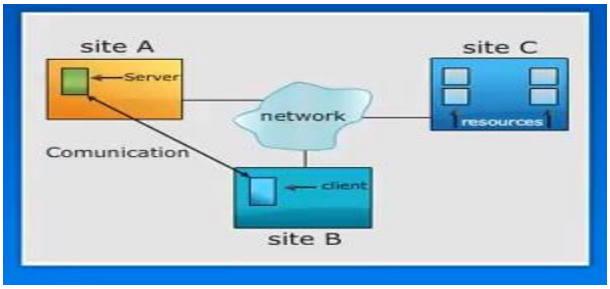
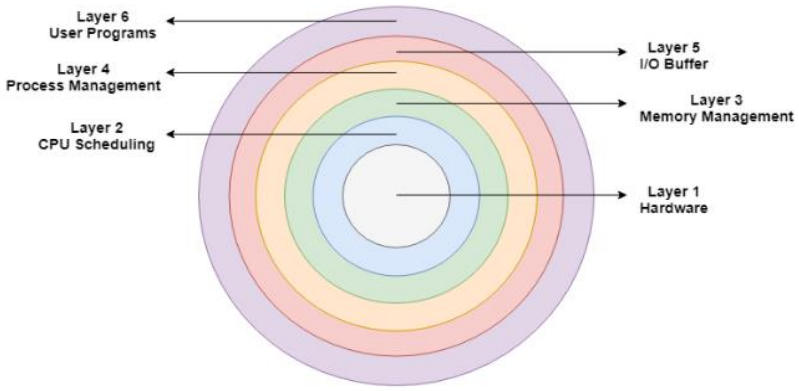
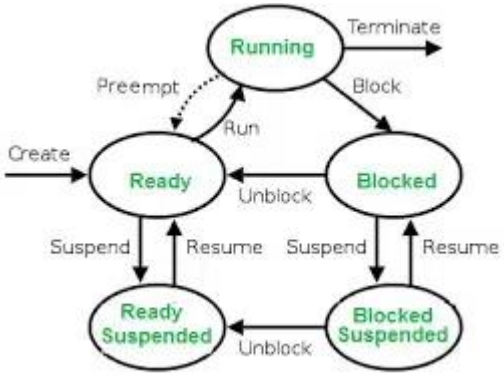
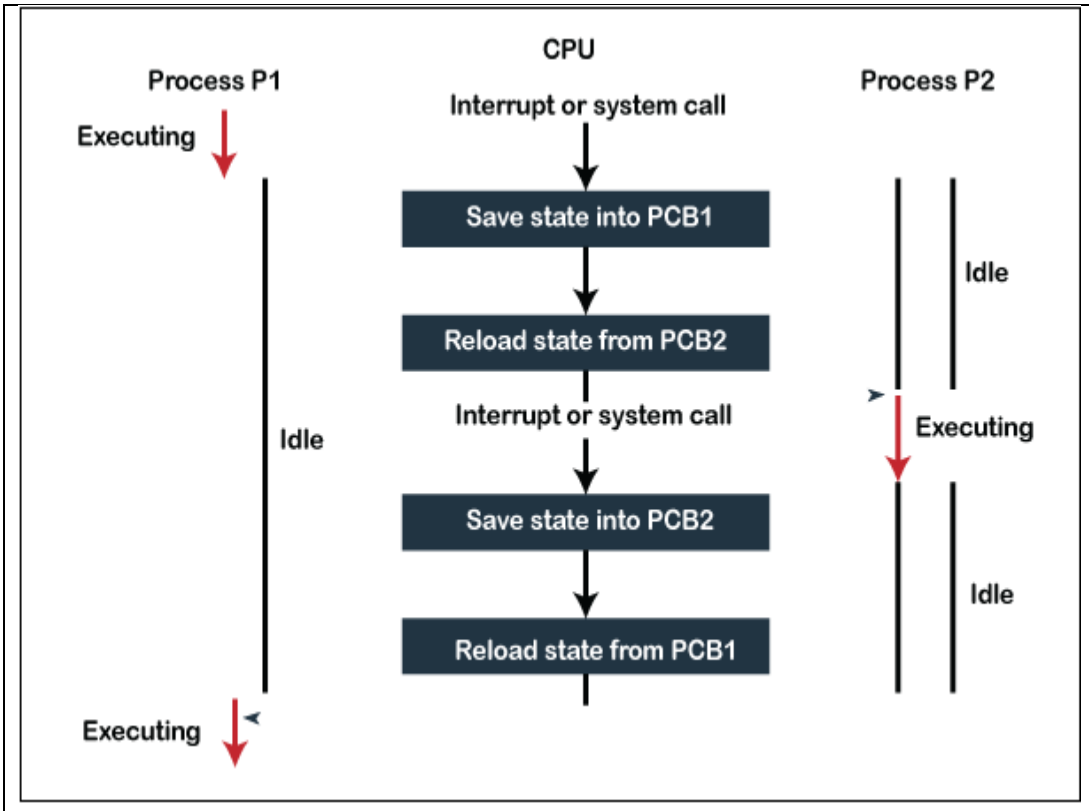
USN											
Subject:	OPERATING SYSTEM							Sub Code:	21CS44		
Date:	07-07-2023	Duration:	90 Min's	Sem/Sec:	IV Sem-A						

