1. What is a process and process table?

A process is an instance of a program in execution. For example, a Web Browser is a process, and a shell (or command prompt) is a process. The operating system is responsible for managing all the processes that are running on a computer and allocates each process a certain amount of time to use the processor. In addition, the operating system also allocates various other resources that processes will need, such as computer memory or disks. To keep track of the state of all the processes, the operating system maintains a table known as the process table. Inside this table, every process is listed along with the resources the process is using and the current state of the process.

2. What are the different states of the process?

Processes can be in one of three states: running, ready, or waiting. The running state means that the process has all the resources it needs for execution and it has been given permission by the operating system to use the processor. Only one process can be in the running state at any given time. The remaining processes are either in a waiting state (i.e., waiting for some external event to occur such as user input or disk access) or a ready state (i.e., waiting for permission to use the processor). In a real operating system, the waiting and ready states are implemented as queues that hold the processes in these states.

3. What is a Thread?

A thread is a single sequence stream within a process. Because threads have some of the properties of processes, they are sometimes called lightweight processes. Threads are a popular way to improve the application through parallelism. For example, in a browser, multiple tabs can be different threads. MS Word uses multiple threads, one thread to format the text, another thread to process inputs, etc.

4. What are the differences between process and thread?

Process is program under action and thread is the smallest segment of instructions (segment of a process) that can be handled independently by a scheduler.

Threads are lightweight processes that share the same address space including the code section, data section and operating system resources such as the open files and signals. However, each thread has its own program counter (PC), register set and stack space allowing them to the execute independently within the same process context. Unlike **processes**, threads are not fully independent entities and can communicate and synchronize more efficiently making them

suitable for the concurrent and parallel execution in the multi-threaded environment.

5. What are the benefits of multithreaded programming?

Multithreaded programming makes the system more responsive and enables resource sharing. It leads to the use of multiprocess architecture. It is more economical and preferred.

7. What is Buffer?

A buffer is a memory area that stores data being transferred between two devices or between a device and an application.

8. What is virtual memory?

Virtual memory creates an illusion that each user has one or more contiguous address spaces, each beginning at address zero. The sizes of such virtual address spaces are generally very high. The idea of virtual memory is to use disk space to extend the RAM. Running processes don't need to care whether the memory is from RAM or disk. The illusion of such a large amount of memory is created by subdividing the virtual memory into smaller pieces, which can be loaded into physical memory whenever they are needed by a process.

9. Explain the main purpose of an operating system?

An operating system acts as an intermediary between the user of a computer and computer hardware. 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 a software that manages computer hardware. 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.

10. What is demand paging?

The process of loading the page into memory on demand (whenever a page fault occurs) is known as demand paging.

11. What is a kernel?

A kernel is the central component of an operating system that manages the operations of computers and hardware. It basically manages operations of memory and CPU time. It is a core component of an operating system. Kernel

acts as a bridge between applications and data processing performed at the hardware level using inter-process communication and system calls.

12. What are the different scheduling algorithms?

1. First-Come, First-Served (FCFS) Scheduling
2. Shortest-Job-Next (SJN) Scheduling
3. Priority Scheduling
4. Shortest Remaining Time
5. Round Robin(RR) Scheduling
6. Multiple-Level Queues Scheduling

13. Describe the objective of multi-programming?

Multi-programming increases CPU utilization by organizing jobs (code and data) so that the CPU always has one to execute. The main objective of multi-programming is to keep multiple jobs in the main memory. If one job gets occupied with IO, the CPU can be assigned to other jobs.

14. What is the time-sharing system?

Time-sharing is a logical extension of multiprogramming. The CPU performs many tasks by switches that are so frequent that the user can interact with each program while it is running. A time-shared operating system allows multiple users to share computers simultaneously.

15. What problem we face in computer system without OS?

- Poor resource management
- Lack of User Interface
- No File System
- No Networking
- Error handling

16. Give some benefits of multithreaded programming?

A thread is also known as a lightweight process. The idea is to achieve parallelism by dividing a process into multiple threads. Threads within the same process run in shared memory space,

17. Briefly explain FCFS?

FCFS stands for First Come First served. In the FCFS scheduling algorithm, the job that arrived first in the ready queue is allocated to the CPU and then the job that came second and so on. FCFS is a non-preemptive scheduling algorithm as a process that holds the CPU until it either terminates or performs I/O. Thus, if a longer job has been assigned to the CPU then many shorter jobs after it will have to wait.

