AIMERZ.ai
Aim.Act.Achieve

OPERATING SYSTEM

INTERVIEW

Questions & Answers

Basics of Operating System

1. What do you understand about operating systems ? List down any majorly used OS.

Operating System is a kind of software which is used to manage computer hardware, schedule processes for all applications to run efficiently and smoothly. It involves various activities, Scheduling Process, file management, device management, e.t.c. There are various types of operating systems. The mainly used OS are Linux, Windows , Android and IOS.

2.What is a file System ?

File system is a format of the file made available inside the operating system based on compatibility of the type of Hardware. It provides a software layer that manages files and folders on a storage device.

3.Differentiate between various file systems?

There are various types of file systems, FAT, NTFS, and ext4 file systems are mainly used amongst them.

FAT : (File Allocation Table) Here In this file system the files are allocated in the form of a table and this file system has only a few limited features.

NTFS: (New Technology File System) It is also a similar file system but comes with more security and stability

ext4 : This file system is different from FAT and NTFS and is used for Linux Systems.

4.What do you understand about Kernel in Operating Systems?

A Kernel is a core component of Operating System which directly deals with system hardware and manages system resources.

5.What do you understand about the process?

A process is an active entity that the OS manages to execute a program. It consists of program code and its current execution context, along with metadata necessary for the OS.

6.What do you understand about a PCB?

A process control block is a kind of data structure which holds information about its state , program counter, CPU registers, and memory information e.t.c.

7.What is the difference between a process and a thread?

A Processes is an independent instance of running programs in their own memory space, a thread is a lighter instance or pieces of a process with a shared memory space. We can execute multiple threads concurrently with this shared memory space.

8.Can you mention any few CPU Scheduling Algorithms?

First-Come, First-Served (FCFS): Processes are executed in the order they arrive.

Shortest Job First (SJF): Processes with the shortest execution time are scheduled next.

Priority Scheduling: Processes are scheduled on the basis of priority as provided.

Round Robin (RR): Each process gets a fixed time slot to execute, with circular wait.

Shortest Remaining Time First (SRTF): It is the preemptive version of the SJF, here the process with least remaining time is given priority for the execution.

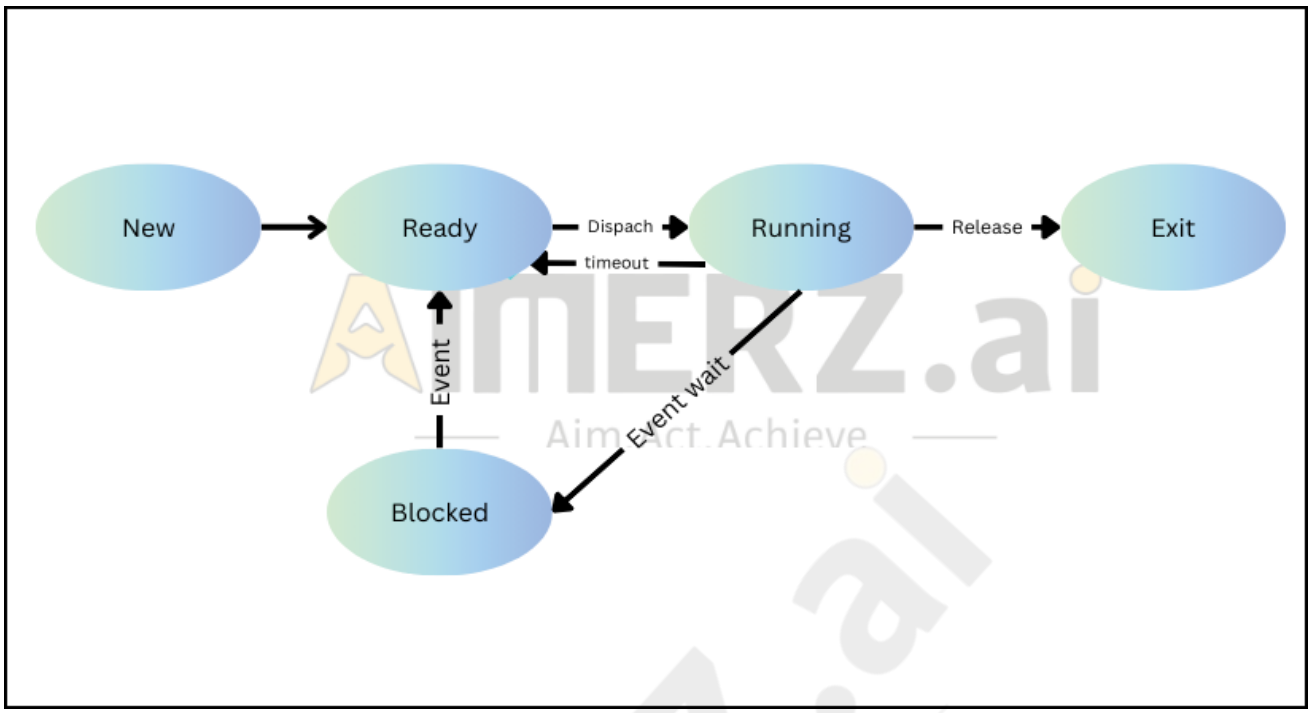
9.What is the benefit of multiprogramming ?

