS over traditional file-based systems?DBMS (Database Management System) is software that manages databases, allowing users to store, retrieve, and manipulate data efficiently. It provides an interface between users and databases, ensuring data is organized, secure, and easily accessible.Advantages of DBMS over traditional file-based systems:Data redundancy and inconsistencyDifficulty in accessing dataData isolation multiple files and formatsIntegrity problemsAtomicity of updatesConcurrent access by multiple usersSecurity problems24. What is RDBMS and what are the advantages?RDBMS (Relational Database Management System) is a type of DBMS that stores data in a structured format using rows and columns, typically in tables. It uses Structured Query Language (SQL) for managing and manipulating relational data. Examples of RDBMS include MySQL, PostgreSQL, and Oracle.Advantages of RDBMS:Data Integrity: RDBMS enforces integrity constraints (e.g., primary keys, foreign keys) to maintain accurate and consistent data.Data Redundancy Control: It minimizes data duplication by organizing data into related tables, reducing redundancy.Flexibility and Scalability: RDBMS can handle large volumes of data efficiently, supporting complex queries and large datasets.Ease of Access: With SQL, users can easily retrieve, update, and manipulate data in a consistent and standardized way.Data Security: RDBMS provides robust security features like user access control and encryption to protect sensitive data.Backup and Recovery: RDBMS supports backup and recovery mechanisms to ensure data can be restored in case of system failure.Concurrency Control: Multiple users can access and modify the data concurrently without conflicts or data inconsistency.25. What is the difference between Horizontal and Vertical Scaling?Horizontal scaling (scaling out) involves adding more machines or servers to distribute the load, increasing system capacity by expanding across multiple nodes. Vertical scaling (scaling up) involves adding more resources (like CPU, RAM, or storage) to a single machine, improving its performance. The main difference is that horizontal scaling adds more systems, while vertical scaling upgrades a single systems capabilities.To know more about the topic refer to Horizontal and Vertical Scaling In Databases26. What are different types of keys in DBMS?In a DBMS, keys are used to uniquely identify records and establish relationships between tables. They are essential for ensuring data integrity and efficient data retrieval.Types of Keys:Primary Key: A field or combination of fields that uniquely identifies each record in a table. It cannot have NULL values.Foreign Key: A field in one table that links to the primary key of another table, establishing a relationship between them.Candidate Key: A set of attributes that can uniquely identify a record, from which one is chosen as the primary key.Unique Key: A field or combination of fields that ensures all values in a column are unique, but unlike the primary key, it can accept NULL values.Composite Key: A key made up of two or more attributes to uniquely identify a record in a table.Superkey: A set of attributes that uniquely identifies a record in a table, which may include unnecessary attributes. Every primary key is a superkey, but not every superkey is a primary key.Alternate Key: A candidate key that was not chosen as the primary key but can still uniquely identify records.27. What are the difference between DDL, DML, and DCL in SQL?Heres the updated explanation with query syntax for DDL, DML, and DCL:1. DDL (Data Definition Language):Used to define and modify the structure of database objects like tables, views, and schemas.Common Queries:CREATE: Defines new database objects. CREATE TABLE table_name (column1 datatype, column2 datatype, ...);ALTER: Modifies an existing database object. ALTER TABLE table_nameADD COLUMN column_name data_type;DROP: Deletes a database object. DROP TABLE table_name;TRUNCATE: Removes all records from a table but keeps its structure. TRUNCATE TABLE table_name;2. DML (Data Manipulation Language):Used to manage and manipulate data within the database (insert, update, delete, and retrieve).Common Queries:INSERT: Adds new data to a table. INSERT INTO table_name (column1, column2, ...)VALUES (value1, value2, ...);UPDATE: Modifies existing data. UPDATE table_nameSET column1 = value1, column2 = value2WHERE condition;DELETE: Removes data from a table. DELETE FROM table_nameWHERE condition;SELECT: Retrieves data from the database. SELECT column1, column2, ...FROM table_nameWHERE condition;3. DCL (Data Control Language):Used to control access to data and manage permissions.Common Queries:GRANT: Assigns new privileges to a user account, allowing access to specific database objects, actions, or functions.GRANT privilege_type [(column_list)] ON [object_type] object_name TO user [WITH GRANT OPTION];REVOKE: Removes previously granted privileges from a user account, taking away their access to certain database objects or actions.REVOKE [GRANT OPTION FOR] privilege_type [(column_list)] ON [object_type] object_name FROM user [CASCADE];Key Differences:DDL deals with creating and modifying the database structure.DML deals with adding, updating, retrieving, and deleting data.DCL deals with managing permissions and access control for users.28. What is a Transaction? What are ACID properties?A transaction in a database is a sequence of operations performed as a single unit of work. It ensures that the database transitions from one consistent state to another. Transactions are crucial in ensuring data integrity, consistency, and reliability, especially in multi-user environments.ACID PropertiesAtomicity: A transaction is atomic, meaning it is treated as a single, indivisible unit. Either all operations in the transaction are completed successfully, or none are. If an error occurs, the transaction is rolled back to maintain consistency.Consistency: The database must always transition from one consistent state to another. After a transaction, the database should remain in a valid state, adhering to all defined rules, constraints, and relationships.Isolation: Transactions are isolated from one another. The operations of one transaction are not visible to others until the transaction is completed. This prevents issues like dirty reads, non-repeatable reads, and phantom reads.Durability: Once a transaction is committed, its effects are permanent, even if the system crashes. The changes made by a committed transaction are saved to stable storage.To know more about the topic refer to ACID Properties29. What is Normalization? What are the different types of normalization? Normalization is the process of organizing data in a database to reduce redundancy and improve data integrity. It involves dividing a large table into smaller, manageable tables and defining relationships between them to ensure that data is stored efficiently.Types of Normalization:First Normal Form (1NF): Ensures that each column contains only atomic (indivisible) values and that each record is unique.Second Normal Form (2NF): Achieved by ensuring that the table is in 1NF and that all non-key attributes are fully dependent on the primary key.Third Normal Form (3NF): Ensures that the table is in 2NF and that there are no transitive dependencies (non-key attributes depending on other non-key attributes).Boyce-Codd Normal Form (BCNF): A stronger version of 3NF, where every determinant (an attribute that determines another attribute) is a candidate key.To know more about the topic refer to Normalization in DBMS30. What is Denormalization?Denormalization is the process of intentionally introducing redundancy into a database by merging tables or adding redundant data to improve performance, typically for read-heavy applications. While normalization reduces redundancy to ensure data integrity, denormalization may be used to optimize query performance, particularly when complex joins and aggregations are frequently required. Denormalization can speed up data retrieval at the cost of increased storage and potential data anomalies during updates.31. What is Concurrency Control Protocol (CCP)?Concurrency Control Protocol (CCP) is used in database systems to manage simultaneous transactions, ensuring data consistency and preventing conflicts like lost updates or temporary inconsistencies. Common techniques include:Lock-based protocols: locking data to prevent conflictsTimestamp-based protocols: enforcing transaction orderOptimistic concurrency control: allowing transactions to execute without locks but checking for conflicts before committingMultiversion concurrency control (MVCC): using multiple versions of data to avoid conflictsThese protocols help maintain the ACID properties and ensure safe transaction execution.32. Explain ER-Model?The Entity-Relationship (ER) Model is a conceptual framework used to design databases by representing entities (objects) and their relationships. Key components include entities (real-world objects like a person or product), attributes (properties of entities), and relationships (associations between entities). The model uses primary keys to uniquely identify entities and defines cardinality (the number of associations between entities, such as one-to-one, one-to-many, or many-to-many). ER diagrams visually represent these elements to help organize and structure data efficiently.33. What is a JOIN? What are the different types of JOIN?A JOIN in SQL is used to combine data from two or more tables based on a related column between them. It allows retrieving data that is spread across multiple tables by linking them through their relationships.Types of JOINs:INNER JOIN: Returns only the rows that have matching values in both tables. If there is no match, the row is excluded.SELECT * FROM table1INNER JOIN table2 ON table1.id = table2.id;LEFT JOIN (or LEFT OUTER JOIN): Returns all rows from the left table, and the matching rows from the right table. If there is no match, NULL values are returned for the right table's columns.SELECT * FROM table1LEFT JOIN table2 ON table1.id = table2.id;RIGHT JOIN (or RIGHT OUTER JOIN): Returns all rows from the right table, and the matching rows from the left table. If there is no match, NULL values are returned for the left table's columns.SELECT * FROM table1RIGHT JOIN table2 ON table1.id = table2.id;FULL JOIN (or FULL OUTER JOIN): Returns all rows when there is a match in either left or right table. If there is no match, NULL values are returned for non-matching columns from either table.SELECT * FROM table1FULL JOIN table2 ON table1.id = table2.id;CROSS JOIN: Returns the Cartesian product of both tables, meaning all possible combinations of rows from both tables. It doesn't require a condition to join the tables.SELECT * FROM table1CROSS JOIN table2;Each type of JOIN serves different purposes depending on how you want to combine the data from multiple tables.34. What is Data Abstraction? What are the different levels of data abstraction?Data abstraction is the process of hiding the complex details of data storage and providing a simplified interface for users to interact with the data. It is achieved through different levels:Physical Level: Describes how data is stored on the hardware (disk, memory) and focuses on storage details.Logical Level: Describes the structure of the data, such as tables and relationships, without worrying about physical storage.View Level: Provides different user-specific views of the data, ensuring users only see relevant information.This abstraction helps in simplifying database management and user interaction.35. What are the differences between DROP, TRUNCATE and DELETE commands?DROP removes an entire table or database object, including its structure and data, and cannot be rolled back. TRUNCATE removes all rows from a table but keeps the structure intact, and is faster than DELETE. DELETE removes specific rows based on a condition and can be rolled back, but is slower as it logs individual row deletions.36. Explain Entity, Entity Type, and Entity Set in DBMS?In DBMS, the concepts of Entity, Entity Type, and Entity Set are fundamental for understanding the structure and organization of data.Entity: An entity is a real-world object or concept that is distinguishable from other objects. It can be a person, place, thing, or event that has a distinct existence in the context of a database. For example, a "Student" or "Car" could be entities.Entity Type: An entity type defines a category or class of entities that share common properties or attributes. It is a blueprint for creating entities. For example, "Student" can be an entity type, with attributes like Student ID, Name, and Date of Birth.Entity Set: An entity set is a collection of similar entities that share the same entity type. For instance, the set of all students in a university would form an "Entity Set" of the entity type "Student."37. What are the different types of relationships in the DBMS?In a DBMS, relationships define how entities are connected to each other. There are three main types of relationships based on how entities interact within the database:One-to-One: In this relationship, each entity in one table is associated with exactly one entity in another table.One-to-Many: In this relationship, one entity in a table is associated with multiple entities in another table, but each entity in the second table is related to only one entity in the first table.Many-to-Many: In this relationship, multiple entities in one table are associated with multiple entities in another table. It often requires a junction table to establish the relationship.These relationships help organize how data in different tables is connected and ensure referential integrity within the database.A lock in a database is a mechanism that ensures controlled access to resources during transactions, preventing conflicts like data inconsistency or lost updates. There are two main types of locks: Shared Lock and Exclusive Lock. Shared Lock: Allows multiple transactions to read a resource simultaneously but prevents any transaction from modifying it. This means many transactions can read the same data, but none can alter it until the shared lock is released. Exclusive Lock: Gives a transaction the ability to both read and modify a resource, while preventing any other transaction from accessing or altering the data, even for reading, until the lock is released. Therefore, while a shared lock supports concurrent reading, an exclusive lock ensures exclusive access to a resource for a transaction.39. What are aggregate functions in SQL?Aggregate functions perform calculations on a set of values and return a single value. Common aggregate functions include:COUNT(): Returns the number of rows.SUM(): Returns the total sum of values.AVG(): Returns the average of values.MIN(): Returns the smallest value.MAX(): Returns the largest value.40. What is the difference between UNION and UNION ALL?UNION:Removes duplicate rows from the result set, ensuring only unique rows are returned.UNION ALL:Includes all rows from each query, including duplicates.Performance-wise,UNION ALLis faster because it doesnt require an additional step to remove duplicates.To read more about SQL Interview Questions, Refer Here.Computer Networks Interview Questions41. What is Network? What are the types of Networks (LAN, WAN, MAN)?A network is a system of interconnected devices that allows sharing of data and resources. Types of Networks:LAN (Local Area Network): Covers a small area like a building or office, enabling fast data sharing among devices within proximity.WAN (Wide Area Network): Spans a large geographical area, connecting multiple LANs across cities, countries, or continents, often using the internet.MAN (Metropolitan Area Network): Covers a city or large campus, connecting multiple LANs within a metropolitan area.42. What is Network Topology? What are the types of Network Topology?Network topology refers to the physical or logical arrangement of devices (like computers, routers, switches) and their connections within a network. It defines how devices are connected and how data flows between them, impacting network performance, management, and reliability.Types of Network Topology:Bus Topology: All devices are connected to a single central cable (the bus). Data sent by one device is available to all other devices, but only the intended recipient receives it. It's simple but can become slow with many devices.Bus TopologyStar Topology: All devices are connected to a central hub or switch. Data from one device passes through the hub to reach others. It's easy to manage and troubleshoot but depends on the central device for operation.Star TopologyRing Topology: Devices are connected in a circular fashion. Data travels in one direction around the ring until it reaches the destination. It's efficient but a failure in any part of the ring can disrupt the entire network.Ring TopologyMesh Topology: Every device is connected to every other device, ensuring high redundancy and reliability. It offers fault tolerance but can be complex and expensive to set up.Mesh TopologyHybrid Topology: A combination of two or more different topologies to leverage their advantages. For example, combining star and bus topologies.Hybrid TopologyTree Topology: A variation of star topology where groups of star-configured networks are connected to a central bus. It allows hierarchical data organization and is scalable.Tree Topology43. What is TCP Model?The TCP Model (Transmission Control Protocol Model), also known as the TCP/IP model, is a conceptual framework used to understand how different protocols work together to enable communication over a network, especially the internet. It is a four-layer model that standardizes the functions required for communication in computer networks.Application Layer: Manages network services for applications (e.g., HTTP, FTP, SMTP).Transport Layer: Ensures reliable data transfer (e.g., TCP, UDP).Internet Layer: Handles logical addressing and routing (e.g., IP).Network Interface Layer: Deals with physical data transmission (e.g., Ethernet, Wi-Fi).To know more about the topic refer to TCP Model.44. What is OSI Model? What are the different layers of OSI Model?The OSI Model (Open Systems Interconnection Model) is a conceptual framework used to understand and standardize the functions of a network in seven distinct layers. It serves as a reference model for how different network protocols interact and provides a guideline for designing and implementing network systems.Layers of OSI ModelTo know more about the topic refer to OSI Model.45. What is the difference between OSI and TCP Model? TCP/IPOSITCP refers to Transmission Control Protocol.OSI refers to Open Systems Interconnection.TCP/IP uses both the session and presentation layer in the application layer itself.OSI uses different session and presentation layers.TCP/IP follows connectionless a horizontal approach.OSI follows a vertical approach.The Transport layer in TCP/IP does not provide assurance delivery of packets.In the OSI model, the transport layer provides assurance delivery of packets.Protocols cannot be replaced easily in TCP/IP model.While in the OSI model, Protocols are better covered and are easy to replace with the technology change.TCP/IP model network layer only provides connectionless (IP) services. The transport layer (TCP) provides connections.Connectionless and connection-oriented services are provided by the network layer in the OSI model.46. What is the difference between gateway and router?A router connects multiple networks and routes data between them based on IP addresses, typically within similar networks like LANs and WANs. A gateway, on the other hand, connects different types of networks (often with different protocols) and translates data between them, allowing communication across diverse systems. While routers manage data traffic within a network, gateways provide access and translation between different network architectures.47. What is MAC Address?A MAC address is a unique identifier assigned to a network device's hardware (NIC) for communication within a local network. It is a 48-bit address used at the data link layer (Layer 2) to ensure correct data delivery between devices.48. What is the difference between hub and switch?A hub is a basic networking device that broadcasts data to all connected devices, regardless of the intended recipient, which can lead to collisions and reduced efficiency. A switch, on the other hand, intelligently forwards data only to the specific device it is intended for, improving network performance and reducing collisions.49. What is the difference between IPv4 and IPv6 Protocols?IPv4 (Internet Protocol version 4) uses 32-bit addresses, allowing for about 4.3 billion unique addresses. It is widely used but limited in address space. IPv6 (Internet Protocol version 6) uses 128-bit addresses, providing a virtually unlimited number of unique addresses, and it also includes enhanced features like better security and more efficient routing.To know more about the topic refer to IPv4 and IPv6IP addressing is the system that assigns a unique numerical identifier (IP address) to each device on a network, allowing them to communicate. IP addresses are divided into two types: IPv4 (32-bit) and IPv6 (128-bit).Subnetting is the process of dividing a larger IP network into smaller, manageable sub-networks (subnets). This improves network performance, security, and IP address utilization by allocating different ranges of IP addresses to different segments of a network. Subnetting involves using subnet masks to identify the network and host portions of an IP address.To know more about this topic refer to IP addressing and Subnetting51. What is 3-Way Handshaking?3-way handshaking is a process used in the TCP protocol to establish a reliable connection between a client and a server. This process ensures both sides are ready for data transmission. It involves three steps:SYN: The client sends a synchronization (SYN) message to the server to initiate the connection.SYN-ACK: The server responds with a synchronization-acknowledgment (SYN-ACK) message, confirming the connection request.ACK: The client sends an acknowledgment (ACK) message back to the server, completing the connection setup.To know more about the topic refer to 3 way handshaking52. What is HTTP and HTTPS protocol?HTTP (Hypertext Transfer Protocol) is a protocol used for transferring data over the web. It defines how messages are formatted and transmitted between a client (like a browser) and a server. However, it is not secure, meaning data is transmitted in plain text.HTTPS (Hypertext Transfer Protocol Secure) is the secure version of HTTP. It uses SSL/TLS encryption to secure the data transfer, ensuring confidentiality and data integrity between the client and server, preventing unauthorized access or tampering with the data.53. What are the differences between TCP and UDP protocols?TCP (Transmission Control Protocol) is a reliable, connection-oriented protocol that ensures data is delivered accurately. It establishes a connection with a 3-way handshake and uses error-checking, acknowledgments, and retransmissions to guarantee correct delivery. This makes TCP suitable for applications like web browsing (HTTP) and file transfers (FTP), but it is slower due to its overhead.UDP (User Datagram Protocol) is a connectionless and faster protocol that prioritizes speed over reliability. It does not establish a connection or guarantee data delivery, making it suitable for real-time applications like video streaming, online gaming, and VoIP calls, where speed is more important than perfect accuracy.To know more about the topic refer to TCP and UDP Protocols54. What are the different types of delays?Delay in networking refers to the time it takes for data to travel from the source to the destination. There are several types of delays that can affect network performance:Propagation Delay: The time it takes for a signal to travel from the sender to the receiver through the transmission medium, dependent on the distance and the speed of the medium.Transmission Delay: The time required to push all the data packets onto the link, determined by the size of the packet and the bandwidth of the link.Processing Delay: The time taken by network devices (routers, switches) to process data, check for errors, and make routing decisions.Queuing Delay: The time data spends in queues at routers or switches when the network is congested or when there is high traffic. This delay varies depending on the network load.To know more about the topic refer to Different Types of Delays55. What is VPN? What are the advantages and disadvantages of VPN?A VPN (Virtual Private Network) is a service that creates a secure and encrypted connection over a public network (like the internet), allowing users to send and receive data privately and securely as if they were on a private network.Advantages:Enhanced Security: Encrypts data, protecting it from hackers and eavesdropping.Privacy: Hides your IP address, ensuring anonymity while browsing.Access to Restricted Content: Allows access to region-restricted websites and services.Disadvantages:Reduced Speed: Encryption and routing through VPN servers can slow down internet speed.Cost: Quality VPN services often require a subscription fee.Complexity: Setting up and configuring a VPN can be complicated for some users.To know more about the topic refer to VPN (Virtual Private Network) .56. What does ping command do?Ping (Packet Internet or Inter-Network Groper) is a basic internet command that is used to check the connectivity, reachability between IP-networked devices. To know more about the topic refer to Ping command.57. What is the SMTP Protocol?SMTP (Simple Mail Transfer Protocol) is a protocol used for sending and relaying emails across networks. It defines the rules for how email messages are sent from the sender's email client to the mail server and then to the recipient's mail server. SMTP operates on a client-server model, typically using port 25, and ensures the reliable delivery of email messages over the internet.To know more about the topic refer to SMTP Protocol58. What happens when you enter "google.com"?When you enter "google.com" in your browser, several steps happen to connect you to the website:DNS Lookup: The browser checks if the domain name "google.com" has a corresponding IP address by querying a DNS (Domain Name System) server. The DNS server returns the IP address of Google's web server.TCP/IP Connection: Using the IP address, the browser establishes a connection to Google's server using the TCP/IP protocol (via port 80 for HTTP or port 443 for HTTPS).HTTP/HTTPS Request: The browser sends an HTTP/HTTPS request to the server, asking for the webpage.Server Response: Google's server processes the request and sends back the requested data (HTML, CSS, JavaScript, images, etc.) via the same connection.Rendering: The browser processes the data, rendering the webpage so that you can see and interact with google.com.59.What is an IPv4 address? What are the different classes of IPv4?An IP address is a 32-bit dynamic address of a node in the network. An IPv4 address has 4 octets of 8-bit each with each number with a value up to 255.IPv4 classes are differentiated based on the number of hosts it supports on the network. There are five types of IPv4 classes and are based on the first octet of IP addresses which are classified as Class A, B, C, D, or E.Classes In IP Addressing60.What is DNS?DNS stands forDomain Name System.It translates domain names into IP addresses, which are used by browsers to load web pages.Example: Human-readable domain names like www.google.com into machine-readable IP addresses such as 142.250.190.14.To know more about the topic refer to DNS.Must Read:ConclusionMastering the core concepts of Operating Systems (OS), Database Management Systems (DBMS), and Computer Networks (CN) is essential for excelling in technical interviews. The 50+ interview questions presented in this article provide a well-rounded review of these topics, helping you sharpen your understanding and boost your confidence. By revisiting key concepts, and familiarizing yourself with common interview patterns, you'll be better equipped to tackle a wide range of questions during your interview. This article aims to be a helpful resource in your journey to securing your dream job. Good luck! An Operating System acts as a communication interface between the user and computer hardware. Its purpose is to provide a platform on which a user can execute programs conveniently and efficiently. The main goal of an operating system is to make the computer environment more convenient to use and to utilize resources most efficiently.Operating System handles the following responsibilities:Controls all the computer resources.Provides valuable services to user programs.Coordinates the execution of user programs.Provides resources for user programs.Provides an interface (virtual machine) to the user.Hides the complexity of software.Supports multiple execution modes.Monitors the execution of user programs to prevent errors.Functions of an Operating SystemFunction of OS1. Process ManagementProcess management in operating system is about managing processes. A Process is a running program. The life cycle of process is from the moment program start until it finishes. Operating system makes sure each process: gets its turn to use the CPU synchronized when needed has access to the resources it needs, like memory, files, and input/output devices.It also handles issues like process coordination and communication, while preventing conflicts such as deadlocks. This way, the OS ensures smooth multitasking and efficient resource use.Process State DiagramCore Functions in Process Management:Process SchedulingAllocates CPU time to processes based on scheduling algorithms like Round Robin or Priority Scheduling.Ensures fair distribution of CPU time, avoiding starvation of lower-priority processes.Maximizes CPU utilization by determining which process runs at any given time.Process SynchronizationCoordinate multiple processes to ensure orderly execution and prevent conflicts.Prevents race conditions by ensuring that only one process can access a shared resource at a time.Uses synchronization mechanisms like locks, semaphores, and monitors to coordinate process access.Deadlock HandlingPrevents deadlocks by using strategies like resource allocation graphs or avoiding circular wait conditions.Detects deadlocks when they occur, allowing the system to identify and resolve the issue.Recovers from deadlocks by aborting or rolling back processes to free up resources.Inter-Process Communication (IPC):Facilitates communication between processes through shared memory, allowing processes to exchange data directly.Uses message passing to send data between processes in different address spaces.Enables efficient