Note: Answer any Five Full Questions	Marks	CO	RBT
1 a. What is a distributed operating system? What are the advantages of the distributed operating system? **Explanation: 2.5 Marks** **Advantages: 2.5 Marks** A distributed system is a collection of systems that are networked to provide the users with access to the various resources in the network.  **Advantages of Distributed Operating System** ● Data Sharing. Distributed systems allow one component to share data with other components in the system easily. ● Ability of extension. Scalability means that we can change the size and scope of a particular system. ● Error management. ● Reliability. ● Effective. ● Less delay.	5	CO1	L2
1 b. What is IPC? Explain direct and indirect communication with respect to message passing. **IPC Definition: 1 Mark** **Direct communication: 2 Marks** **Indirect communication: 2 Marks** **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 co-operation between them. Processes can communicate with each other through both: 1. Shared Memory 2. Message passing **Direct and indirect communication :** **Direct communication**:The process which wants to communicate must explicitly name the recipient or sender of the communication. e.g. **send(p1, message)** means send the message to p1. Similarly, **receive(p2, message)** means to receive the message from p2. In this method of communication, the communication link gets established automatically, which can be either unidirectional or bidirectional, but one link can be used between one	5	CO1	L2

pair of the sender and receiver and one pair of sender and receiver should not possess more than one pair of links. Symmetry and asymmetry between sending and receiving can also be implemented i.e. either both processes will name each other for sending and receiving the messages or only the sender will name the receiver for sending the message and there is no need for the receiver for naming the sender for receiving the message. The problem with this method of communication is that if the name of one process changes, this method will not work. **Indirect communication:** processes use mailboxes (also referred to as ports) for sending and receiving messages. Each mailbox has a unique id and processes can communicate only if they share a mailbox. Link established only if processes share a common mailbox and a single link can be associated with many processes. Each pair of processes can share several communication links and these links may be unidirectional or bi-directional. Suppose two processes want to communicate through Indirect message passing, the required operations are: create a mailbox, use this mailbox for sending and receiving messages, then destroy the mailbox. The standard primitives used are: **send(A, message)** which means send the message to mailbox A. The primitive for the receiving the message also works in the same way e.g. **received (A, message)**.			
2 a. Explain the advantages of the layered approach with a neat diagram? **Explanation: 2.5 Marks** **Diagram: 2.5 Marks** Layering provides a distinct advantage in an operating system. All the layers can be defined separately and interact with each other as required. Also, it is easier to create, maintain and update the system if it is done in the form of layers. Change in one layer specification does not affect the rest of the layers. Each of the layers in the operating system can only interact with the layers that are above and below it. The lowest layer handles the hardware and the uppermost layer deals with the user applications.  LAYERED OPERATING SYSTEM	5	CO1	L1
2 b. Explain the various functions of operating system with respect to process and memory management. **Explanation: 3 Marks** **Diagram: 2 Marks** **Processor Management:** In a multi-programming environment, the OS decides the order in which processes have access to the processor, and how much processing time each process has. This function	5	CO1	L1