Multiprogramming ensures that the CPU utilization is maximum and the Idle CPU Cycles are avoided. executing a process, improving system efficiency and throughput.

10.What are the various states of a process?

Generally we have five stages of Process execution.

1. **New:** THe every new born process comes under this stage . The Program Control Block is already made but the program is yet to be loaded in the main memory. The program remains in the new state until the scheduler moves the process to the ready state as per the requirement .
2. **Ready:** A process that is waiting for its execution.
3. **Running:** The currently executing process.
4. **Waiting/Blocked:** Process waiting for some event such as completion of I/O operation,bounded wait, synchronization, etc.
5. **Terminated/Exit:** A process that is finished or aborted due to some reason.



11. What do you understand about Ready Queue in Process Management?

Ready queue is maintained in primary memory. The short term scheduler picks the job from the ready queue and dispatches it to the CPU for the execution.

12. How can you differentiate between a monolithic kernel and microkernel ?

Monolithic Kernel	Micro Kernel
(a) Size is larger	(a) Size is smaller
(b) Considerably faster due to direct service handling in kernel space.	(b) slower due to user-space communication overhead.
(c) Less Reliable as a single bug may lead to failure of the whole system.	(c) More reliable due to separate isolated services.

(d) Very difficult to modify and extend due to tightly coupled services.	(d) It is easier to alter or update any of the services without affecting others.
(e) All OS services run in kernel space	(e) Some Minimal Core services in Kernel Spaces and others in User Space.

13. What are the benefits of a multiprocessor system?

A Multiprocessor system is a type of processor which employs use of more than one CPU at the same time to improve efficiency. It involves the processing of different computer programs at the same time mostly by a computer system with two or more CPUs that are sharing single memory.

14. What is GUI?

GUI (Graphical User Interface) is a type of user friendly interface that allows users to interact with OS. GUI is created to make the OS more convenient to all the user base or people who use the operating system may understand rather than using a command-line interface .

15. What is the use of Pipe in OS ?

Pipe is a kind of tool used to do inter process communication. Mainly pipe is used for inter process message passing between processes or in IPC (Inter Process Communication).

16. What do you understand by multiprogramming in the OS ?

The ability to run multiple processes at the same time on a single Processor is known as multiprogramming. It simply keeps the CPU busy in tasks with at least one process in running state to minimize the idle CPU time.

17. What do you mean by virtual memory ?

It is a memory management technique feature of the OS that creates the illusion to users of a very large (main) memory. It is simply a space where a greater number of programs can be stored by themselves in the form of pages. It enables us to increase the use of physical memory by using a disk and also allows us to have memory protection.

18. What do you understand about the critical section ?

Critical Section is known as a piece of a program which tries to access shared resources. The resource may be any resource in any computer memory, Data structure, or CPU or any IO device.

Multiple processes cannot execute the vital portion simultaneously; the operating system has challenges when it comes to permitting and prohibiting processes from accessing the critical section.

To create a set of protocols that can guarantee that the race condition between the processes won't ever occur, the critical section problem is utilized.

Our primary goal is to resolve the critical section problem so that the cooperative processes can be synchronized. In order to satisfy the following requirements, our solution must be provided.