18. What is the RR scheduling algorithm?

A round-robin scheduling algorithm is used to schedule the process fairly for each job in a time slot or quantum and interrupting the job if it is not completed by then the job comes after the other job which is arrived in the quantum time makes these scheduling fairly.

- Round-robin is cyclic in nature, so starvation doesn't occur
- Round-robin is a variant of first-come, first-served scheduling
- No priority or special importance is given to any process or task
- RR scheduling is also known as Time slicing scheduling

19. Enumerate the different RAID levels?

A redundant array of independent disks is a set of several physical disk drives that the operating system sees as a single logical unit. It played a significant role in narrowing the gap between increasingly fast processors and slow disk drives. RAID has different levels:

- Level-0
- Level-1
- Level-2
- Level-3
- Level-4
- Level-5
- Level-6

20. What is Banker's algorithm?

The banker's algorithm is a resource allocation and deadlock avoidance algorithm that tests for safety by simulating the allocation for the predetermined maximum possible amounts of all resources, then makes an "s-state" check to

test for possible activities, before deciding whether allocation should be allowed to continue.

21. State the main difference between logical and physical address space?

Parameter	LOGICAL ADDRESS	PHYSICAL ADDRESS
Basic	Logical address is generated by the CPU.	It is located in a memory unit.
Address Space	Logical Address Space is a set of all logical addresses generated by the CPU in reference to a program.	Physical Address is a set of all physical addresses mapped to the corresponding logical addresses.
Visibility	Users can view the logical address of a program.	Users can never view the physical address of the program.
Generation	generated by the CPU.	Computed by MMU.
Access	The user can use the logical address to access the physical address.	The user can indirectly access physical addresses but not directly.

22. How does dynamic loading aid in better memory space utilization?

With dynamic loading, a routine is not loaded until it is called. This method is especially useful when large amounts of code are needed in order to handle infrequently occurring cases such as error routines.

23. What is fragmentation?

Processes are stored and removed from memory, which makes free memory space, which is too little to even consider utilizing by different processes. Suppose, that process is not ready to dispense to memory blocks since its little size and memory hinder consistently staying unused is called fragmentation. This kind of issue occurs during a dynamic memory allotment framework when free blocks are small, so it can't satisfy any request.

24. What is the basic function of paging?

Paging is a method or technique which is used for non-contiguous memory allocation. It is a fixed-size partitioning theme (scheme). In paging, both main memory and secondary memory are divided into equal fixed-size partitions. The partitions of the secondary memory area unit and the main memory area unit are known as pages and frames respectively.

Paging is a memory management method accustomed fetch processes from the secondary memory into the main memory in the form of pages. in paging, each process is split into parts wherever the size of every part is the same as the page size. The size of the last half could also be but the page size. The pages of the process area unit hold on within the frames of main memory relying upon their accessibility

25. How does swapping result in better memory management?

Swapping is a simple memory/process management technique used by the operating system(os) to increase the utilization of the processor by moving some blocked processes from the main memory to the secondary memory thus forming a queue of the temporarily suspended processes and the execution continues with the newly arrived process. During regular intervals that are set by the operating system, processes can be copied from the main memory to a backing store and then copied back later. Swapping allows more processes to be run that can fit into memory at one time

26. Write a name of classic synchronization problems?

- Bounded-buffer
- Readers-writers
- Dining philosophers
- Sleeping barber

27. What is the Direct Access Method?

The direct Access method is based on a disk model of a file, such that it is viewed as a numbered sequence of blocks or records. It allows arbitrary blocks to be read or written. Direct access is advantageous when accessing large amounts of information. Direct memory access (DMA) is a method that allows an input/output (I/O) device to send or receive data directly to or from the main memory, bypassing the CPU to speed up memory operations. The process is managed by a chip known as a DMA controller (DMAC).