of OS is called Process Scheduling. An Operating System performs the following activities for Processor Management. Keeps track of the status of processes. The program which performs this task is known as a traffic controller. Allocates the CPU that is a processor to a process. De-allocates processor when a process is no more required.  Memory management: An Operating System performs the following activities for Memory Management: • It keeps track of primary memory, i.e., which bytes of memory are used by which user program. The memory addresses that have already been allocated and the memory addresses of the memory that has not yet been used. • In multiprogramming, the OS decides the order in which processes are granted memory access, and for how long. • It Allocates the memory to a process when the process requests it and deallocates the memory when the process has terminated or is performing an I/O operation.			
3 Explain PCB. When the context switching will occur explain with a neat diagram? How will you calculate the context switch time between two running processes. **<u>PCB Explanation: 4 Marks</u>** **<u>Context Switching Explanation: 4 Marks</u>** **<u>Diagram: 2 Marks</u>** **PCB:** A PCB (Process Control Block) is a data structure used in the operating system to store all data related information to the process. For example, when a process is created in the operating system, updated information of the process, switching information of the process, terminated process in the PCB. **Steps for Context Switching** There are several steps involves in context switching of the processes. The following diagram represents the context switching of two processes, P1 to P2, when an interrupt, I/O needs, or priority-based process occurs in the ready queue of PCB.	10	CO1	L3



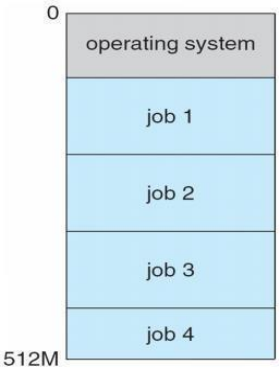
when switching Process P1 to Process 2:

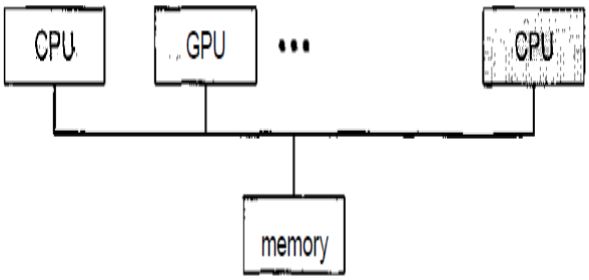
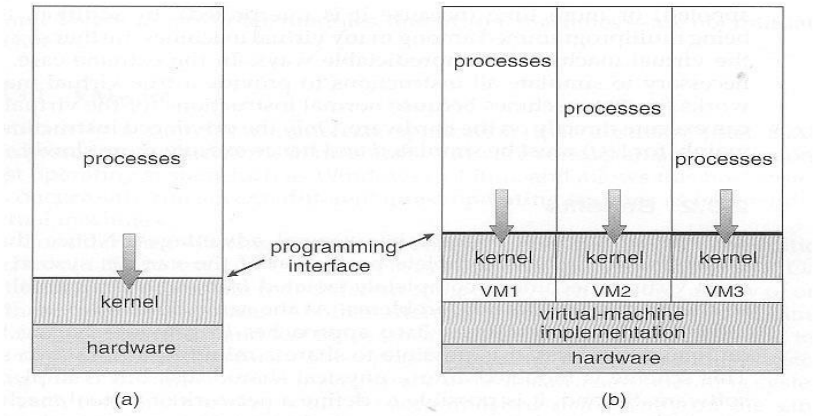
1. First, thes context switching needs to save the state of process P1 in the form of the program counter and the registers to the PCB (Program Counter Block), which is in the running state.
2. Now update PCB1 to process P1 and moves the process to the appropriate queue, such as the ready queue, I/O queue and waiting queue.
3. After that, another process gets into the running state, or we can select a new process from the ready state, which is to be executed, or the process has a high priority to execute its task.
4. Now, we have to update the PCB (Process Control Block) for the selected process P2. It includes switching the process state from ready to running state or from another state like blocked, exit, or suspend.
5. If the CPU already executes process P2, we need to get the status of process P2 to resume its execution at the same time point where the system interrupt occurs.