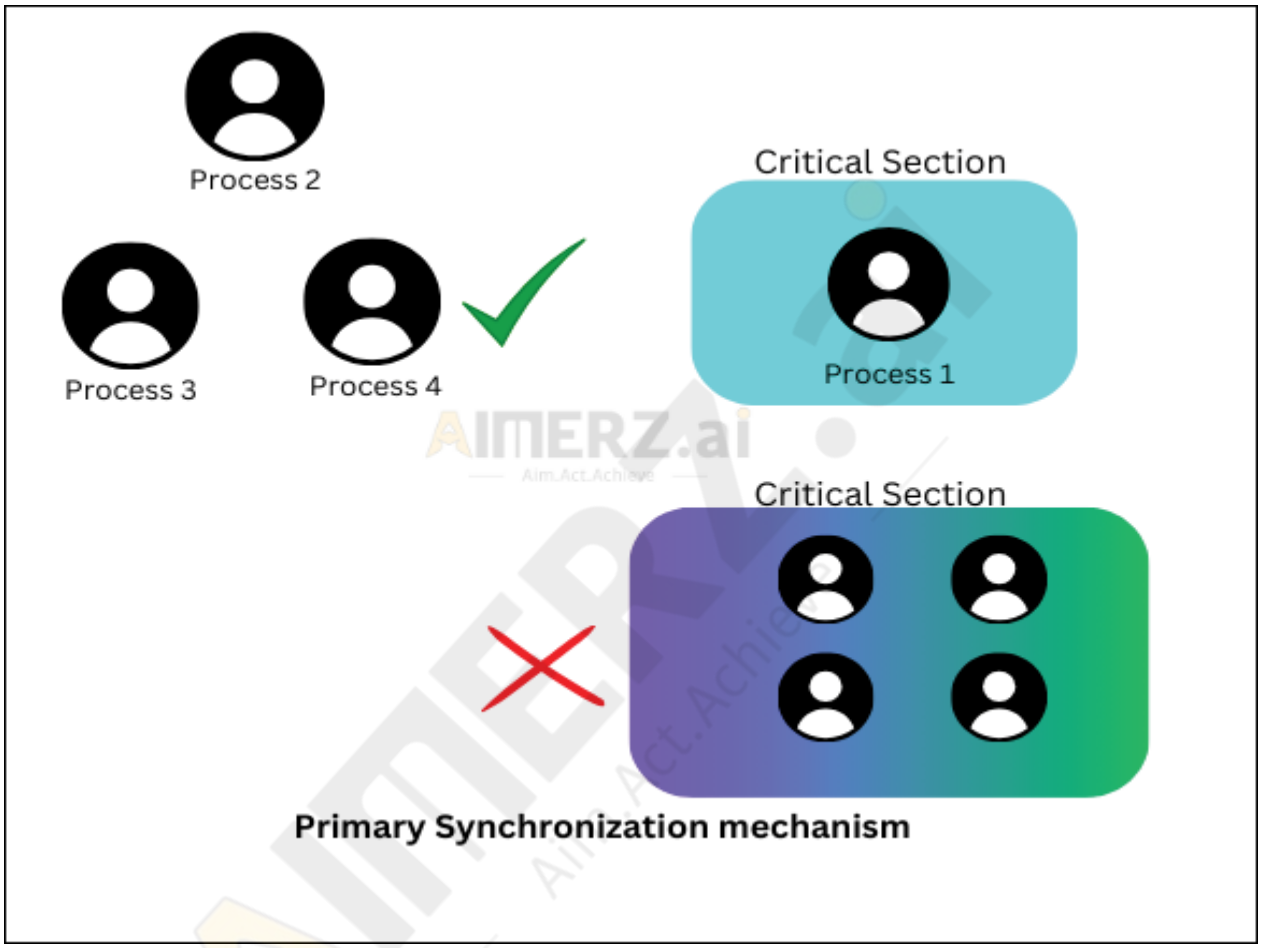
- **Primary**

    1. **Mutual Exclusion**

        Our solution must provide mutual exclusion. By Mutual Exclusion, we mean that if one process is executing inside the critical section then the other process must not enter in the critical section.

2. **Progress**

   Progress means that if one process doesn't need to execute into the critical section then it should not stop other processes from getting into the critical section.



- ## Secondary

1. **Bounded Waiting**

   a. The amount of time each process will take to reach the crucial stage should be predictable. The procedure shouldn't wait interminably to reach the crucial part.

2. **Architectural Neutrality**

   a. Our mechanism must be architecturally natural. It means that if our solution is working fine on one architecture then it should also run on the other ones as well.

19. Difference between 32 bit and 64 bit Operating System.

- A 32-bit system can access 2^{32} memory addresses, i.e., 4 GB of RAM or physical memory; ideally, it can also access more than 4 GB of RAM.