28. What is the best page size when designing an operating system?

The best paging size varies from system to system, so there is no single best when it comes to page size. There are different factors to consider in order to come up with a suitable page size, such as page table, paging time, and its effect on the overall efficiency of the operating system.

29. What is multitasking?

Multitasking is a logical extension of a multiprogramming system that supports multiple programs to run concurrently. In multitasking, more than one task is executed at the same time. In this technique, the multiple tasks, also known as processes, share common processing resources such as a CPU.

30. What is caching?

The cache is a smaller and faster memory that stores copies of the data from frequently used main memory locations. There are various different independent caches in a CPU, which store instructions and data. Cache memory is used to reduce the average time to access data from the Main memory.

31. What is spooling?

Spooling refers to simultaneous peripheral operations online, spooling refers to putting jobs in a buffer, a special area in memory, or on a disk where a device can access them when it is ready. Spooling is useful because devices access data at different rates.

32. What is the functionality of an Assembler?

The Assembler is used to translate the program written in Assembly language into machine code. The source program is an input of an assembler that contains assembly language instructions. The output generated by the assembler is the object code or machine code understandable by the computer.

33. What are interrupts?

The interrupts are a signal emitted by hardware or software when a process or an event needs immediate attention. It alerts the processor to a high-priority process requiring interruption of the current working process. In I/O devices one of the bus control lines is dedicated to this purpose and is called the Interrupt Service Routine (ISR).

34. What is GUI?

GUI is short for Graphical User Interface. It provides users with an interface wherein actions can be performed by interacting with icons and graphical symbols.

35. What is preemptive multitasking?

Preemptive multitasking is a type of multitasking that allows computer programs to share operating systems (OS) and underlying hardware resources. It divides the overall operating and computing time between processes, and the switching of resources between different processes occurs through predefined criteria.

36. What is a pipe and when is it used?

A Pipe is a technique used for inter-process communication. A pipe is a mechanism by which the output of one process is directed into the input of another process. Thus it provides a one-way flow of data between two related processes.

37. What are the advantages of semaphores?

- They are machine-independent.
- Easy to implement.
- Correctness is easy to determine.
- Can have many different critical sections with different semaphores.
- Semaphores acquire many resources simultaneously.
- No waste of resources due to busy waiting.

38. What is a bootstrap program in the OS?

Bootstrapping is the process of loading a set of instructions when a computer is first turned on or booted. During the startup process, diagnostic tests are performed, such as the power-on self-test (POST), which set or checks configurations for devices and implements routine testing for the connection of peripherals, hardware, and external memory devices. The bootloader or bootstrap program is then loaded to initialize the OS.

39. What is IPC?

Inter-process communication (IPC) is a mechanism that allows processes to communicate with each other and synchronize their actions. The communication between these processes can be seen as a method of cooperation between them.

40. What are the different IPC mechanisms?

These are the methods in IPC:

- **Pipes (Same Process)**: This allows a flow of data in one direction only. Analogous to simplex systems (Keyboard). Data from the output is usually buffered until the input process receives it which must have a common origin.
- **Named Pipes (Different Processes)**: This is a pipe with a specific name it can be used in processes that don't have a shared common process origin. E.g. FIFO where the details written to a pipe are first named.
- **Message Queuing**: This allows messages to be passed between processes using either a single queue or several message queues. This is managed by the system kernel these messages are coordinated using an API.
- **Semaphores**: This is used in solving problems associated with synchronization and avoiding race conditions. These are integer values that are greater than or equal to 0.
- **Shared Memory**: This allows the interchange of data through a defined area of memory. Semaphore values have to be obtained before data can get access to shared memory.
- **Sockets**: This method is mostly used to communicate over a network between a client and a server. It allows for a standard connection which is computer and OS independent

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

- In preemptive scheduling, the CPU is allocated to the processes for a limited time whereas, in Non-preemptive scheduling, the CPU is allocated to the process till it terminates or switches to waiting for the state.
- The executing process in preemptive scheduling is interrupted in the middle of execution when a higher priority one comes whereas, the executing process in non-preemptive scheduling is not interrupted in the middle of execution and waits till its execution.
- In Preemptive Scheduling, there is the overhead of switching the process from the ready state to the running state, vice-verse, and maintaining the ready queue. Whereas the case of non-preemptive scheduling has no overhead of switching the process from running state to ready state.

