1)What is Operating system? Explain various functions and objectives?

- An operating system is a program that acts as an interface between the software and the computer hardware.

The objectives of the operating system are −

- To make the computer system convenient to use in an efficient manner.
- To hide the details of the hardware resources from the users.
- To provide users a convenient interface to use the computer system.

2) Differentiate between monolithic and microkernel structure of OS.

Microkernel	Monolithic kernel
In microkernel user services and kernel, services are kept in separate address space.	In monolithic kernel, both user services and kernel services are kept in the same address space.
OS is complex to design.	OS is easy to design and implement.

3) Write a short note on system calls.

a **system call** is the programmatic way in which a computer program requests a service from the kernel of the operating system it is executed on.

4) Explain different types of OS.

Batch Operating System

In a Batch Operating System, the similar jobs are grouped together into batches with the help of some operator and these batches are executed one by one

Time-Sharing Operating System

In a Multi-tasking Operating System, more than one processes are being executed at a particular time with the help of the time-sharing concept

Distributed Operating System

In a Distributed Operating System, we have various systems and all these systems have their own CPU, main memory, secondary memory, and resources.

Real-time Operating System

The Real-time Operating Systems are used in the situation where we are dealing with some real-time data. So, as soon as the data comes, the execution of the process should be done and there should be no dealy i.e. no buffer delays should be there

5)Explain any five system calls.

Process Control

These system calls deal with processes such as process creation, process termination etc.

File Management

These system calls are responsible for file manipulation such as creating a file, reading a file, writing into a file etc.

Device Management

These system calls are responsible for device manipulation such as reading from device buffers, writing into device buffers etc.

Information Maintenance

These system calls handle information and its transfer between the operating system and the user program.

Communication

These system calls are useful for interprocess communication. They also deal with creating and deleting a communication connection.

6) explain five state process model

1. **Running:** It means a process that is currently being executed. Assuming that there is only a single processor in the below execution process, so there will be at most one processor at a time that can be running in the state.
2. **Ready:** It means a process that is prepared to execute when given the opportunity by the OS.
3. **Waiting:** It means that a process cannot continue executing until some event occurs like for example, the completion of an input-output operation.
4. **New:** It means a new process that has been created but has not yet been admitted by the OS for its execution. A new process is not loaded into the main memory, but its process control block (PCB) has been created.
5. **Exit/Terminate:** A process or job that has been released by the OS, either because it is completed or is aborted for some issue.

7) What is Process Control Block (PCB)?

For each process, the operating system maintains the **data structure**, which keeps the complete information about that process. This record or data structure is called **Process Control Block (PCB).**

8)What do you mean by process?

A **process** is a series of actions which are carried out in order to achieve a particular result.

10) What is context switch?

The Context switching is a technique or method used by the operating system to switch a process from one state to another to execute its function using CPUs in the system.

11) Differentiate between process and threads.

process	
A process is a program under execution i.e an active program.	A thread is a lightweight process that can be managed independently by a scheduler.
Processes require more time for context switching as they are more heavy.	Threads require less time for context switching a they are lighter than processes.

13) Compare short term, medium term and long-term scheduler along with diagram

Long Term	Short Term	Medium Term
It is a job scheduler.	It is a CPU scheduler.	It is swapping.
Speed is less than short term scheduler.	Speed is very fast.	Speed is in between both
It controls the degree of multiprogramming.	Less control over the degree of multiprogramming.	Reduce the degree of multiprogramming.
Absent or minimal in a time-sharing system.	Minimal in a time-sharing system.	Time-sharing system uses a medium-term scheduler.

14) Describe microkernel operating system structure

Microkernel is a software or code which contains the required minimum amount of functions, data, and features to implement an operating system.

Q) What is Kernel?

A kernel is an important part of an OS that manages system resources. It also acts as a bridge between the software and hardware of the computer. It is one of the first program which is loaded on start-up after the bootloader.

15) What is thread?

A thread is a path of execution within a process. A process can contain multiple threads.

16) Differentiate Multitasking & Multiprogramming?

Multiprogramming	Multi-tasking
Both of these concepts are for single CPU.	Both of these concepts are for single CPU.
Concept of Context Switching is used.	Concept of Context Switching and Time Sharing is used.
In multiprogrammed system, the operating system simply switches to, and executes, another job when current job needs to wait.	The processor is typically used in time sharing mode. Switching happens when either allowed time expires or where there other reason for current process needs to wait (example process needs to do IO).

17) Explain various operating system services?

Following are a few common services provided by an operating system −

- Program execution
- I/O operations
- File System manipulation

- Communication
- Error Detection
- Resource Allocation
- Protection

19) Discuss various scheduling criteria

1. **CPU utilisation –**
   The main objective of any CPU scheduling algorithm is to keep the CPU as busy as possible. Theoretically, CPU utilisation can range from 0 to 100 but in a real-time system, it varies from 40 to 90 percent depending on the load upon the system.

2. **Throughput –**
   A measure of the work done by CPU is the number of processes being executed and completed per unit time. This is called throughput. The throughput may vary depending upon the length or duration of processes.