- A 64-bit system can access 2^{64} memory addresses, i.e., actually 18-Quintillion bytes of RAM. In short, any amount of memory greater than 4 GB can be easily handled by it.

20. What do you mean by demand paging ?

Demand paging is a process in which we load the page whenever it is on demand, i.e. the page fault that has a fault occurred is generally known as demand paging.

21. Explain RAID Levels?

RAID (Redundant Array of Independent Disks) is a technology in operating systems that combines multiple physical hard drives into a single logical unit for data redundancy, performance improvement, or both. There are different RAID levels, each offering a different balance of these benefits:

1. **RAID 0 (Striping):** Distributes data across multiple disks to improve performance. No redundancy is provided, so if one disk fails, all data is lost.
2. **RAID 1 (Mirroring):** Duplicates the same data on two or more disks. This provides redundancy, as the data is still accessible if one disk fails, but it doesn't improve performance.
3. **RAID 5 (Striping with Parity):** Distributes data and parity (error-checking) information across three or more disks. It offers a balance between performance, redundancy, and storage efficiency. If one disk fails, data can be reconstructed from the parity information.
4. **RAID 6 (Striping with Double Parity):** Similar to RAID 5, but with an additional parity block, allowing for the failure of two disks without data loss.

5. **RAID 10 (Combination of RAID 1 and 0):** Combines mirroring and striping. Data is mirrored for redundancy and striped for performance, requiring at least four disks.

RAID is commonly used in servers and storage systems to enhance reliability and performance.

22. What is fragmentation?

Fragmentation in operating systems refers to inefficient use of memory or storage due to divided or scattered free space.

- **Internal Fragmentation**: Wasted space within allocated memory blocks because the allocated block is larger than needed.
- **External Fragmentation**: Free memory is split into small, non-contiguous chunks, making it hard to allocate large blocks of memory even if the total free space is sufficient.

23. What is multitasking?

Multitasking is a type of multiprogramming extension . In multitasking, more than one task is made possible to run with shared hardware. In this technique, the multiple forms of tasks, that are available in the form of process, share common computing resources such as a CPU.

24. What is the difference between DOS and UNIX?

DOS is a basic, single-tasking operating system with a command-line interface, primarily used in early PCs. UNIX is a powerful, multitasking, multiuser operating system designed for complex, networked systems. UNIX is more advanced, supporting robust security, networking, and scalability, while DOS is simpler and limited to basic file and process management.

25. Explain the Resource Allocation Graph in Operating System.

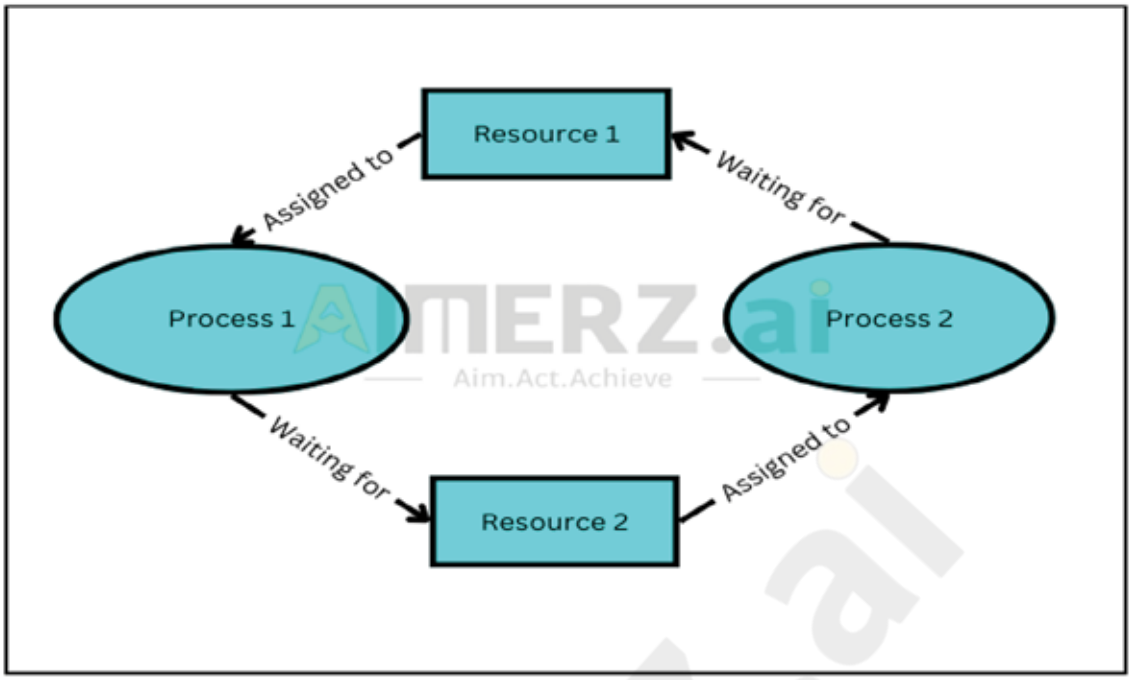
Resource Allocation Graph explains how the resources are allocated to the processes of our Operating System. There are many advantages of Using Resource Allocation Graph, one of the crucial advantages is sometimes we can observe deadlock directly seeing the RAG.