- In preemptive scheduling, if a high-priority process frequently arrives in the ready queue then the process with low priority has to wait for a long, and it may have to starve. On the other hand, in non-preemptive scheduling, if CPU is allocated to the process having a larger burst time then the processes with a small burst time may have to starve.
- Preemptive scheduling attains flexibility by allowing the critical processes to access the CPU as they arrive in the ready queue, no matter what process is executing currently. Non-preemptive scheduling is called rigid as even if a critical process enters the ready queue the process running CPU is not disturbed.
- Preemptive Scheduling has to maintain the integrity of shared data that's why it is cost associative which is not the case with Non-preemptive Scheduling.

42. What are starvation in OS?

Starvation: Starvation is a resource management problem where a process does not get the resources it needs for a long time because the resources are being allocated to other processes.

43. What is Context Switching?

Switching of CPU to another process means saving the state of the old process and loading the saved state for the new process. In Context Switching the process is stored in the Process Control Block to serve the new process so that the old process can be resumed from the same part it was left.

44. What is the difference between the Operating system and kernel?

Operating System	Kernel
Operating System is system software. I	The kernel is a core component of an operating system and serves as the main interface between the computer's physical hardware and the processes running on it.

Operating System	Kernel
Operating System provides an interface between the user and the computer hardware.	The kernel provides an interface between the application and hardware.
Its main purpose is memory management, disk management, process management and task management.	It manages the system resources, including processor, memory and device drivers.
Type of operating system includes single and multiuser OS, multiprocessor OS, real-time OS, Distributed OS.	Type of kernel includes Monolithic and Microkernel.

45. What is the difference between process and thread?

Process	Thread
Process means any program is in execution.	Thread means a segment of a process.
The process is less efficient in terms of communication.	Thread is more efficient in terms of communication.
The process is isolated.	Threads share memory.
The process is called heavyweight the process.	Thread is called lightweight process.

Process	Thread
Process switching uses, another process interface in operating system.	Thread switching does not require to call an operating system and cause an interrupt to the kernel.
If one process is blocked then it will not affect the execution of other process	The second, thread in the same task could not run, while one server thread is blocked.
The process has its own Process Control Block, Stack and Address Space.	Thread has Parents' PCB, its own Thread Control Block and Stack and common Address space.

46. What is PCB?

the process control block (PCB) is a block that is used to track the process's execution status. A process control block (PCB) contains information about the process, i.e. registers, quantum, priority, etc. The process table is an array of PCBs, that means logically contains a PCB for all of the current processes in the system.

47. Write a difference between program and process?

Program	Process
Program contains a set of instructions designed to complete a specific task.	Process is an instance of an executing program.
Program is a passive entity as it resides in the secondary memory.	Process is an active entity as it is created during execution and loaded into the main memory.

Program	Process
The program exists in a single place and continues to exist until it is deleted.	Process exists for a limited span of time as it gets terminated after the completion of a task.
A program is a static entity.	The process is a dynamic entity.
Program does not have any resource requirement, it only requires memory space for storing the instructions.	Process has a high resource requirement, it needs resources like CPU, memory address, and I/O during its lifetime.
The program does not have any control block.	The process has its own control block called Process Control Block.

48. What is a dispatcher?

The dispatcher is the module that gives process control over the CPU after it has been selected by the short-term scheduler. This function involves the following:

- Switching context
- Switching to user mode
- Jumping to the proper location in the user program to restart that program

49. What are the goals of CPU scheduling?

- Max CPU utilization [Keep CPU as busy as possible]Fair allocation of CPU.
- Max throughput [Number of processes that complete their execution per time unit]
- Min turnaround time [Time taken by a process to finish execution]
- Min waiting time [Time a process waits in ready queue]
- Min response time [Time when a process produces the first response]

50. What is a critical- section?

When more than one processes access the same code segment that segment is known as the critical section. The critical section contains shared variables or resources which are needed to be synchronized to maintain the consistency of data variables. In simple terms, a critical section is a group of instructions/statements or regions of code that need to be executed atomically such as accessing a resource (file, input or output port, global data, etc.).