3. **Turnaround time –**
   For a particular process, an important criteria is how long it takes to execute that process. The time elapsed from the time of submission of a process to the time of completion is known as the turnaround time. Turn-around time is the sum of times spent waiting to get into memory, waiting in ready queue, executing in CPU, and waiting for I/O.

4. **Waiting time –**
   A scheduling algorithm does not affect the time required to complete the process once it starts execution. It only affects the waiting time of a process i.e. time spent by a process waiting in the ready queue.

5. **Response time –**
   In an interactive system, turn-around time is not the best criteria. A process may produce some output fairly early and continue computing new results while previous results are being output to the user. Thus another criteria is the time taken from submission of the process of request until the first response is produced. This measure is called response time.

20) Explain schedulers and its types?

- A scheduler is a type of system software that allows you to handle process scheduling.

- Three types of the scheduler are 1) Long term 2) Short term 3) Medium-term
- Long term scheduler regulates the program and select process from the queue and loads them into memory for execution.

22) Describe the operating System as a resource Manager.

- Operating system allows multiple programs to be in memory and run at the same time.
- Resource management includes multiplexing or sharing resources in two different ways: in time and in space.
- In time multiplexed, different programs take a chance of using CPU.

Q)What Is deadlock?

A set of blocked processes each holding a resource and waiting to acquire a resource held by another process in the set

Q) What are the Necessary Conditions to Occur Deadlock?

There are multiple necessary conditions to occur deadlock in the operating system. Even if one condition does not get satisfied, deadlock cannot occur. These conditions are as follows.

- Mutual Exclusion
- Hold and Wait
- No Preemption
- Circular Wait

What are the different types of Deadlock Types and Handling Strategies?

- Prevention of Deadlock
- Avoidance of Deadlock
- Deadlock Detection and Recover
- Deadlock Ignorance

What is Banker's algorithm?

Banker's algorithm is used to avoid deadlock. It is the one of deadlock-avoidance method. It is named as Banker's algorithm on the banking system where bank never allocates available cash in such a manner that it can no longer satisfy the requirements of all of its customers.

What is semaphore?

Semaphore is a protected variable or abstract data type that is used to lock the resource being used. The value of the semaphore indicates the status of a common resource.

What is aging in Operating System?

Aging is a technique used to avoid the starvation in resource scheduling system.

Disk Scheduling Algorithms

Disk scheduling is done by operating systems to schedule I/O requests arriving for the disk. Disk scheduling is also known as I/O scheduling.
Disk scheduling is important because:

- Multiple I/O requests may arrive by different processes and only one I/O request can be served at a time by the disk controller. Thus other I/O requests need to wait in the waiting queue and need to be scheduled.
- Two or more request may be far from each other so can result in greater disk arm movement.
- Hard drives are one of the slowest parts of the computer system and thus need to be accessed in an efficient manner.

FCFS: FCFS is the simplest of all the Disk Scheduling Algorithms. In FCFS, the requests are addressed in the order they arrive in the disk queue.Let us understand this with the help of an example.

SSTF: In SSTF (Shortest Seek Time First), requests having shortest seek time are executed first. So, the seek time of every request is calculated in advance in the queue and then they are scheduled according to their calculated seek time. As a result, the request near the disk arm will get executed first. SSTF is certainly an improvement over FCFS as it decreases

the average response time and increases the throughput of system.Let us understand this with the help of an example.

SCAN: In SCAN algorithm the disk arm moves into a particular direction and services the requests coming in its path and after reaching the end of disk, it reverses its direction and again services the request arriving in its path. So, this algorithm works as an elevator and hence also known as **elevator algorithm.** As a result, the requests at the midrange are serviced more and those arriving behind the disk arm will have to wait.

1. **CSCAN**: In SCAN algorithm, the disk arm again scans the path that has been scanned, after reversing its direction. So, it may be possible that too many requests are waiting at the other end or there may be zero or few requests pending at the scanned area.

LOOK: It is similar to the SCAN disk scheduling algorithm except for the difference that the disk arm in spite of going to the end of the disk goes only to the last request to be serviced in front of the head and then reverses its direction from there only. Thus it prevents the extra delay which occurred due to unnecessary traversal to the end of the disk.

CLOOK: As LOOK is similar to SCAN algorithm, in similar way, CLOOK is similar to CSCAN disk scheduling algorithm. In CLOOK, the disk arm in spite of going to the end goes only to the last request to be serviced in front of the head and then from there goes to the other end's last request. Thus, it also prevents the extra delay which occurred due to unnecessary traversal to the end of the disk.

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

Page Fault – A page fault happens when a running program accesses a memory page that is mapped into the virtual address space but not loaded in physical memory.

Page Replacement Algorithms :
1. First In First Out (FIFO) –
This is the simplest page replacement algorithm. In this algorithm, the operating system keeps track of all pages in the memory in a queue, the oldest page is in the front of the queue. When a page needs to be replaced page in the front of the queue is selected for removal.

2. Optimal Page replacement –
In this algorithm, pages are replaced which would not be used for the longest duration of time in the future.