Most Repeated Questions

26. What is a Deadlock in an Operating System?

Deadlock is a situation where two or more processes are waiting for the other process to release the resource before proceeding.

Let us consider an example, as listed in the figure below. We can see that Resource 1 is already assigned to the Process 1 (P1) and still our Process (P1) is waiting for Resource 2 (R2) . Same situation is with Process 2(P2) where Resource 2(R2) is assigned to Process 2 (P2) while Process 2(P2) is still waiting for Resource 1 (R1) . This situation creates a deadlock.



Necessary Conditions For Deadlock:

There are following 4 necessary conditions for the occurrence of deadlock-

1. Mutual Exclusion
2. Hold and Wait
3. No preemption
4. Circular wait

1. Mutual Exclusion:

There must exist at least one resource in the system which can be used by only one process at a time. If there is such a resource, then deadlock will not happen.

2. Hold and Wait:

There must exist a process which holds some resource and waits for another resource held by some other process.

3. No preemption:

Once a resource has been allocated to the process, it cannot be preempted or rolled back until the process completes exec and releases the resource by

itself. It means resource can not be snatched forcefully from one process and given to the other process. Process is free to voluntarily release the resource whenever it wants.

4. Circular Wait:

All the processes are supposed to wait for the resource allocation in cyclic order. Where the last process waits for the resource held by the first process.

In the above example,

- Process P1 waits for a resource held by process P2.
- Process P2 waits for a resource held by process P1.

27. What do you understand about BIOS?

BIOS (Basic Input/Output System) is firmware that initializes and tests hardware components during the boot process before handing control to the operating system. It provides a low-level interface between the operating system and the computer's hardware.

NOTE: In modern operating systems a new kind of firmware is employed, UEFI (Unified Extensible Firmware Interface), UEFI provides a graphical user interface nowadays it is more commonly used and opted widely.

28. Define the Program thread?

The kernel and operating system processes carry out the assigned tasks as directed. Program Threats are what happen when a user program commands these processes to perform harmful operations. Programs installed on computers that have the ability to store and transmit user credentials to hackers over a network are among the most prevalent examples of program threats.

29. Explain something about System thread in Operating System?

System threats are when network connections and system services are abused to cause problems for users. Program attacks, also known as system threats, can be used to introduce program threats across an entire network. Threats to the system foster an environment where user files and operating system resources are abused.

30. List down some properties of the operating system?

- **Batch processing :** Execution of a series of processes in Batch by Batch of multiple processes.
- **Multitasking :** Running multiple tasks or processes simultaneously by sharing CPU time in a single operating system.
- **Multiprogramming :** Running multiple programs concurrently by keeping several programs in memory and switching between them.
- **Interactivity :** The ability of an operating system to interact with users by responding to their input in real time.
- **Real Time System :** A system that processes data and provides responses within strict timing constraints, often used in critical applications.
- **Distributed Environment :** A computing environment where processing and data are distributed across multiple interconnected machines.
- **Spooling :** The process of placing data in a buffer (spool) for asynchronous processing, commonly used in print spooling.

31. Explain the concept of the Time Sharing OS?

It is a logical extension of the multiprogrammed OS where users can interact with the program. The CPU executes multiple jobs by switching among them,

but the switches occur so frequently that the user feels as if the operating system is running only his program.