data exchange and coordination in a multitasking environment, improving system performance.Read more about Process Management in OS2. Memory ManagementMemory management is an essential task of the operating system that handles the storage and organization of data in both main (primary) memory and secondary storage. The OS ensures that memory is allocated and deallocated properly to keep programs running smoothly. It also manages the interaction between volatile main memory and non-volatile secondary storage.Memory ManagementKey Activities in Memory Management:Main Memory ManagementMemory Allocation: Assigns memory to processes using techniques like paging and segmentation.Memory Deallocation: Frees memory when no longer needed.Memory Protection: Prevents processes from accessing each others memory.Virtual Memory: Uses disk space as extra memory to run larger processes.Fragmentation: Manages wasted memory space (internal/external) through compaction.Secondary Memory ManagementDisk Space Allocation: Organizes how files are stored on the disk (contiguous, linked, indexed).File System Management: Manages files and directories for efficient data access.Free Space Management: Tracks available space on the disk.Disk Scheduling: Organizes the order of disk read/write requests.Backup and Recovery: Ensures data is backed up and can be restored after failure.Read more about Memory Management in OS3. File System ManagementFile management in the operating system ensures the organized storage, access and control of files. The OS abstracts the physical storage details to present a logical view of files, making it easier for users to work with data. It manages how files are stored on different types of storage devices (like hard drives or SSDs) and ensures smooth access through directories and permissions.File System ManagementFile System Management includes managing of:File AttributesFile Name: Identifies the file with a name and extension (e.g., .txt, .jpg).File Type: Defines the format of the file (e.g., text, image, executable).Size: The amount of storage the file occupies.Permissions: Determines who can read, write, or execute the file.File TypesText Files: Contain human-readable content (e.g., .txt, .md).Binary Files: Store data in binary format (e.g., .jpg, .mp3).Executable Files: Contain program code (e.g., .exe, .out).Operations on FilesCreate: Allows users to create new files.Read: Opens files to read their contents.Write: Modifies the contents of a file.Delete: Removes a file from the system.Access MethodsSequential Access: Reads data in order, from start to finish.Direct Access: Jumps to a specific part of the file.Indexed Access: Uses an index for quick data retrieval.Read more about File System Management in OS4. Device Management (I/O System)Device management of an operating system handles the communication between the system and its hardware devices, like printers, disks or network interfaces. The OS provides device drivers to control these devices, using techniques like Direct Memory Access (DMA) for efficient data transfer and strategies like buffering and spooling to ensure smooth operation.Device ManagementMajor components in Device Management:Device Drivers: The operating system uses device drivers to interact with hardware devices. There are two types of device drivers:Kernel-space drivers run in the OS kernel, offering direct access to hardware.User-space drivers run outside the kernel and are more isolated, providing safety but less performance.Buffering & Caching:Buffering temporarily stores data in memory to manage differences in device speeds. Block devices (e.g., hard drives) use larger blocks of data for buffering, while character devices (e.g., keyboards, mice) use smaller, byte-by-byte buffering.Caching improves access speed by storing frequently accessed data in a faster storage medium (like RAM).Spooling: Spooling manages data waiting to be processed, particularly in devices like printers. The OS places print jobs in a spool (a temporary storage area), allowing the CPU to continue other tasks while the printer works through the queue. Other examples include mail spooling (for managing outgoing email) and batch-job spooling (for managing scheduled tasks).Read more about Device Management in OS5. Protection and SecurityProtection and security mechanisms in an operating system are designed to safeguard system resources from unauthorized access or misuse. These mechanisms control which processes or users can access specific resources (such as memory, files, and CPU time) and ensure that only authorized users can perform specific actions. While protection ensures proper access control, security focuses on defending the system against external and internal attacks.SecurityAccess Control: The operating system ensures that processes and users can only access resources they are authorized to. This is achieved through mechanisms like memory-addressing hardware (which keeps processes within their own address space) and timer interrupts (which prevent processes from monopolizing the CPU).User Authentication: A system identifies users through user IDs (UIDs) or Security IDs (SIDs). During login, the operating system verifies the users credentials, ensuring that only authorized users can access their data or system resources.Resource Protection: Mechanisms like file protection (ensuring only authorized users can access or modify files) and device protection (restricting direct access to device-control registers) ensure the integrity of system resources, preventing unauthorized or harmful use.Security Against Attacks: Security mechanisms defend against external threats like viruses, worms, denial-of-service attacks, and identity theft. These attacks can misuse system resources, steal sensitive data, or disrupt system operation. The OS works to prevent these threats and minimize damage.Read more about Security and Protection in OS6. Additional Functions of the Operating SystemBeyond core tasks like process and memory management, operating systems also focus on system performance, resource utilization and error detection. These functions ensure smooth operation, efficient resource allocation and system reliability.Control Over System Performance: The OS monitors system performance by recording delays between service requests and system responses. This helps identify bottlenecks and optimize resource usage, ensuring efficient system operation.Job Accounting: The OS keeps track of time and resources used by various tasks and users. This information is valuable for auditing, billing, and optimizing resource allocation among users and applications.Error-Detecting Aids: The OS produces dumps, traces, error messages, and other debugging aids to detect and diagnose errors. This proactive approach helps maintain system stability and aids in troubleshooting issues. Operating System Tutorial Batch Processing Operating System Multiprogramming in Operating System Time Sharing Operating System What is a Network Operating System? Real Time Operating System (RTOS) Introduction of Process Management Process Table and Process Control Block (PCB) Operations on Processes Process Schedulers in Operating System Inter Process Communication (IPC) Context Switching in Operating System Preemptive and Non-Preemptive Scheduling Peterson's Algorithm in Process Synchronization Semaphores in Process Synchronization Semaphores and its types Producer Consumer Problem using Semaphores | Set 1 Readers-Writers Problem | Set 1 (Introduction and Readers Preference Solution) Dining Philosopher Problem Using Semaphores Hardware Synchronization Algorithms : Unlock and Lock, Test and Set, Swap Introduction of Deadlock in Operating System Conditions for Deadlock in Operating System Banker's Algorithm in Operating System Wait For Graph Deadlock Detection in Distributed System Handling Deadlocks Deadlock Prevention And Avoidance Deadlock Detection And Recovery Deadlock Ignorance in Operating System Recovery from Deadlock in Operating System An Operating system is an interface between users and the hardware of a computer system. It is a system software that is viewed as an organized collection of software consisting of procedures and functions, providing an environment for the execution of programs. The operating system manages system software and computer hardware resources. It allows computing resources to be used in an efficient way. Programs interact with computer hardware with the help of operating system. A user can interact with the operating system by making system calls or using OS commands.Important Components of the Operating SystemProcess ManagementFile ManagementCommand InterpreterSystem CallsSignalsNetwork ManagementSecurity ManagementI/O Device ManagementSecondary Storage ManagementMain Memory ManagementProcess ManagementA process is a program in execution. It consists of the followings:Executable programProgram dataStack and stack pointerProgram counter and other CPU registersDetails of opened filesA process can be suspended temporarily and the execution of another process can be taken up. A suspended process can be restarted later. Before suspending a process, its details are saved in a table called the process table so that it can be executed later on. An operating system supports two system calls to manage processes Create and Kill -Create a system call used to create a new process.Kill system call used to delete an existing process.Five State Process ModelA process can create a number of child processes. Processes can communicate among themselves either using shared memory or by message-passing techniques. Two processes running on two different computers can communicate by sending messages over a network.Files ManagementFiles are used for long-term storage. Files are used for both input and output. Every operating system provides a file management service. This file management service can also be treated as an abstraction as it hides the information about the disks from the user. The operating system also provides a system call for file management. The system call for file management includes:File creationFile deletionRead and Write operationsFiles are stored in a directory. System calls provide to put a file in a directory or to remove a file from a directory. Files in the system are protected to maintain the privacy of the user. Below shows the Hierarchical File Structure directory.File Structure DirectoryCommand InterpreterThere are several ways for users to interface with the operating system. One of the approaches to user interaction with the operating system is through commands. Command interpreter provides a command-line interface. It allows the user to enter a command on the command line prompt (cmd). The command interpreter accepts and executes the commands entered by a user. For example, a shell is a command interpreter under UNIX. The commands to be executed are implemented in two ways:The command interpreter itself contains code to be executed.The command is implemented through a system file. The necessary system file is loaded into memory and executed.System CallsSystem calls provide an interface to the services made by an operating system. The user interacts with the operating system programs through System calls. These calls are normally made available as library functions in high-level languages such as C, Java, Python etc. It provides a level of abstraction as the user is not aware of the implementation or execution of the call made. Details of the operating system is hidden from the user. Different hardware and software services can be availed through system calls.System calls are available for the following operations:Process ManagementMemory ManagementFile OperationsInput / Output Operations SignalsSignals are used in the operating systems to notify a process that a particular event has occurred. Signals are the software or hardware interrupts that suspend the current execution of the task. Signals are also used for inter-process communication. A signal follows the following pattern :A signal is generated by the occurrence of a particular event it can be the clicking of the mouse, the execution of the program successfully or an error notifying, etc.A generated signal is delivered to a process for further execution.Once delivered, the signal must be handled.A signal can be synchronous and asynchronous which is handled by a default handler or by the user-defined handler.The signal causes temporarily suspends the current task it was processing, saves its registers on the stack, and starts running a special signal handling procedure, where the signal is assigned to it.Network ManagementThe complexity of networks and services has created modern challenges for IT professionals and users. Network management is a set of processes and procedures that help organizations to optimize their computer networks. Mainly, it ensures that users have the best possible experience while using network applications and services.Network management is a fundamental concept of computer networks. Network Management Systems is a software application that provides network administrators with information on components in their networks. It ensures the quality of service and availability of network resources. It also examines the operations of a network, reconstructs its network configuration, modifies it for improving performance of tasks.Security ManagementThe security mechanisms in an operating system ensure that authorized programs have access to resources, and unauthorized programs have no access to restricted resources. Security management refers to the various processes where the user changes the file, memory, CPU, and other hardware resources that should have authorization from the operating system.I/O Device ManagementThe I/O device management component is an I/O manager that hides the details of hardware devices and manages the main memory for devices using cache and spooling. This component provides a buffer cache and general device driver code that allows the system to manage the main memory and the hardware devices connected to it. It also provides and manages custom drivers for particular hardware devices.The purpose of the I/O system is to hide the details of hardware devices from the application programmer. An I/O device management component allows highly efficient resource utilization while minimizing errors and making programming easy on the entire range of devices available in their systems.Secondary Storage ManagementBroadly, the secondary storage area is any space, where data is stored permanently and the user can retrieve it easily. Your computers hard drive is the primary location for your files and programs. Other spaces, such as CD-ROM/DVD drives, flash memory cards, and networked devices, also provide secondary storage for data on the computer. The computers main memory (RAM) is a volatile storage device in which all programs reside, it provides only temporary storage space for performing tasks. Secondary storage refers to the media devices other than RAM (e.g. CDs, DVDs, or hard disks) that provide additional space for permanent storing of data and software programs which is also called non-volatile storage.Main Memory ManagementMain memory is a flexible and volatile type of storage device. It is a large sequence of bytes and addresses used to store volatile data. Main memory is also called Random Access Memory (RAM), which is the fastest computer storage available on PCs. It is costly and low in terms of