**Module1:**

**Chapter1:**

1. How are network computers different from traditional personal computers? Describe some usage scenarios in which it is advantageous to use network computers.
2. Give two reasons why caches are useful. What problems do they solve? What problems do they cause? If a cache can be made as large as the device for which it is caching (for instance, a cache as large as a disk), why not make it that large and eliminate the device?
3. Under what circumstances would a user be better off using a timesharing system rather than a PC or a single-user workstation?
4. List the four steps that are necessary to run a program on a completely dedicated machine computer that is running only that program.
5. How does the distinction between kernel mode and user mode function as a rudimentary form of protection (security) system?
6. In a multiprogramming and time-sharing environment, several users share the system simultaneously. This situation can result in various security problems.

a. What are two such problems?

b. Can we ensure the same degree of security in a time-shared machine as in a dedicated machine? Explain your answer.

1. Describe a mechanism for enforcing memory protection in order to prevent a program from modifying the memory associated with other programs.
2. What are the tradeoffs inherent in handheld computers?
3. Distinguish between the client-server and peer-to-peer models of distributed systems.
4. Some computer systems do not provide a privileged mode of operation in hardware. Is it possible to construct a secure operating system for these computer systems? Give arguments both that it is and that it is not possible.
5. What are the main differences between operating systems for mainframe computers and personal computers?
6. Discuss, with examples, how the problem of maintaining coherence of cached data manifests itself in the following processing environments:

a. Single-processor systems

b. Multiprocessor systems

c. Distributed systems

1. Identify several advantages and several disadvantages of open-source operating systems. Include the types of people who would find each aspect to be an advantage or a disadvantage.
2. How do clustered systems differ from multiprocessor systems? What is required for two machines belonging to a cluster to cooperate to provide a highly available service?
3. What is the main difficulty that a programmer must overcome in writing an operating system for a real-time environment?
4. Direct memory access is used for high-speed I/O devices in order to avoid increasing the CPU's execution load.

a. How does the CPU interface with the device to coordinate the transfer?

b. How does the CPU know when the memory operations are complete?

c. The CPU is allowed to execute other programs while the DMA controller is transferring data. Does this process interfere with the execution of the user programs? If so, describe what forms of interference are caused.

1. Some CPUs provide for more than two modes of operation. What are two possible uses of these multiple modes?
2. Define the essential properties of the following types of operating systems:

a. Batch

b. Interactive

c. Time sharing

d. Real time

e. Network

f. Parallel

g.Distributed

h. Clustered

i. Handheld

1. Describe the differences between symmetric and asymmetric multiprocessing.
2. What are three advantages and one disadvantage of multiprocessor systems?
3. The issue of resource utilization shows up in different forms in different types of operating systems. List what resources must be managed carefully in the following settings:

a. Mainframe or minicomputer systems

b. Workstations connected to servers

c. Handheld computers

1. What is the purpose of interrupts? What are the differences between a trap and an interrupt? Can traps be generated intentionally by a user program? If so, for what purpose?
2. Consider a computing cluster consisting of two nodes running a database. Describe two ways in which the cluster software can manage access to the data on the disk. Discuss the benefits and disadvantages of each.

**Chapter2:**

1. What are the five major activities of an operating system with regard to file management?
2. What are the three major activities of an operating system with regard to memory management?
3. Why is a just-in-time compiler useful for executing Java programs?
4. The services and functions provided by an operating system can be divided into two main categories. Briefly describe the two categories and discuss how they differ.
5. Why is the separation of mechanism and policy desirable?
6. Would it be possible for the user to develop a new command interpreter using the system-call interface provided by the operating system?
7. What is the purpose of the command interpreter? Why is it usually

separate from the kernel?

1. What is the main advantage for an operating-system designer of using

a virtual-machine architecture? What is the main advantage for a user?

1. It is sometimes difficult to achieve a layered approach if two components of the operating system are dependent on each other. Identify a scenario in which it is unclear how to layer two system components that require tight coupling of their functionalities.
2. What is the main advantage of the layered approach to system design? What are the disadvantages of using the layered approach?
3. What is the relationship between a guest operating system and a host operating system in. a system like VMware? What factors need to be considered in choosing the host operating system?
4. Describe three general methods for passing parameters to the operating system.
5. What is the main advantage of the microkernel approach to system design? How do user programs and system services interact in a microkernel architecture? What are the disadvantages of using the microkernel approach?
6. What system calls have to be executed by a command interpreter or shell in order to start a new process?
7. What are the two models of interprocess communication? What are the strengths and weaknesses of the two approaches?
8. In what ways is the modular kernel approach similar to the layered approach? In what ways does it differ from the layered approach?
9. How could a system be designed to allow a choice of operating systems from which to boot? What would the bootstrap program need to do?
10. What are the advantages and disadvantages of using the same system call interface for manipulating both files and devices?
11. Describe how you could obtain a statistical profile of the amount of time spent by a program executing different sections of its code. Discuss the importance of obtaining such a statistical profile.
12. Why do some systems store the operating system in firmware, while others store it on disk?