Advantages of Time Sharing operating systems are −

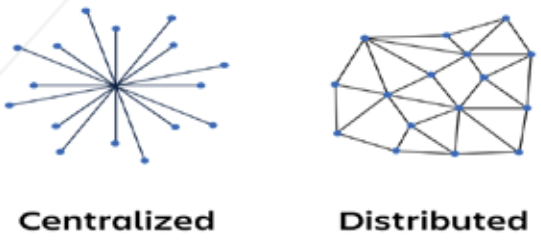
- It provides the advantage of quick response.
- This type of operating system avoids duplication of software.
- It reduces CPU idle time.

Disadvantages of Time-sharing operating systems are –

- Time sharing has a problem of reliability.
- Questions of security and integrity of user programs and data can be raised.
- Problem of data communication occurs.

32. Describe the concept of Distributed Systems?

In a network, distributed systems operate in multiple forms but with matching objectives. They are able to converse and share resources on the network.



33. What is the dining philosophers' problem in OS Resource Allocation?

Think of five philosophers who live contemplative lives and eat. Each of the five chairs surrounding the circular table that the philosophers share is owned by one of the philosophers. A bowl of rice sits in the middle of the table, which has five single chopsticks spread out across it. A philosopher does not communicate with other philosophers while she is thinking. A philosopher will occasionally become hungry and attempt to grab the two chopsticks that are nearest to her. A single chopstick may be picked up by a philosopher at a time. She obviously cannot pick the stick that is in someone else's hand. A philosopher who is hungry uses both of her chopsticks at the same time and eats without taking them out. After eating, she sets down both of her chopsticks and resumes her thought process.

Fig A: All philosophers thinking and none of them is holding resource (chopstick).
.

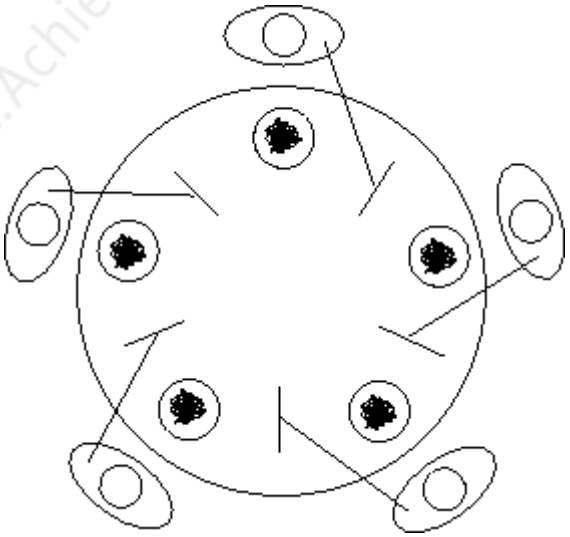
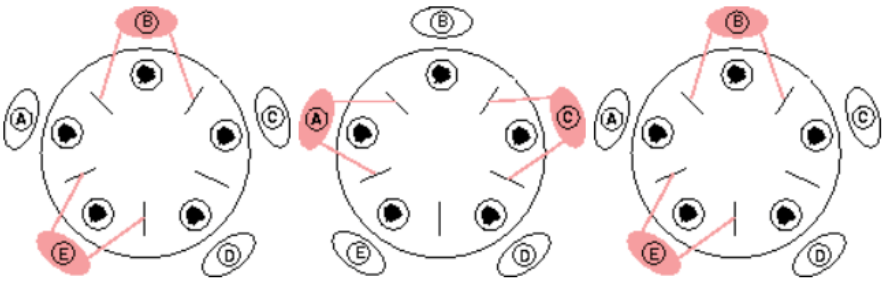


Fig B: All the philosophers are having one chopstick each and everyone is waiting for a resource (chopstick).

Possibility of Starvation



- If philosophers take two chopsticks at a time, there is a possibility of starvation, as shown in the figure. Philosophers B & E, and A & C can alternate in a way that starves out philosopher D.
- This possibility of starvation means that any solution to the problem must include some provision for preventing starvation.

Solution:

- **Resource hierarchy (Dijkstra)**: Number the chopsticks from 1 to 5; introduce the convention that each philosopher has to pick up a chopstick with a lower number first. This prevents deadlock because, after allocating four chopsticks to four philosophers, the last philosopher cannot pick up chopstick number 5, since 5 is the highest number.
- **Introduce the notion of cleanliness (Chandy/Misra):** label philosophers with an id ranging from 1 to *n*. For each pair of philosophers fighting over one particular chopstick, give it to the philosopher with the lower id. Each chopstick can be dirty or clean, and initially they are all dirty. When a philosopher wants to eat, he needs to request chopsticks. Any philosopher who receives a request keeps a chopstick if it is clean, but if it is dirty, he relinquished it. Before handing over a chopstick, he cleans it. After a philosopher is done eating, all chopsticks are marked as dirty.
- **Restrict Number of Philosophers:** Remove one chair (Stallings) take *n* philosophers and *n* chopsticks. Now, remove one chair so that only *n*-1 philosophers can take a seat.