storage as compared to secondary storage devices. Whenever computer programs are executed, it is temporarily stored in the main memory for execution. Later, the user can permanently store the data or program in the secondary storage device.ConclusionIn Conclusion, the components of an operating system work together to make a computer system function easily. Each component, from process management to security, plays a unique role in handling tasks, managing resources, and ensuring data safety. By coordinating these functions, the operating system provides a stable and efficient environment for users and applications. os.environin Python is a mapping object that represents the users OS environmental variables. It returns a dictionary having the users environmental variable as key and their values as value.os.environbehaves like a Python dictionary, so all the common dictionary operations like get and set can be performed. We can also modifyos.environbut any changes will be effective only for the current process where it was assigned and it will not change the value permanently.os.environ Object Syntax in PythonSyntax: os.environ Parameter: It is a non-callable object. Hence, no parameter is required Return Type: This returns a dictionary representing the users environmental variables Python os.environ Object ExamplesBelow are some examples by which we can fetch environment variables with os.environ in Python and set an environment variable using the OS module in Python:Access User Environment Variables Using os.environ ObjectIn this example, the below code uses the `os.environ` object to retrieve and print the list of user's environment variables, employing the `pprint` module to display them in a readable format. Python # importing os module import os import pprint # Get the list of user's env_var = os.environ # Print the list of user's print("User's Environment variable:") pprint.pprint(dict(env_var), width = 1) Output:{'CLUTTER_IM_MODULE': 'xim', 'COLORTERM': 'truecolor', 'DBUS_SESSION_BUS_ADDRESS': 'unix:path=/run/user/1000/bus', 'DESKTOP_SESSION': 'ubuntu', 'DISPLAY': ':0', 'GDMSESSION': 'ubuntu', 'GJS_DEBUG_OUTPUT': 'stderr', 'GJS_DEBUG_TOPICS': 'JS ' 'ERROR;JS ' 'LOG', 'GNOME_DESKTOP_SESSION_ID': 'this-is-deprecated', 'GNOME_SHELL_SESSION_MODE': 'ubuntu', 'GTK_IM_MODULE': 'ibus', 'HOME': '/home/ihritik', 'IM_CONFIG_PHASE': '2', 'JAVA_HOME': '/opt/jdk-10.0.1', 'JOURNAL_STREAM': '9:28586', 'JRE_HOME': '/opt/jdk-10.0.1/jre', 'LANG': 'en_IN', 'LANGUAGE': 'en_IN:en', 'LESSCLOSE': '/usr/bin/lesspipe ' '%s ' '%s', 'LESSOPEN': '| ' '/usr/bin/lesspipe ' '%s', 'LOGNAME': 'ihritik', 'PATH': '/usr/local/sbin:/usr/local/bin:/usr/sbin:/usr/bin:/sbin:/bin:/usr/games:/usr/local/games:/snap/bin:/usr/local/java/jdk-10.0.1/bin:/usr/local/java/jdk-10.0.1/jre/bin:/opt/jdk-10.0.1/bin:/opt/jdk-10.0.1/jre/bin', 'PWD': '/home/ihritik', 'QT4_IM_MODULE': 'xim', 'QT_IM_MODULE': 'ibus', 'SESSION_MANAGER': 'local/hritik:@/tmp/.ICE-unix/1127,

unix/hritik:/tmp/.ICE-unix/1127', 'SHELL': '/bin/bash', 'SHLVL': '2', 'SSH_AUTH_SOCK': '/run/user/1000/keyring/ssh', 'TERM': 'xterm-256color', 'TEXTDOMAIN': 'im-config', 'TEXTDOMAINDIR': '/usr/share/locale/', 'USER': 'ihritik', 'USERNAME': 'ihritik', 'VTE_VERSION': '4804', 'WAYLAND_DISPLAY': 'wayland-0', 'XDG_CONFIG_DIRS': '/etc/xdg/xdg-ubuntu:/etc/xdg', 'XDG_CURRENT_DESKTOP': 'ubuntu:GNOME', 'XDG_MENU_PREFIX': 'gnome-', 'XDG_RUNTIME_DIR': '/run/user/1000', 'XDG_SEAT': 'seat0', 'XDG_SESSION_DESKTOP': 'ubuntu', 'XDG_SESSION_ID': '2', 'XDG_SESSION_TYPE': 'wayland', 'XDG_VTNR': '2', 'XMODIFIERS': '@im=ibus', '_': '/usr/bin/python3'}Retrieving Specific Environment Variables Using os.environ ObjectIn this example, this code uses the os.environ object to retrieve and print the values of specific environment variables ('HOME' and 'JAVA_HOME'). It demonstrates accessing the values directly using square bracket notation for an existing variable ('HOME') and using the os.environ.get() method for a variable that may not exist ('JAVA_HOME'). Python # importing os module import os # 'HOME' environment variable home = os.environ['HOME'] print("HOME:", home) # 'JAVA_HOME' environment variable java_home = os.environ.get('JAVA_HOME') # 'JAVA_HOME' environment variable print("JAVA_HOME:", java_home) Output:HOME: /home/ihritikJAVA_HOME: /opt/jdk-10.0.1Set an Environment Variable Using the OS ModuleIn this example, the Python code prints the current value of the 'JAVA_HOME' environment variable using `os.environ['JAVA_HOME']`, then modifies the value of 'JAVA_HOME' to '/home/ihritik/jdk-10.0.1', and prints the modified value using the same method. Python # importing os module import os # Print the value print("JAVA_HOME:", os.environ['JAVA_HOME']) # Modify the value os.environ['JAVA_HOME'] = '/home / ihritik / jdk-10.0.1' # Print the modified value print("Modified JAVA_HOME:", os.environ['JAVA_HOME']) Output:JAVA_HOME: /opt/jdk-10.0.1Modified JAVA_HOME: /home/ihritik/jdk-10.0.1Add New Environment Variable Using os.environ ObjectIn this example, the Python code uses the `os.environ` object to add a new environment variable named 'GeeksForGeeks' with the value 'www.geeksforgeeks.org'. It then retrieves and prints the value of the added environment variable using os.environ['GeeksForGeeks']. Python # importing os module import os # Add a new environment variable os.environ['GeeksForGeeks'] = 'www.geeksforgeeks.org' # Get the value print("GeeksForGeeks:", os.environ['GeeksForGeeks']) Output:GeeksForGeeks: www.geeksforgeeks.orgAccess Environment Variable that Does Not Exists In this example, the Python code attempts to print the value of the 'MY_HOME' environment variable using os.environ['MY_HOME']. However, there is a syntax error in the code due to the missing closing parenthesis in the print statement, which would result in a SyntaxError. Python # importing os module import os # Print the value print("MY_HOME:", os.environ.get('MY_HOME', '/home'))Output:Traceback (most recent call last): File "osenviron.py", line 8, in print("MY_HOME:", os.environ['MY_HOME']) File "/usr/lib/python3.6/os.py", line 669, in __getitem__ raise KeyError(key) from NoneKeyError: 'MY_HOME'Handling Error while Access Environment Variable which Does Not ExistsIn this example, the code demonstrates two methods to access the value of the environment variable 'MY_HOME.' The first method uses `os.environ.get()` with a default message if the variable is not found, while the second method uses a try-except block to catch a KeyError if the variable is not present. Python # importing os module import os # Method 1 print("MY_HOME:", os.environ.get('MY_HOME', "Environment variable does not exist")) # Method 2 try: print("MY_HOME:", os.environ['MY_HOME']) except KeyError: print("Environment variable does not exist") Output:MY_HOME: Environment variable does not existEnvironment variable does not exist An Operating System(OS) is a software that manages and handles hardware and software resources of a computing device. Responsible for managing and controlling all the activities and sharing of computer resources among different running applications.A low-level Software that includes all the basic functions like processor management, memory management, file management, etc.It mainly acts a government for your system that has different departments to manage different resources.Examples are Linux, Unix, Windows 11, MS DOS, Android, macOS and iOS. Basics : IntroductionTypes of OSKernel in OSSystem Call System Initialization Process Scheduling : >> Quiz on CPU Scheduling Process Synchronization >> Quiz on Process Management in OS Deadlock : >> Quiz on Deadlock Multithreading: Memory Management : >> Quiz on Memory Management Disk Management: >> Quiz on Input Output Systems Misc Types of Operating System Batch OS (e.g. Transactions Process, Payroll System, etc.)Multi-programmed OS(e.g. Windows, UNIX, macOS, etc.)Timesharing OS(e.g. Multics, Linux, etc.)Real-Time OS(e.g. PSOS, VRTX, etc.)Distributed OS(e.g. LOCUS, Solaris, etc.) Operating System Functions Memory and processor ManagementNetwork ManagementSecurity ManagementFile ManagementError DetectionJob Accounting Introduction to Operating System Types of Operating System Multithreading Introduction OS - Thread vs Process Difference between User Threads and Kernel Threads Operating System Tutorial Memory Management in Operating System Fixed (or static) Partitioning in Operating System Variable (or Dynamic) Partitioning in Operating System Paging in Operating System Segmentation in Operating System Segmentation in Operating System Page Replacement Algorithms in Operating Systems File Systems in Operating System File Systems in Operating System OS module in Python provides functions for interacting with the operating system. OS comes under Python's standard utility modules. This module provides a portable way of using operating system-dependent functionality.*os* and *os.path* modules include many functions to interact with the file system.Python-OS-ModuleFunctionsHere we will discuss some important functions of the Python os module:Handling the Current Working DirectoryCreating a DirectoryListing out Files and Directories with PythonDeleting Directory or Files using PythonFile Permissions and MetadataHandling the Current Working DirectoryConsider Current Working Directory(CWD) as a folder, where Python is operating. Whenever the files are called only by their name, Python assumes that it starts in the CWD which means that name-only reference will be successful only if the file is in the Pythons CWD.Note: The folder where the Python script is running is known as the Current Directory. This is not the path where the Python script is located.Getting the Current working directoryTo get the location of the current working directory os.getcwd() is used.Example: This code uses the 'os' module to get and print the current working directory (CWD) of the Python script. It retrieves the CWD using the 'os.getcwd()' and then prints it to the console. Python import os cwd = os.getcwd() print("Current working directory:", cwd) Output:Current working directory: /home/nikhil/Desktop/gfgChanging the Current working directoryTo change the current working directory(CWD) os.chdir() method is used. This method changes the CWD to a specified path. It only takes a single argument as a new directory path.Note: The current working directory is the folder in which the Python script is operating.Example: The code checks and displays the current working directory (CWD) twice: before and after changing the directory up one level using os.chdir('../'). It provides a simple example of how to work with the current working directory in Python. Python import os def current_path(): print("Current working directory before") print(os.getcwd()) print() current_path() os.chdir('../') current_path() Output:Current working directory beforeC:\Users\Nikhil Aggarwal\Desktop\gfgCurrent working directory afterC:\Users\Nikhil Aggarwal\DesktopCreating a DirectoryThere are different methods available in the OS module for creating a directory. These are Using os.mkdir()By using os.mkdir() method in Python is used to create a directory named path with the specified numeric mode. This method raises FileExistsError if the directory to be created already exists.Example: This code creates two directories: "GeeksforGeeks" within the "D:/Pycharm projects/" directory and "Geeks" within the "D:/Pycharm projects" directory. The first directory is created using the os.mkdir() method without specifying the mode. The second directory is created using the same method, but a specific mode (0o666) is provided, which grants read and write permissions. The code then prints messages to indicate that the directories have been created. Python import osdirectory = "GeeksforGeeks"parent_dir = "D:/Pycharm projects/"path = os.path.join(parent_dir, directory) os.mkdir(path)print("Directory '% s' created" % directory)directory = "Geeks"parent_dir = "D:/Pycharm projects"mode = 0o666path = os.path.join(parent_dir, directory)os.mkdir(path, mode)print("Directory '% s' created" % directory)Output:Directory 'GeeksforGeeks' createdDirectory 'Geeks' createdUsing os.makedirs()os.makedirs() method in Python is used to create a directory recursively. That means while making leaf directory if any intermediate-level directory is missing, os.makedirs() method will create them all.Example:This code creates two directories, "Nikhil" and "c", within different parent directories. It uses the os.makedirs function to ensure that parent directories are created if they don't exist.It also sets the permissions for the "c" directory. The code prints messages to confirm the creation of these directories Python import osdirectory = "Nikhil"parent_dir = "D:/Pycharm projects/GeeksForGeeks/Authors"path = os.path.join(parent_dir, directory)os.makedirs(path)print("Directory '% s' created" % directory)directory = "c"parent_dir = "D:/Pycharm projects/GeeksforGeeks/a/b"mode = 0o666path = os.path.join(parent_dir, directory)os.makedirs(path, mode)print("Directory '% s' created" % directory)Output:Directory 'Nikhil' createdDirectory 'c' createdListing out Files and Directories with PythonThere is os.listdir() method in Python is used to get the list of all files and directories in the specified directory. If we dont specify any directory, then the list of files and directories in the current working directory will be returned.Example: This code lists all the files and directories in the root directory ("/"). It uses the os.listdir function to get the list of files and directories in the specified path and then prints the results. Python import os path = "/"dir_list = os.listdir(path) print("Files and directories in '", path, "' :") print(dir_list) Output:Files and directories in ' / ' :['sys', 'run', 'tmp', 'boot', 'mnt', 'dev', 'proc', 'var', 'bin', 'lib64', 'usr', 'lib', 'srv', 'home', 'etc', 'opt', 'sbin', 'media']Deleting Directory or Files using PythonOS module provides different methods for removing directories and files in Python. These are -Using os.remove()Using os.rmdir()os.remove() method in Python is used to remove or delete a file path. This method can not remove or delete a directory. If the specified path is a directory then OSError will be raised by the method.Example: Suppose the file contained in the folder are:This code removes a file named "file1.txt" from the specified location "D:/Pycharm/projects/GeeksforGeeks/Authors/Nikhil/". It uses the os.remove function to delete the file at the specified path. Python import os file = 'file1.txt'location = "D:/Pycharm projects/GeeksforGeeks/Authors/Nikhil/"path = os.path.join(location, file) os.remove(path) Output:Using os.rmdir()os.rmdir() method in Python is used to remove or delete an empty directory. OSError will be raised if the specified path is not an empty directory.Example: Suppose the directories areThis code attempts to remove a directory named "Geeks" located at "D:/Pycharm projects/". It uses the os.rmdir function to delete the directory. If the directory is empty, it will be removed. If it contains files or subdirectories, you may encounter an error. Python import os directory = "Geeks"parent = "D:/Pycharm projects/"path = os.path.join(parent, directory) os.rmdir(path) Output:Apart from basic file and directory operations, Pythons os module provides access to lower-level file system metadata and permission handling useful for scripting, administration, and system-level tasks.Three important methods in this category are:os.chmod(): Change file