51. Write the name of synchronization techniques?

- Mutexes
- Condition variables
- Semaphores
- File locks

52. Write a difference between a user-level thread and a kernel-level thread?

User-level thread	**Kernel level thread**
User threads are implemented by users.	kernel threads are implemented by OS.
OS doesn't recognize user-level threads.	Kernel threads are recognized by OS.
Implementation of User threads is easy.	Implementation of the perform kernel thread is complicated.
Context switch time is less.	Context switch time is more.
Context switch requires no hardware support.	Hardware support is needed.

User-level thread	Kernel level thread
If one user-level thread performs a blocking operation then entire process will be blocked.	If one kernel thread perform a the blocking operation then another thread can continue execution.
User-level threads are designed as dependent threads.	Kernel level threads are designed as independent threads.

53. Write down the advantages of multithreading?

Some of the most important benefits of MT are:

- Improved throughput. Many concurrent compute operations and I/O requests within a single process.
- Simultaneous and fully symmetric use of multiple processors for computation and I/O.
- Superior application responsiveness. If a request can be launched on its own thread, applications do not freeze or show the “hourglass”. An entire application will not block or otherwise wait, pending the completion of another request.
- Improved server responsiveness. Large or complex requests or slow clients don’t block other requests for service. The overall throughput of the server is much greater.
- Minimized system resource usage. Threads impose minimal impact on system resources. Threads require less overhead to create, maintain, and manage than a traditional process.
- Program structure simplification. Threads can be used to simplify the structure of complex applications, such as server-class and multimedia applications. Simple routines can be written for each activity, making complex programs easier to design and code, and more adaptive to a wide variation in user demands.
- Better communication. Thread synchronization functions can be used to provide enhanced process-to-process communication. In addition, sharing large amounts of data through separate threads of execution within the

same address space provides extremely high-bandwidth, low-latency communication between separate tasks within an application

54. Difference between Multithreading and Multitasking?

Multi-threading	**Multi-tasking**
Multiple threads are executing at the same time at the same or different part of the program.	Several programs are executed concurrently.
CPU switches between multiple threads.	CPU switches between multiple tasks and processes.
It is the process of a lightweight part.	It is a heavyweight process.
It is a feature of the process.	It is a feature of the OS.
Multi-threading is sharing of computing resources among threads of a single process.	Multitasking is sharing of computing resources(CPU, memory, devices, etc.) among processes.

55. What are the drawbacks of semaphores?

- Priority Inversion is a big limitation of semaphores.
- Their use is not enforced but is by convention only.
- The programmer has to keep track of all calls to wait and signal the semaphore.
- With improper use, a process may block indefinitely. Such a situation is called Deadlock.

56. What is Peterson's approach?

It is a concurrent programming algorithm. It is used to synchronize two processes that maintain the mutual exclusion for the shared resource. It uses two variables, a bool array flag of size 2 and an int variable turn to accomplish it.

57. Define the term Bounded waiting?

A system is said to follow bounded waiting conditions if a process wants to enter into a critical section will enter in some finite time.

58. What are the solutions to the critical section problem?

There are three solutions to the critical section problem:

- Software solutions
- Hardware solutions
- Semaphores

59. What is a Banker's algorithm?

The banker's algorithm is a resource allocation and deadlock avoidance algorithm that tests for safety by simulating the allocation for the predetermined maximum possible amounts of all resources, then makes an "s-state" check to test for possible activities, before deciding whether allocation should be allowed to continue.

60. What is concurrency?

A state in which a process exists simultaneously with another process than those it is said to be concurrent.

61. Write a drawback of concurrency?

- It is required to protect multiple applications from one another.
- It is required to coordinate multiple applications through additional mechanisms.
- Additional performance overheads and complexities in operating systems are required for switching among applications.
- Sometimes running too many applications concurrently leads to severely degraded performance.

62. What are the necessary conditions which can lead to a deadlock in a system?

Mutual Exclusion: There is a resource that cannot be shared.
Hold and Wait: A process is holding at least one resource and waiting for another resource, which is with some other process.
No Preemption: The operating system is not allowed to take a resource back

from a process until the process gives it back.
Circular Wait: A set of processes waiting for each other in circular form.