34. What are the primary functions of a virtual file system or VFS?

A virtual file system (VFS) or virtual filesystem switch is an abstract layer on top of a more concrete file system. The purpose of a VFS is to allow client applications to access different types of concrete file systems in a uniform way.

For example, be used to access local and network storage devices transparently without the client application noticing the difference. It can be used to bridge the differences in Windows, classic Mac OS/macOS and Unix filesystems, so that applications can access files on local file systems of those types without having to know what type of file system they are accessing.

35. What are the operating system components?

1. Process management
2. Main memory management
3. File management
4. I/O system management
5. Secondary storage management
6. Networking
7. Protection system
8. Command interpreter system

36. What does the term "real-time operating systems" mean?

Real-time operating systems are frequently used as control devices in specialized applications because they are employed when strict time constraints are imposed on processor operation or data flow. Here, the computer receives data from the sensors. To change the sensor input, the computer must evaluate the data and possibly make control adjustments.

There are two categories for them:

1. A robust real-time operating system

2. Real-time soft OS

Hard-real-time operating systems have precise, set time limits. On the other hand, timing restrictions are looser for soft real-time operating systems.

37. Describe context switching.

The answer is that in order to move control to one process to another, the previous process' state must be saved and loaded into the new process. We call this process context switching.

38. What drawbacks do context switches have?

The amount of time it takes to switch between processes is entirely unnecessary. since the system doesn't perform any beneficial tasks when switching. Threading is one way to solve this, so use it whenever possible.

39. Describe cooperating processes and list an example about it.

The processes that exchange data and system resources with one another. Additionally, the processes can speak with one another through the communication between processes mechanism, which is frequently employed in distributed systems. The chat feature on the internet is the best example.

40. What methods are there to avoid deadlocks?

a. Mutual exclusion: Certain resources shouldn't be mutually exclusive, such as read-only files. They need to be transferable. However, some resources—like printers—must be incompatible with one another.

b. Hold and wait: We must make sure that a process shouldn't hold any resources when it requests one in order to prevent this scenario.

c. No preemption: All resources that are presently being held are preempted (released automatically) if a process that is holding some resources asks for a resource that cannot be immediately assigned to it; instead, the process must wai**t.**

d. Circular wait: a total ordering of all resource types and a requirement that each process request resources in a descending order of enumeration are the best ways to guarantee that this condition never holds.

41. Distinguish between an interpreter and a complier?

An interpreter reads each instruction one at a time and performs the tasks each one suggests. It doesn't translate anything. All of the instructions are translated, though, by a compiler**.**

42. What kinds of tasks does Lexical Analysis involve?

The lexical analyzer's job is to separate the input text and provide a series of basic symbols and comments. While basic symbols are character sequences that match terminal symbols of the grammar defining the input's phrase structure, comments are character sequences that should be disregarded.

43. What is Dispatcher?

Dispatcher module gives control of the CPU to the process selected by the short-term scheduler; this involves: Switching context Switching to user mode Jumping to the proper location in the user program to restart that program Dispatch latency – time it takes for the dispatcher to stop one process and start another running.

44. What are aging and starvation?

Starvation: This is a resource management issue in which resources are allotted to other processes, depriving a process of the resources it requires for an extended period of time.

Aging: In a scheduling system, aging is a way to prevent starvation. It functions by raising each request's priority by an aging factor. A request must eventually become the highest priority request (after it has waited long enough) and the aging factor must ensure that the request's priority increases over time.

45. Describe benefits of using segmentation and paging.

Because segments vary in length, it is more difficult to locate a spot in memory for a segment than for a page. The advantages of virtual memory are still available with segmented virtual memory, but dynamic storage allocation of physical memory is still required. Segmentation and paging can be combined to create a two-level virtual memory layout to prevent this. The page table for each segment is indicated by the segment descriptor.This

combines some of the benefits of segments (the program's logical division) with those of paging (easy placement).

THANK YOU

AIMERZ.ai
Aim.Act.Achieve