or directory permissionsos.chown(): Change file owner and group (Unix only)os.stat(): Fetch metadata like file size, modification time, permissions, etc.These functions allow you to interact with file system properties similar to what youd do with chmod, chown, and ls -l in Unix/Linux.os.chmod()os.chmod() method is used to change the permissions (read, write, execute) of a file or directory. Permissions must be passed in octal format (e.g., 0o600). Python import os # Set read-write permissions for owner (0o600)os.chmod("sample.txt", 0o600)Explanation: 0o600 means:Owner: Read & writeGroup: No permissionOthers: No permissionos.chown()os.chown() method allows you to change the owner and group ID of a file. This is typically used in system scripts and requires appropriate permissions. It is used in Unix/Linux systems only. Python import os # Change owner and group to user ID 1000 and group ID 1000os.chown("example.txt", 1000, 1000)Explanation:Both uid and gid must be integers.Requires root access to change file ownership.On unsupported platforms (e.g., Windows), it will raise AttributeError.os.stat()os.stat() method is used to retrieve metadata about a file such as its size, permissions, and timestamps. Python import os stats = os.stat("example.txt") print("Size:", stats.st_size, "bytes")print("Last modified:", stats.st_mtime)print("Permissions:", oct(stats.st_mode)[-3:])OutputSize: 48 bytesLast modified: 1720948800.0Permissions: 600Explanation:st_size: Size in bytesst_mtime: Last modified timestamp (UNIX time)st_mode: File mode bits; we extract the last 3 digits for permission infoSimilar to output of ls -l in Unix/LinuxCommonly Used FunctionsUsing os.name functionThis function gives the name of the operating system dependent module imported. The following names have currently been registered: 'posix', 'nt', 'os2', 'ce', 'java' and 'riscos'. PythonOutput:posixNote: It may give different output on different interpreters, such as 'posix' when you run the code here.Using os.error FunctionAll functions in this module raise OSError in the case of invalid or inaccessible file names and paths, or other arguments that have the correct type, but are not accepted by the operating system. os.error is an alias for built-in OSError exception.This code reads the contents of a file named 'GFG.txt'. It uses a 'try...except' block to handle potential errors, particularly the 'IOError' that may occur if there's a problem reading the file.If an error occurs, it will print a message saying, "Problem reading: GFG.txt." Python import ostry: filename = 'GFG.txt' f = open(filename, 'rU') text = f.read() f.close()except IOError: print('Problem reading: ' + filename)Output:Problem reading: GFG.txtUsing os.popen() FunctionThis method opens a pipe to or from command. The return value can be read or written depending on whether the mode is 'r' or 'w'.Syntax: os.popen(command[, mode[, bufsize]])Parameters mode & bufsize are not necessary parameters, if not provided, default 'r' is taken for mode.This code opens a file named 'GFG.txt' in write mode, writes "Hello" to it, and then reads and prints its contents. The use of os.popen is not recommended, and standard file operations are used for these tasks. Python import osfd = "GFG.txt" file = open(fd, 'w')file.write("Hello")file.close()file = open(fd, 'r')text = file.read()print(text) file = os.popen(fd, 'w')file.write("Hello")Output:HelloNote: Output for popen() will not be shown, there would be direct changes into the file.Using os.close() FunctionClose file descriptor fd. A file opened using open(), can be closed by close()only. But file opened through os.popen(), can be closed with close() or os.close(). If we try closing a file opened with open(), using os.close(), Python would throw TypeError. Python import osfd = "GFG.txt"file = open(fd, 'r')text = file.read()print(text)os.close(file)Output:Traceback (most recent call last): File "C:\Users\GFG\Desktop\GeeksForGeeksOSFile.py", line 6, in os.close(file)TypeError: an integer is required (got type _io.TextIOWrapper)Note: The same error may not be thrown, due to the non-existent file or permission privilege.Using os.rename() FunctionA file old.txt can be renamed to new.txt, using the function os.rename(). The name of the file changes only if, the file exists and the user has sufficient privilege permission to change the file. Python import osfd = "GFG.txt"os.rename(fd,'New.txt')os.rename(fd,'New.txt')Output:Traceback (most recent call last): File "C:\Users\GFG\Desktop\ModuleOS\GeeksForGeeksOSFile.py", line 3, in os.rename(fd,'New.txt')FileNotFoundError: [WinError 2] The system cannot find thefile specified: 'GFG.txt' -> 'New.txt' A file name "GFG.txt" exists, thus when os.rename() is used the first time, the file gets renamed. Upon calling the function os.rename() second time, file "New.txt" exists and not "GFG.txt" thus Python throws FileNotFoundError.Using the Os module we can remove a file in our system using the os.remove() method. To remove a file we need to pass the name of the file as a parameter. Python import os #importing os module.os.remove("file_name.txt") #removing the file.The OS module provides us a layer of abstraction between us and the operating system. When we are working with os module always specify the absolute path depending upon the operating system the code can run on any os but we need to change the path exactly. If you try to remove a file that does not exist you will get FileNotFoundError.Using os.path.exists() FunctionThis method will check whether a file exists or not by passing the name of the file as a parameter. OS module has a sub-module named PATH by using which we can perform many more functions. Python import os #importing os module result = os.path.exists("file_name") #giving the name of the file as a parameter. print(result)Output:FalseAs in the above code, the file does not exist it will give output False. If the file exists it will give us output True.Using os.path.getsize() FunctionIn os.path.getsize() function, python will give us the size of the file in bytes. To use this method we need to pass the name of the file as a parameter. Python import os #importing os modulesize = os.path.getsize("filename")print("Size of the file is", size," bytes.")Output:Size of the file is 192 bytes. The operating system can be implemented with the help of various structures. The structure of the OS depends mainly on how the various standard components of the operating system are interconnected and merge into the kernel. This article discusses a variety of operating system implementation structures and explains how and why they function.What is a System Structure for an Operating System?A system structure for an operating system is like the blueprint of how an OS is organized and how its different parts interact with each other. Because operating systems have complex structures, we want a structure that is easy to understand so that we can adapt an operating system to meet our specific needs. Similar to how we break down larger problems into smaller, more manageable subproblems, building an operating system in pieces is simpler. The operating system is a component of every segment. The strategy for integrating different operating system components within the kernel can be thought of as an operating system structure. As will be discussed below, various types of structures are used to implement operating systems.Types of Operating Systems StructuresDepending on this, we have the following structures in the operating system:Simple StructureMonolithic StructureMicro-Kernel StructureHybrid-Kernel StructureExo-Kernel StructureLayered StructureModular StructureVirtual MachinesSimple StructureSimple structure operating systems do not have well-defined structures and are small, simple, and limited. The interfaces and levels of functionality are not well separated. MS-DOS is an example of such an operating system. In MS-DOS, application programs are able to access the basic I/O routines. These types of operating systems cause the entire system to crash if one of the user programs fails.Simple OS StructureAdvantages of Simple StructureIt delivers better application performance because of the few interfaces between the application program and the hardware.It is easy for kernel developers to develop such an operating system.Disadvantages of Simple StructureThe structure is very complicated, as no clear boundaries exist between modules.It does not enforce data hiding in the operating system.Monolithic StructureA monolithic structure is a type of operating system architecture where the entire operating system is implemented as a single large process in kernel mode. Essential operating system services, such as process management, memory management, file systems, and device drivers, are combined into a single code block.Advanatages of Monolithic StructurePerformance of Monolithic structure is fast as since everything runs in a single block, therefore communication between components is quick.It is easier to build because all parts are in one code block.Disadvanatges of Monolithic ArchitectureIt is hard to maintain as a small error can affect entire system.There are also some security risks in the Monolithic architecture.Micro-Kernel StructureMicro-Kernel structure designs the operating system by removing all non-essential components from the kernel and implementing them as system and user programs. This results in a smaller kernel called the micro-kernel.Advantages of this structure are that all new services need to be added to user space and does not require the kernel to be modified. Thus it is more secure and reliable as if a service fails, then rest of the operating system remains untouched. Mac OS is an example of this type of OS.Advantages of Micro-kernel StructureIt makes the operating system portable to various platforms.As microkernels are small so these can be tested effectively.Disadvantages of Micro-kernel StructureIncreased level of inter module communication degrades system performance.Hybrid-Kernel StructureHybrid-Kernel structure is nothing but just a combination of both monolithic-kernel structure and micro-kernel structure. Basically, it combines properties of both monolithic and micro-kernel and make a more advance and helpful approach. It implement speed and design of monolithic and modularity and stability of micro-kernel structure.Advantages of Hybrid-Kernel StructureIt offers good performance as it implements the advantages of both structure in it.It supports a wide range of hardware and applications.It provides better isolation and security by implementing micro-kernel approach.It enhances overall system reliability by separating critical functions into micro-kernel for debugging and maintenance.Disadvantages of Hybrid-Kernel StructureIt increases overall complexity of system by implementing both structure (monolithic and micro) and making the system difficult to understand.The layer of communication between micro-kernel and other component increases time complexity and decreases performance compared to monolithic kernel.Exo-Kernel StructureExokernel is an operating system developed at MIT to provide application-level management of hardware resources. By separating resource management from protection, the exokernel architecture aims to enable application-specific customization. Due to its limited operability, exokernel size typically tends to be minimal.The OS will always have an impact on the functionality, performance, and scope of the apps that are developed on it because it sits in between the software and the hardware. The exokernel operating system makes an attempt to address this problem by rejecting the notion that an operating system must provide abstractions upon which to base applications. The objective is to limit developers use of abstractions as little as possible while still giving them freedom.Advantages of Exo-KernelSupport for improved application control.Separates management from security.It improves the performance of the application.A more efficient use of hardware resources is made possible by accurate resource allocation and revocation.It is simpler to test and create new operating systems.Each user-space program is allowed to use a custom memory management system.Disadvantages of Exo-KernelA decline in consistency.Exokernel interfaces have a complex architecture.Layered StructureAn OS can be broken into pieces and retain much more control over the system. In this structure, the OS is broken into a number of layers (levels). The bottom layer (layer 0) is the hardware, and the topmost layer (layer N) is the user interface. These layers are so designed that each layer uses the functions of the lower-level layers. This simplifies the debugging process, if lower-level layers are debugged and an error occurs during debugging, then the error must be on that layer only, as the lower-level layers have already been debugged.The main disadvantage of this structure is that at each layer, the data needs to be modified and passed on which adds overhead to the system. Moreover, careful planning of the layers is necessary, as a layer can use only lower-level layers. UNIX is an example of this structure.Layered OS StructureAdvantages of Layered StructureLayering makes it easier to enhance the operating system, as the implementation of a layer can be changed easily without affecting the other layers.It is very easy to perform debugging and system verification.Disadvantages of Layered StructureIn this structure, the application's performance is degraded as compared to simple structure.It requires careful planning for designing the layers, as the higher layers use the functionalities of only the lower layers.Modular StructureIt is considered as the best approach for an OS. It involves designing of a modular kernel. The kernel has only a set of core components and other services are added as dynamically loadable modules to the kernel either during runtime or boot time. It resembles layered structure due to the fact that each kernel has defined and protected interfaces, but it is more flexible than a layered structure as a module can call any other module.For example Solaris OS is organized as shown in the figure.Modular OS StructureVirtual Machines (VMs)Based on our needs, a virtual machine abstracts the hardware of our personal computer, including the CPU, disc drives, RAM, and NIC (Network Interface Card), into a variety of different execution contexts, giving us the impression that each execution environment is a different computer. An illustration of it is a virtual box.An operating system enables us to run multiple processes concurrently while making it appear as though each one is using a different processor and virtual memory by using CPU scheduling and virtual memory techniques.The fundamental issue with the virtual machine technique is disc systems. Let's say the physical machine only has three disc drives, but it needs to host seven virtual machines. The program that creates virtual machines would need a significant amount of disc space in order to provide virtual memory and spooling, so it should be clear that it is impossible to assign a disc drive to every virtual machine. The answer is to make virtual discs available.ConclusionOperating systems can be built using various structures, each affecting how components interact within the kernel. These structures include Simple/Monolithic, Micro-Kernel, Hybrid-Kernel, Exo-Kernel, Layered, Modular, and Virtual Machines. The structure chosen