63. What are the issues related to concurrency?

- **Non-atomic**: Operations that are non-atomic but interruptible by multiple processes can cause problems.
- Race conditions: A race condition occurs of the outcome depends on which of several processes gets to a point first.
- **Blocking**: Processes can block waiting for resources. A process could be blocked for a long period of time waiting for input from a terminal. If the process is required to periodically update some data, this would be very undesirable.
- Starvation: It occurs when a process does not obtain service to progress.
- Deadlock: It occurs when two processes are blocked and hence neither can proceed to execute

64. Explain the resource allocation graph?

The resource allocation graph is explained to us what is the state of the system in terms of processes and resources. One of the advantages of having a diagram is, sometimes it is possible to see a deadlock directly by using RAG.

65. What is a deadlock?

Deadlock is a situation when two or more processes wait for each other to finish and none of them ever finish. Consider an example when two trains are coming toward each other on the same track and there is only one track, none of the trains can move once they are in front of each other. A similar situation occurs in operating systems when there are two or more processes that hold some resources and wait for resources held by other(s).

66. What is the goal and functionality of memory management?

The goal and functionality of memory management are as follows;

- Relocation
- Protection
- Sharing
- Logical organization
- Physical organization

67. Write a difference between physical address and logical address?

Parameters	Logical address	Physical Address
Basic	It is the virtual address generated by CPU.	The physical address is a location in a memory unit.
Address	Set of all logical addresses generated by the CPU in reference to a program is referred to as Logical Address Space.	Set of all physical addresses mapped to the corresponding logical addresses is referred to as a Physical Address.
Visibility	The user can view the logical address of a program.	The user can never view the physical address of the program
Access	The user uses the logical address to access the physical address.	The user can not directly access the physical address
Generation	The Logical Address is generated by the CPU	Physical Address is Computed by MMU

68. Explain address binding?

The Association of program instruction and data to the actual physical memory locations is called Address Binding.

69. Write different types of address binding?

Address Binding is divided into three types as follows.

- Compile-time Address Binding
- Load time Address Binding
- Execution time Address Binding

70. Write an advantage of dynamic allocation algorithms?

- When we do not know how much amount of memory would be needed for the program beforehand.
- When we want data structures without any upper limit of memory space.
- When you want to use your memory space more efficiently.
- Dynamically created lists insertions and deletions can be done very easily just by the manipulation of addresses whereas in the case of statically allocated memory insertions and deletions lead to more movements and wastage of memory.
- When you want to use the concept of structures and linked lists in programming, dynamic memory allocation is a must

71. Write a difference between internal fragmentation and external fragmentation?

Internal fragmentation	**External fragmentation**
In internal fragmentation fixed-sized memory, blocks square measure appointed to process.	In external fragmentation, variable-sized memory blocks square measure appointed to the method.
Internal fragmentation happens when the method or process is larger than the memory.	External fragmentation happens when the method or process is removed.
The solution to internal fragmentation is the best-fit block.	Solution for external fragmentation is compaction, paging and segmentation.
Internal fragmentation occurs when memory is divided into fixed-sized partitions.	External fragmentation occurs when memory is divided into variable-size partitions based on the size of processes.
The difference between memory allocated and required space or	The unused spaces formed between non-contiguous memory fragments are too

Internal fragmentation	External fragmentation
memory is called Internal fragmentation.	small to serve a new process, which is called External fragmentation.

72. Write about the advantages and disadvantages of a hashed-page table?

Advantages

- The main advantage is synchronization.
- In many situations, hash tables turn out to be more efficient than search trees or any other table lookup structure. For this reason, they are widely used in many kinds of computer software, particularly for associative arrays, database indexing, caches, and sets.

Disadvantages

- Hash collisions are practically unavoidable. when hashing a random subset of a large set of possible keys.
- Hash tables become quite inefficient when there are many collisions.
- Hash table does not allow null values, like a hash map.
- Define Compaction.

73. Write a difference between paging and segmentation?

Paging	Segmentation
In paging, program is divided into fixed or mounted-size pages.	In segmentation, the program is divided into variable-size sections.
For the paging operating system is accountable.	For segmentation compiler is accountable.
Page size is determined by hardware.	Here, the section size is given by the user.