influences the ease of use, modification, and overall efficiency. By breaking down the OS into manageable parts, developers can create more adaptable and robust systems. Understanding these structures helps in designing operating systems that meet specific needs while maintaining functionality and performance. An operating system is an interface between the user and the hardware. It is a program that helps to utilize system hardware so it is also known as a resource manager. There are different types of operating systems based on different system architectures. , each with its own features and uses. Windows is popular for personal computers and offers a user-friendly interface. macOS, used on Apple computers, is known for its sleek design and strong performance. Linux is open-source and favored by developers for its flexibility and security. Mobile devices typically use iOS or Android, which are optimized for touchscreens and mobile apps. Each OS has unique strengths, making them suitable for various needs and preferences Let's discuss each one by one.There are different Operating Systems are present. Some of them are mentioned below:1. MS-DOSMS-DOS which is short for Microsoft Disk Operating System is a non-graphical command line operating system developed for IBM-compatible computers with x86 microprocessorsIBM-compatible. The operating system used a command line interface for the user to input commands to navigate, open, and manipulate files on their computer.FeaturesIt is a single-user operating system meaning only one user can operate at a time.It is a lightweight operating system allowing users to have direct access to the BIOS and its underlying hardware.Loads data and programs from external sources and brings them into the internal memory so they can be used on the computer.Enables the computer to perform input and output operations such as taking commands from the keyboard, and printing information on the screen.It is very helpful in making file management like creating, editing, deleting files, etc.It also controls and manages other external devices such as the printer, keyboard or external hard drive using various drive utilities.DrawbacksIt does not allow multiple users to operate on the system.It does not support a graphical interface hence mouse cannot be used to operate it.It does not support multiprogramming meaning it can only have one process in the ram.It lacked memory protection which meant no security, and less stability.It has difficulty in memory access when addressing more than 640 MB of RAM.2. Windows Operating SystemWindows is an operating system designed by Microsoft to be used on a standard x86 Intel and AMD processors. It provides an interface, known as a graphical user interface(GUI) which eliminates the need to memorize commands for the command line by using a mouse to navigate through menus, dialog boxes, buttons, tabs, and icons. The operating system was named windows since the programs are displayed in the shape of a square. This Windows operating system has been designed for both a novice user just using at home as well as for professionals who are into development.FeaturesIt is designed to run on any standard x86 Intel and AMD hence most of the hardware vendors make drivers for windows like Dell, HP, etc.It supports enhanced performance by utilizing multi-core processors.It comes preloaded with many productivity tools which helps to complete all types of everyday tasks on your computer.Windows has a very large user base so there is a much larger selection of available software programs, utilities.Windows is backward compatible meaning old programs can run on newer versions.Hardware is automatically detected eliminating need of manually installing any device drivers.DrawbacksWindows can be expensive since the OS is paid license and majority of its applications are paid products.Windows has high computer resource requirement like it should have high ram capacity, a lot of hard drive space and good graphics card.Windows slows and hangs up if the user loads up many programs at the same time.Windows includes network sharing that can be useful if user has a network with many PCs.Windows is vulnerable to virus attacks since it has a huge user base and users have to update OS to keep up-to-date with security patches.3. LINUX Operating SystemThe Linux OS is an open source operating system project that is a freely distributed, cross-platform operating system developed based on UNIX. This operating system is developed by Linus Torvalds. The name Linux comes from the Linux kernel. It is basically the system software on a computer that allows apps and users to perform some specific task on the computer. The development of Linux operating system pioneered the open source development and became the symbol of software collaboration.FeaturesLinux is free can be downloaded from the Internet or redistribute it under GNU licenses and has the best community support.Linux OS is easily portable which means it can be installed on various types of devices like mobile, tablet computers.It is a multi-user, multitasking operating system.BASH is the Linux interpreter program which can be used to execute commands.Linux provides multiple levels of file structures i.e. hierarchical structure in which all the files required by the system and those that are created by the user are arranged.Linux provides user security using authentication features and also threat detection and solution is very fast because Linux is mainly community driven.DrawbacksThere's no standard edition of Linux hence confusing for users and also becoming familiar with the Linux may be a problem for new users.More difficult to find applications to support user needs since Linux does not dominate the market.Since some applications are developed specifically for Windows and Mac, those might not be compatible with linux and sometimes users might not have much of a choice to choose between different applications like in Windows or Mac since most apps are developed for operating systems that have a huge user base.Some hardware may not be incompatible with Linux since it has patchier support for drivers which may result in malfunction.There are plenty of forums to resolve Linux issues, but it may not always match the user's own level of technical understanding.4. Solaris Operating SystemSolaris or SunOS is the name of the Sun company's Unix variant operating system that was originally developed for its family of Scalable Processor Architecture-based processors (SPARC) as well as for Intel-based processors. The UNIX workstation market had been largely dominated by this operating system during its time. As the Internet grew Sun's Solaris systems became the most widely installed servers for Web sites. Oracle purchased Sun and later renamed to Oracle Solaris.FeaturesSolaris is known for its scalability. It can handle a large workload and still delivers indisputable performance advantages for database, Web, and Java technology-based services.Solaris systems were known to their availability meaning that these operating systems hardly crashes at anytime and because of its internet networking oriented design and broad scope of features it makes the job of adding new features or fixing any problems easy.It is built for network computing as it provides optimized network stack and support for advanced network computing protocols that delivers high-performance networking to most applications.Solaris has advanced, unique security capabilities which includes some of the world's most advanced security features, such as user rights management, cryptographic Framework and secure by default networking that allows users to safely deliver new solutions.Provides tools to enable seamless interoperability, test new software and efficiently consolidate application workloads.DrawbacksSolaris is quite expensive since its an enterprise operating system. Also, Solaris doesnt provide updates for free.Solaris lacks a good graphical user interface support and is not user friendly.Hardware support is not nearly as good as many other operating systems.Performance would degrade considerably since Solaris cannot make use of different hardware that efficiently.Solaris sometimes becomes unstable and crashes due to total consumption of CPU and memory.5. Symbian Operating SystemSymbian OS was the most widely-used smartphone operating system in the world based on ARM architecture, until it was discontinued in 2014. It was developed by Symbiant Ltd, which was a partnership among PDA devices and smartphone manufacturers like Psion, Motorola, Ericsson, and Nokia. The Symbian Operating System was developed of two sub system where the first is the microkernel-based operating system with its associated libraries, and the other being interface of the OS with which the user interacts. It was explicitly developed for smartphones and hand held digital devices since this operating system consumes very low power, battery-based devices and also for ROM-based systems.FeaturesIts kernel known as EKA2 features preemptive multithreading, scheduling, memory management system and device drivers.Allows third party software to enhance the platform for better performance of the operating system.Symbian Interface is easy to use and very user friendly.Applications for Symbian are normally written in C++ or Symbian C++ using Symbian Software Development Kit (SDK).Symbian can also run applications written in Python, Java ME, Flash Lite, Ruby and .NET.Connectivity is lot easier and faster.Symbian OS has good efficiency and stability.DrawbacksResponsiveness is not smooth and sensitive as other operating systems.The Symbian OS is very vulnerable and can be easily affected by a Virus.Lack of virtual memory.6. Android Mobile Operating SystemAndroid is a Google's Linux based operating system it is designed primarily for touch screen mobile devices such as smart phones and tablet computers. The hardware which can be used to support android is based on three architectures namely ARM, Intel and MIPS design lets users manipulate the mobile devices intuitively, with finger movements that mirror common motions, such as pinching, swiping, and tapping making these applications comfortable for the users.FeaturesThe android operating system is an open source operating system means that its free and any one can use it.Android offers optimized 2D and 3D graphics, multimedia, GSM connectivity, multi-tasking.Android OS is known for its friendly user interface and exceptional customizable according to the user's taste.Huge choice of applications for its users since Playstore offer over one million apps.Software developers who want to create applications for the Android OS can download the Android Software Development Kit(SDK) to easily develop apps for android.Android would consume very little power but deliver extreme performance since its hardware is based on ARM architecture.DrawbacksThe design and coding of intuitive modern user experiences and interfaces poses a difficulty because of its dependency on Java.Most apps tend to run in the background even when closed by the user draining the battery.Performance is bound to take a hit as multiple programs run simultaneously in the background at any given time.Android phones overheat especially when indulged in hardcore productivity tasks or heavy graphics.Apps have lower security profiles and make users more susceptible to data breaches.7. iOS Mobile Operating SystemiOS which is short for iPhone OS is a mobile operating system created and developed by Apple Inc. exclusively for its hardware like A12 Bionic chip that presently powers many of its mobile devices, including the iPhone, iPad, and iPod. The iOS user interface is based upon using multi-touch gestures such as swipe, tap, pinch, and reverse pinch. The purpose of these finger actions is to provide the user with fast responsive inputs given from multiple fingers to the multi-touch capacitive screen display.FeaturesIt is written in C, C++, Objective-C and Swift and is based on the Macintosh OS X.Has excellent and intuitive user interface and very fluid response.Performance of iOS is unbeatable.iOS comes with a lot of default apps, including an email client, web browser, media player and the phone app.Availability of higher quality apps which can be downloaded from the Appstore.Apple has provided its own iOS software development kit (SDK) for the developers to create applications for Apple mobile devices.iOS is much safer than other mobile operating systems and has fewer security breaches as well.Provides regular updates and security patches.DrawbacksThe iOS is closed source instead of open source hence beta testing taking a lot of time since its only available to limited developers.The amount of memory space the iOS applications occupy is very large when compared with other mobile platforms.Lack of customization compared to other operating systems.Doesn't allow third party installations.Having intense graphics and animations consumes more power and causes battery drains.iOS is resource intensive operating system due to which older devices struggle to run it.8. FreeBSDFreeBSD is a free and open-source operating system based on the Unix operating system. It is widely used as a server operating system, but it can also be used as a desktop operating system.Free BSDFeaturesRobust and stable: FreeBSD is known for its robust and stable nature, making it a popular choice for servers and other mission-critical applications.Performance: FreeBSD is designed for performance, and its use of advanced memory management and process scheduling algorithms results in fast and efficient system performance.Security: FreeBSD has a strong focus on security, and includes many features to help secure a system, such as security-enhanced kernels, mandatory access controls, and secure levels.Flexibility: FreeBSD is highly flexible and customizable, allowing users to configure the operating system to meet their specific needs.Large repository of software: FreeBSD has a large repository of software packages available, making it easy for users to install and use the software they need.Networking: FreeBSD is designed for networking, and includes a range of advanced networking features, such as support for IPv6, TCP/IP, and virtual networking.BSD license: FreeBSD is released under the BSD license, which allows users to freely use, modify, and redistribute the operating system as they see fit.DrawbacksLike any operating system, FreeBSD has some drawbacks and limitations. Some of the most notable ones are:Limited hardware support: While FreeBSD has a wide range of hardware support, it may not support some of the latest or most cutting-edge hardware devices, especially when compared to other operating systems like Windows or macOS.Complexity: FreeBSD is known for its advanced features and technical capabilities, but this can also make it a more complex operating system to use and set up compared to other options, especially for less experienced users.Smaller community: While the FreeBSD community is dedicated and active, it is smaller than some of the other open-source communities, such as those around Linux or Ubuntu. This can make finding help or support more challenging for some users.Overall, FreeBSD is a powerful and versatile operating system that is well-suited for use as a server or desktop operating system, particularly for users who value stability, performance, and security.9. Chrome OSChrome OS is a Linux-based operating system developed by Google. It is primarily designed for use on Chromebook laptops, although it can also be run on other devices such as desktop computers and tablets. Chrome OS is designed to be simple and fast, with a focus on web-based applications and cloud-based storage.Chrome OSFeaturesSpeed and simplicity: Chrome OS is designed to start up quickly and be easy to use, with a simple and clean user interface.Web-based apps: The majority of applications used on Chrome OS are web-based, accessed through the Chrome web browser.Integration with Google services: Chrome OS integrates seamlessly with Google services like Gmail, Google Drive, and Google Calendar.Automatic updates: Chrome OS is designed to receive automatic updates, ensuring that the operating system and its security features are always up-to-date.Security: Chrome OS is known for its strong security features, including automatic updates, built-in virus protection, and secure boot.DrawbacksLimited offline functionality: Chrome OS relies heavily on an internet connection, and many of its applications and features may not work properly without an internet connection.Limited software options: While the selection of web-based apps available for Chrome OS is growing, it is still limited compared to other operating systems like Windows or macOS.Limited hardware compatibility: Chrome OS is designed to work best on Chromebooks, and may not work as well on other devices, especially those with limited hardware specifications.Dependence on Google services: If you use Chrome OS, you are heavily dependent on Google services, which may not be suitable for everyone.These are some of the features and drawbacks of Chrome OS. While it may not be the best operating system for everyone, it is well-suited for users who prioritize speed, simplicity, and security and rely primarily on web-based applications.10. Mac OSmacOS is a proprietary, Unix-based operating system developed by Apple Inc. It is the primary operating system for Apple's Mac computers and laptops. It was first introduced in 2001 as Mac OS X, and was later renamed to macOS in 2016.Mac OSFeaturesUser-friendly interface: macOS has a clean and intuitive graphical user interface, making it easy to use for both beginners and advanced users.Integration with Apple's ecosystem: macOS integrates well with Apple's other products, such as the iPhone, iPad, and Apple Watch, allowing for seamless connectivity and data transfer between devices.Built-in applications: macOS comes with a range of built-in applications, such as iMessage, FaceTime, and Safari, making it possible to perform a wide range of tasks without having to install additional software.High-quality software: Apple is known for its focus on quality, and macOS is no exception. The operating system includes high-quality applications and is known for its stability and reliability.DrawbacksPrice: macOS is a proprietary operating system, and Mac computers are typically more expensive than other types of computers.Limited hardware compatibility: Because macOS is only available on Apple's Mac computers and laptops, users are limited in terms of hardware compatibility, especially when compared to other operating systems like Windows or Linux.Proprietary software: Many of the applications and software available for macOS are proprietary and only available from Apple's App Store, which can limit user choice and flexibility.Lack of customization: Unlike other operating systems, such as Linux, macOS has limited customization options, making it less flexible for advanced users who want to change the look and feel of their operating system.In conclusion, macOS is a high-quality operating system with a user-friendly interface and tight integration with Apple's ecosystem. However, it is more expensive and less customizable than other operating systems, and is limited to Apple's hardware.Applications And Their Suitable Operating Systems1. Database and Web Server Management: The best suitable operating system for database and web server management is SOLARIS, is Unix Operating system, which itself is designed for enterprise web servers where robust applications and database is deployed where throughput is very high and needs the server 24x7 up and less down time.Solaris has Zettabyte File System(ZFS) which is a revolutionary file and logical volume manager that control the storage and retrieval of data which is not found in any other operating system.Solaris is provided with a Service Management Facility (SMF) that is responsible for management of system and application services that improves the availability of a system by ensuring that essential system and application services function normally without any troubles even in the case of hardware or software failures which is also known as Oracle Solaris Predictive Self-Healing capability.Solaris provides awesome observability utilities which can be used to find performance bugs, functional bugs and kernel bugs.Solaris scales well on multi-socket multicore systems with large number of cores due to Solaris' broader CPU support but also to the architecture of the OS itself. Solaris you can mix and match vertical and horizontal scalability whereas most of the other operating systems only provide vertical scaling.Solaris has always provided strict security providing additional barriers and immune systems to prevent any data breaches to their servers.Hence since Solaris is reliable, very stable, extremely fast, and secure operating system comparing to other OS which have raw performance and security problems it is the ideal operating system for any database management systems like SQL and web service servers .2. Cluster Computing: Clustering is a technique where multiple computers, storage devices and redundant interconnections are used to create a single highly available system. Each computer in it is a node. The best preferred operating system for cluster computing is LINUX which is a UNIX based open source freely distributed operating system which offers many robust network features.Since Linux operating system is highly customizable, which plays an important role in resource management and performance optimization which makes it most convenient for building or setting up large, multi-processor clusters while such modifications are very limited or in some cases restricted in other operating systems like windows.Management of storage access is often troublesome when multiple computers are organized to form a cluster, this where Linux's clustered file system may be come in handy to organize data storage and access across all of the cluster nodes connected.Linux clustered file systems will employ utilities to actually power down the faulty node.Linux is scalable hence making it easy to add resources to the cluster.Linux has the added advantage of availability of most scientific and engineering software and compilers readily through package managers and they can be easily configured and compiled without hassle.Since clustering applications are such that they require an OS with utmost portability, performance, scalability and serious computational power, Linux distributions like Redhat or CentOS are best suited for these computer cluster like Beowulf cluster.3. Productivity and Daily Tasks: The best suitable operating system for productivity is WINDOWS because it is intuitive, cohesive, functional and very user friendly. Windows offers best selection of software and can run on widest variety of hardware that the user has.Windows operating systems offer a suite of built-in utilities and applications like text processors, e-mail clients, calendars, calculators, image and video viewers, editors, snapshot tools, voice recorders, and web browsers, and apps for maps, cameras, news, weather, and contacts and many more can be installed from the internet if the user desires.Windows provides a suite of productivity applications under Office suite like Word, Presentations, Spreadsheets etc.Microsofts cloud storage platform OneDrive which helps the users to backup files and use them in any device anywhere.Windows provides a huge range of interface customization options.Windows 10 had full touch screen support, and it offers a Tablet mode that lets users to switch between using the pen and keyboard for inputs to increase productivity.Windows even provides a voice assistant that is capable of tasks like opening apps and web pages, informing the weather, etc.Windows OS makes it easier to arrange windows on the screen and lets users to create multiple virtual desktops for multi-tasking applications.Hence for creative, productive and daily tasks of users like creating presentations, editing images, surfing the internet, etc Windows is the best operating system.ConclusionIn conclusion, the variety of operating systems available caters to different user needs and preferences, from personal computing and professional workstations to mobile devices and specialized applications. Understanding the unique features and advantages of each OS can help users choose the best one for their requirements, ensuring an optimal and efficient computing experience. An operating system acts as an intermediary between the user of a computer and computer hardware. In short its an interface between computer hardware and user. The purpose of an operating system is to provide an environment in which a user can execute programs conveniently and efficiently. An operating system is software that manages computer hardware and software. The hardware must provide appropriate mechanisms to ensure the correct operation of the computer system and to prevent user programs from interfering with the proper operation of the system.The operating system (OS) is a program that runs at all times on a computer. All other programs, including application programs, run on top of the operating system.It does assignment of resources like memory, processors and input / output devices to different processes that need the resources. The assignment of resources has to be fair and secure. Functionalities of Operating System Resource Management: When multiple processes run on the system and need different resources like memory, input/output devices, the OS works as Resource Manager, Its responsibility is to provide hardware to the user. It decreases the load in the system. Process Management: It includes various tasks like scheduling and synchronization of processes. Process scheduling is done with the help of CPU Scheduling algorithms . Process Synchronization is mainly required because processes need to communicate with each other. When processes communicate different problems arise like two processes can update the same memory location in incorrect order. Storage Management: The file system mechanism used for the management of the secondary storage like Hard Disk. NIFS, CIFS, CFS, NFS, etc. are some file systems that are used by operating systems to manage the storage. All the data is stored in various tracks of Hard disks that are all managed by the storage manager. Memory Management: Refers to the management of primary memory, mainly allocation and de-allocation of memory to processes. The operating system has to keep track of how much memory has been used and by which process. It has to decide which process needs memory space and how much. Security/Privacy Management: Privacy is also provided by the Operating system using passwords so that unauthorized applications can't access programs or data. For example, Windows uses Kerberos authentication to prevent unauthorized access to data. Operating System as a User Interface: User (Interacts with the system) System and application programs (Software tools for the user) Operating system (Manages hardware and software resources) Hardware (Physical computing devices)Every general-purpose computer consists of hardware, an operating system(s), system programs and application programs. The hardware consists of memory, CPU, ALU, I/O devices, peripheral devices and storage devices. The system program consists of compilers, loaders, editors, OS, etc. Conceptual View of Computer System Every computer must have an operating system to run other programs. The operating system coordinates the use of the hardware among the various system programs and application programs for various users. It simply provides an environment within which other programs can do useful work. An OS is a package of some programs that runs on a computer machine, allowing it to perform efficiently. It manages the simple tasks of recognizing input from the keyboard, managing files and directories on disk, displaying output on the screen and controlling peripheral devices. Goals of Operating System Primary GoalsThe primary goals of an operating system (OS) are to provide a easy to use and convenient environment for executing user programs. User Convenience : It should be easy to use, providing a user-friendly interface and making it simple to interact with the system.Program Execution: It facilitates the execution of user programs, providing the necessary environment and services for them to run.Resource Management: The OS manages and allocates the computer's resources, including the CPU, memory, disk storage, and input/output devices, to ensure fair utilization.Security: The OS protects the system and user data from unauthorized access, ensuring the confidentiality, integrity, and availability of information.Efficient Resource Utilization: It should aim to maximize the performance and utilization of computer resources like CPU, Memory and IO devices, ensuring that the system runs smoothly and efficiently. Reliability: It should be robust and reliable, able to handle errors and exceptions gracefully, ensuring that the system continues to operate smoothly. It should be modular in design and easy to debug.List of Common Operating Systems There are multiple types of operating systems each having its own unique features: Windows OS Developer : Microsoft Key Features : User-friendly interface, software compatibility, hardware support, Strong gaming support. Advantages : Easy to use for most users, Broad support from third-party applications ,Frequent updates and support. Typical Use Cases : Personal computing, Business environment, Gaming. macOS Developer : Apple. Key Features : Sleek, intuitive user interface, Strong integration with other Apple products, Robust security features, High performance and stability. Advantages : Optimized for Apple hardware, Seamless experience across Apple ecosystem, Superior graphics and multimedia capabilities. Typical Use Cases : Creative industries (design, video editing, music production), Personal computing, Professional environments. Linux Developer : Community-driven (various distributions). Key Features : Open-source and highly customizable, Robust security and stability, Lightweight and can run on older hardware, Large selection of distributions (e.g., Ubuntu, Fedora, Debian). Advantages : Free to use and distribute, Strong community support, Suitable for servers and development environments. Typical Use Cases : Servers and data centers, Development and programming, Personal computing for tech enthusiasts. Unix Developer: Originally AT&T Bell Labs, various commercial and open-source versions available Key Features: Multiuser and multitasking capabilities, Strong security and stability, Powerful command-line interface, Portability across different hardware platforms Advantages: Reliable and robust performance, Suitable for high-performance computing and