Paging	Segmentation
It is faster in comparison of segmentation.	Segmentation is slow.
Paging could result in internal fragmentation.	Segmentation could result in external fragmentation.
In paging, logical address is split into that page number and page offset.	Here, logical address is split into section number and section offset.
Paging comprises a page table which encloses the base address of every page.	While segmentation also comprises the segment table which encloses the segment number and segment offset.
A page table is employed to keep up the page data.	Section Table maintains the section data.
In paging, operating system must maintain a free frame list.	In segmentation, the operating system maintains a list of holes in the main memory.
Paging is invisible to the user.	Segmentation is visible to the user.
In paging, processor needs page number, offset to calculate the absolute address.	In segmentation, the processor uses segment number, and offset to calculate the full address.

74. Write a definition of Associative Memory and Cache Memory?

Associative Memory	Cache Memory
A memory unit accessed by content is called associative memory.	Fast and small memory is called cache memory.
It reduces the time required to find the item stored in memory.	It reduces the average memory access time.
Here data is accessed by its content.	Here, data are accessed by their address.
It is used where search time is very short.	It is used when a particular group of data is accessed repeatedly.
Its basic characteristic is its logic circuit for matching its content.	Its basic characteristic is its fast access

75. Write down the advantages of virtual memory?

- A higher degree of multiprogramming.
- Allocating memory is easy and cheap
- Eliminates external fragmentation
- Data (page frames) can be scattered all over the PM
- Pages are mapped appropriately anyway
- Large programs can be written, as the virtual space available is huge compared to physical memory.
- Less I/O required leads to faster and easy swapping of processes.
- More physical memory is available, as programs are stored on virtual memory, so they occupy very less space on actual physical memory.
- More efficient swapping

76. How to calculate performance in virtual memory?

The performance of virtual memory of a virtual memory management system depends on the total number of page faults, which depend on “paging policies” and “frame allocation”.

Effective access time = (1-p) x Memory access time + p x page fault time

77. Write down the basic concept of the file system?

A file is a collection of related information that is recorded on secondary storage. Or file is a collection of logically related entities. From the user’s perspective, a file is the smallest allotment of logical secondary storage.

78. Write the names of different operations on file?

Operation on file:

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

79. Define the term Bit-Vector?

A Bitmap or Bit Vector is a series or collection of bits where each bit corresponds to a disk block. The bit can take two values: 0 and 1: 0 indicates that the block is allocated and 1 indicates a free block.

80. What is a File allocation table?

FAT stands for File Allocation Table and this is called so because it allocates different files and folders using tables. This was originally designed to handle small file systems and disks. A file allocation table (FAT) is a table that an operating system maintains on a hard disk that provides a map of the cluster (the basic units of logical storage on a hard disk) that a file has been stored in.

Advanced OS Interview Questions

81. What is Belady’s Anomaly?

Bélády's anomaly is an anomaly with some page replacement policies increasing the number of page frames resulting in an increase in the number of page faults. It occurs when the First In First Out page replacement is used.

82. What are the advantages of a multiprocessor system?

There are some main advantages of a multiprocessor system:

- Enhanced performance.
- Multiple applications.
- Multi-tasking inside an application.
- High throughput and responsiveness.
- Hardware sharing among CPUs.

83. What are real-time systems?

A real-time system means that the system is subjected to real-time, i.e., the response should be guaranteed within a specified timing constraint or the system should meet the specified deadline.

84. How to recover from a deadlock?

We can recover from a deadlock by following methods:

- Process termination
    - Abort all the deadlock processes
    - Abort one process at a time until the deadlock is eliminated
- Resource preemption
    - Rollback
    - Selecting a victim

85. What factors determine whether a detection algorithm must be utilized in a deadlock avoidance system?

One is that it depends on how often a deadlock is likely to occur under the implementation of this algorithm. The other has to do with how many processes will be affected by deadlock when this algorithm is applied.

86. Explain the resource allocation graph?

The resource allocation graph is explained to us what is the state of the system in terms of processes and resources. One of the advantages of having a diagram is, sometimes it is possible to see a deadlock directly by using RAG.