servers, Extensive support for networking Typical Use Cases: Servers and workstations, Development environments, Research and academic settings History of Operating System The operating system has been evolving through the years. The following table shows the history of OS. EraKey DevelopmentsExamples1956The first Operating System was GM-NAA I/O in 1956 by General Motors. GM-NAA I/O (1956)1960sIBM developed a time sharing system TSS/360OS/360, DOS/360 and TSS/3601970sUnix popularized simplicity and multitasking; rise of personal computers with basic OSs.Unix (1971), CP/M (1974)1980sGUI-based OSs gained traction; networking features became standard.Apple Macintosh (1984), Windows (1985)1990sOpen-source Linux emerged; GUIs in Windows and Mac OS improved.Linux (1991), Windows 95 (1995)2000s-PresentMobile OSs dominated; cloud and virtualization technologies advanced computing.iOS (2007), Android (2008)Characteristics of Operating Systems Let us now discuss some of the important characteristic features of operating systems: Device Management: The operating system keeps track of all the devices. So, it is also called the Input/Output controller that decides which process gets the device, when and for how much time. File Management: It allocates and de-allocates the resources and also decides who gets the resource. Job Accounting: It keeps track of time and resources used by various jobs or users. Error-detecting Aids: These contain methods that include the production of dumps, traces, error messages and other debugging and error-detecting methods. Memory Management: It is responsible for managing the primary memory of a computer, including what part of it are in use by whom also check how much amount free or used and allocate process Processor Management: It allocates the processor to a process and then de-allocates the processor when it is no longer required or the job is done. Security: It prevents unauthorized access to programs and data using passwords or some kind of protection technique. Layered Design of Operating System Fig. Layered OS The extended machine provides operations like context save, dispatching, swapping and I/O initiation. The operating system layer is located on top of the extended machine layer. This arrangement considerably simplifies the coding and testing of OS modules by separating the algorithm of a function from the implementation of its primitive operations. It is now easier to test, debug and modify an OS module than in a monolithic OS. We say that the lower layer provides an abstraction that is the extended machine. We call the operating system layer the top layer of the OS. Components of an Operating Systems There are two basic components of an Operating System. Shell is the outermost layer of the Operating System and it handles the interaction with the user. It interprets input for the OS and handles the output from the OS. Kernel is the core component of operating system. The kernel is the primary interface between the Operating system and Hardware. Recommended Next Reads An operating system (OS) is software that manages computer hardware and software resources. It acts as a bridge between users and the computer, ensuring smooth operation. Different types of OS serve different needs some handle one task at a time, while others manage multiple users or real-time processes.Types of Operating Systems1. Batch Operating SystemA Batch Operating System is designed to handle large groups of similar jobs efficiently. It does not interact with the computer directly but instead processes jobs that are grouped by an operator. These jobs are queued and executed one after the other, without user interaction during the process.Batch Operating SystemAdvantages of Batch Operating SystemEfficient Job Management: Multiple users can efficiently share the system, making it cost-effective.Minimal Idle Time: The system minimizes idle time by processing jobs in a continuous sequence without human intervention.Handling Repetitive Tasks: Ideal for managing large, repetitive tasks, such as payroll and billing, with minimal effort.Improved Throughput: Batch systems can handle high volumes of jobs at once, improving overall system throughput.Disadvantages of Batch Operating SystemInefficient CPU Utilization: When a job is waiting for input/output (I/O), the CPU remains idle, leading to poor utilization of resources.Unpredictable Job Completion: If one job fails, others may be delayed indefinitely, making job completion time unpredictable.Increased Response Time: The time between job submission and output can be high as all jobs are processed sequentially.Lack of Real-Time Feedback: Users cannot interact with the system in real-time, making it less suitable for interactive tasks.Examples:Payroll Systems Bank Statements2. Multi-Programming Operating SystemIn a Multi-Programming Operating System, multiple programs run in memory at the same time. The CPU switches between programs, utilizing its resources more effectively and improving overall system performance.Multi ProgrammingAdvantages of Multi-Programming Operating SystemCPU is better utilized and the overall performance of the system improves.It helps in reducing the response time.3. Multi-tasking/Time-sharing Operating systemsMultitasking OS is a type of Multiprogramming system with every process running in round robin manner. Each task is given some time to execute so that all the tasks work smoothly. Each user gets the time of the CPU as they use a single system. These systems are also known as Multitasking Systems. The task can be from a single user or different users. The time that each task gets to execute is called quantum. After this time interval is over, the OS switches over to the next task.Advantages of Time-Sharing OSEach task gets an equal opportunity.Fewer chances of duplication of software.CPU idle time can be reduced.Resource Sharing: Time-sharing systems allow multiple users to share hardware resources such as the CPU, memory and peripherals, reducing the cost of hardware and increasing efficiency.Improved Productivity: Time-sharing allows users to work concurrently, thereby reducing the waiting time for their turn to use the computer. This increased productivity translates to more work getting done in less time.Improved User Experience: Time-sharing provides an interactive environment that allows users to communicate with the computer in real time, providing a better user experience than batch processing.Disadvantages of Time-Sharing OSReliability problem.One must take care of the security and integrity of user programs and data.Data communication problem.High Overhead: Time-sharing systems have a higher overhead than other operating systems due to the need for scheduling, context switching and other overheads that come with supporting multiple users.Complexity: Time-sharing systems are complex and require advanced software to manage multiple users simultaneously. This complexity increases the chance of bugs and errors.Security Risks: With multiple users sharing resources, the risk of security breaches increases. Time-sharing systems require careful management of user access, authentication and authorization to ensure the security of data and software.Examples:IBM VM/CMSTSO (Time Sharing OptionWindows Terminal Services4. Multi-Processing Operating SystemA Multi-Processing Operating System is a type of Operating System in which more than one CPU is used for the execution of resources. It betters the throughput of the System.Multiprocessing Operating SystemAdvantages of aMulti-ProcessingOperating SystemIt increases the throughput of the system as processes can be parallelized.As it has several processors, so, if one processor fails, we can proceed with another processor.5. Distributed Operating SystemDistributed operating systems are a recent advancement in the world of computer technology and are being widely accepted all over the world and, that too, at a great pace. Various autonomous interconnected computers communicate with each other using a shared communication network. Independent systems possess their own memory unit and CPU.Systems. These systems' processors differ in size and function. The major benefit of working with these types of operating systems is that it is always possible that one user can access the files or software which are not present on his system but on some other system connected within this network, i.e., remote access is enabled within the devices connected to that network.Distributed OSAdvantages of Distributed Operating SystemFailure of one will not affect the other network communication, as all systems are independent of each other.Electronic mail increases the data exchange speed.Since resources are being shared, computation is highly fast and durable.Load on host computer reduces.These systems are easily scalable as many systems can be easily added to the network.Delay in data processing reduces.Disadvantages of Distributed Operating SystemFailure of the main network will stop the entire communication.To establish distributed systems, the language is not yet well-defined.These types of systems are not readily available as they are very expensive. Not only that the underlying software is highly complex and not understood well yet.Issues With Distributed Operating SystemNetworking causes delays in the transfer of data between nodes of a distributed system. Such delays may lead to an inconsistent view of data located in different nodes and make it difficult to know the chronological order in which events occurred in the system.Control functions like scheduling, resource allocation and deadlock detection have to be performed in several nodes to achieve computation speedup and provide reliable operation when computers or networking components fail.Messages exchanged by processes present in different nodes may travel over public networks and pass through computer systems that are not controlled by the distributed operating system. An intruder may exploit this feature to tamper with messages or create fake messages to fool the authentication procedure and masquerade as a user of the system.Examples:LOCUSMICROSAmoeba6. Network Operating SystemThese systems run on a server and provide the capability to manage data, users, groups, security, applications and other networking functions. These types of operating systems allow shared access to files, printers, security, applications and other networking functions over a small private network. One more important aspect of Network Operating Systems is that all the users are well aware of the underlying configuration, of all other users within the network, their connections, etc. and thats why these computers are popularly known atightly coupled systems.Network Operating SystemAdvantages of Network Operating SystemHighly stable, centralized servers.Security concerns are handled through servers.New technologies and hardware upgrades are easily integrated into the system.Server access is possible remotely from different locations and types of systems.Disadvantages of Network Operating SystemServers are costly.The user has to depend on a central location for most operations.Maintenance and updates are required regularly.Examples:Microsoft Windows Server 2003Microsoft Windows Server 2008UNIX, LinuxMac OS XNovell NetWare7. Real-Time Operating SystemThese types of OSs serve real-time systems. The time interval required to process and respond to inputs is very small. This time interval is calledresponse time. Real-time systemsare used when there are time requirements that are very strict like missile systems, air traffic control systems, robots, etc.Types of Real-Time Operating SystemsHard Real-Time Systems:Hard Real-Time OSs are meant for applications where time constraints are very strict and even the shortest possible delay is not acceptable. These systems are built for saving lives like automatic parachutes or airbags which are required to be readily available in case of an accident. Virtual memory is rarely found in these systems.Soft Real-Time Systems:These OSs are for applications where time is less strict.Difference Between Hard Real-Time OS and Soft Real-Time OS.Real-Time Operating SystemAdvantages of RTOSMaximum Consumption:Maximum utilization of devices and systems, thus more output from all the resources.Task Shifting:The time assigned for shifting tasks in these systems is very less. For example, in older systems, it takes about 10 microseconds to shift from one task to another and in the latest systems, it takes 3 microseconds.Focus on Application:Focus on running applications and give less importance to applications that are in the queue.Real-timeoperating system intheembedded system:Since the size of programs is small, RTOS can also be used in embedded systems like in transport and others.Error-Free,:These types of systems are error-free.Memory Allocation:Memory allocation is best managed in these types of systems.Disadvantages of RTOSLimited Tasks:Very few tasks run at the same time and their concentration is very less on a few applications to avoid errors.Use heavy system resources:Sometimes the system resources are not so good and they are expensive as well.Complex Algorithms:The algorithms are very complex and difficult for the designer to write.Device driver and interrupt signals:It needs specific device drivers and interrupt signals to respond earliest to interrupts.Thread Priority:It is not good to set thread priority, as these systems are much less prone to switching tasks.Examples:Scientific experimentsMedical imaging systemsIndustrial control systemsWeapon systemsRobotsAir traffic control systems8. Mobile Operating SystemsMobile operating systems are designed specifically for mobile devices such as smartphones and tablets. Examples of such operating systems are Android and iOS. These operating systems manage the hardware and software resources of the device, providing a platform for running applications and ensuring a seamless user experience.Advantagesof Mobile Operating SystemsUser-Friendly Interfaces:

Mobile operating systems are designed to be intuitive and easy to use, making them accessible to a wide range of users.Extensive App Ecosystems: The availability of a vast number of applications allows users to customize their devices to meet their specific needs.Connectivity Options: Mobile operating systems support multiple connectivity options, enabling users to stay connected wherever they go.Regular Updates: Mobile operating systems receive regular updates, including new features, security patches and performance improvements.DisadvantagesMobile Operating SystemsBattery Life Constraints: Despite advancements in power management, battery life remains a challenge for mobile devices, especially with heavy usage.Security Risks: Mobile devices are susceptible to various security threats, such as malware and phishing attacks, which can compromise user data.Fragmentation: In the case of Android, the wide range of devices and customizations can lead to fragmentation, making it difficult for developers to ensure compatibility across all devices.Limited Hardware Resources: Mobile devices have limited processing power, memory and storage compared to desktop computers, which can affect the performance of resource-intensive applications.

Can os x el capitan version 10.11.6 be upgraded. Os x el capitan 10.11.6 upgrade. Os x el capitan 10.11.6 upgrade to sierra. Os x el capitan versie 10.11.6. Os x el capitan version 10.11.6 how to get old versions of mac os.