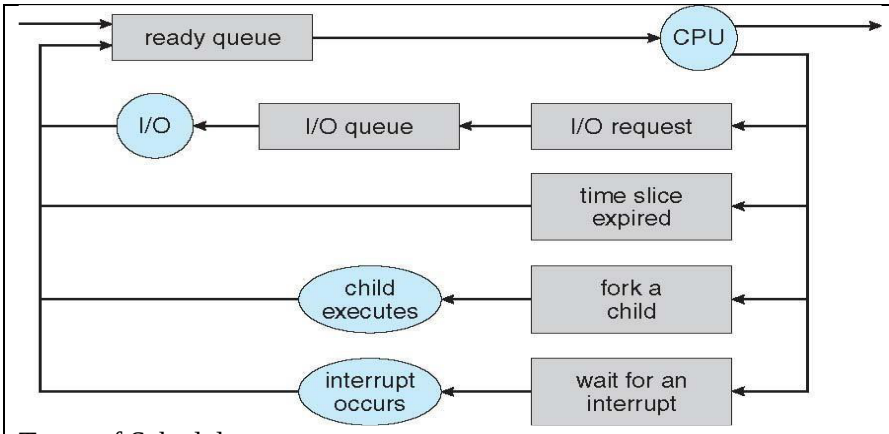
calculate the context switch time between two running processes.:

If all the processes' total execution time was T, then the context switch time = T – (SUM for all processes (waiting time + execution time)).

4 List and explain services provided by an operating system that are designed to make using computer systems more convenient for users. **<u>Explanation for each service: 1 Mark (Totally 9 Marks)</u>** **<u>User point of view: 1 mark</u>** OS provide services for the users of the system, including:	10	CO1	L2

• **User Interfaces** - Means by which users can issue commands to the system. Depending on the operating system these may be a command-line interface (e.g. sh, csh, ksh, tcsh, etc.), a Graphical User Interface (e.g. Windows, X-Windows, KDE, Gnome, etc.), or a batch command systems. **Program Execution** - The OS must be able to load a program into RAM, run the program, and terminate the program, either normally or abnormally. **I/O Operations** - The OS is responsible for transferring data to and from I/O devices, including keyboards, terminals, printers, and files. For specific devices, special functions are provided (device drivers) by OS. **File-System Manipulation** – Programs need to read and write files or directories. The services required to create or delete files, search for a file, list the contents of a file and change the file permissions are provided by OS. **Communications** - Inter-process communications, IPC, either between processes running on the same processor, or between processes running on separate processors or separate machines. May be implemented by using the service of OS- like shared memory or message passing. **Error Detection** - Both hardware and software errors must be detected and handled appropriately by the OS. Errors may occur in the CPU and memory hardware (such as power failure and memory error), in I/O devices (such as a parity error on tape, a connection failure on a network, or lack of paper in the printer), and in the user program (such as an arithmetic overflow, an attempt to access an illegal memory location). OS provide services for the efficient operation of the system, including. **Resource Allocation** – Resources like CPU cycles, main memory, storage space, and I/O devices must be allocated to multiple users and multiple jobs at the same time. **Accounting** – There are services in OS to keep track of system activity and resource usage, either for billing purposes or for statistical record keeping that can be used to optimize future performance. **Protection and Security** – The owners of information (file) in multiuser or networked computer system may want to control the use of that information.			
5 a. Differentiate between multiprogramming, multiprocessing, and multitasking systems with examples. **<u>Explanation for each system: 1 Mark (Totally 3 Marks)</u>** **<u>Diagram: 2 Marks</u>** Multiprogramming One of the most important aspects of operating systems is the ability to multiprogram. A single user cannot keep either the CPU or the I/O devices busy at all times. Multiprogramming increases CPU utilization by organizing jobs, so that the CPU always has one to execute.  Multitasking In **Time sharing** (or **multitasking) systems,** a single CPU executes multiple jobs by switching among them, but the switches occur so frequently that the users can interact	5	CO1	L3

with each program while it is running. The user feels that all the programs are being executed at the same time.  Multiprocessing Multiprocessor operating systems are used in operating systems to boost the performance of multiple CPUs within a single computer system. Multiple CPUs are linked together so that a job can be divided and executed more quickly.			
5 b. What are virtual machines? Explain with block diagram the benefits of using virtual machines. **Explanation: 2.5 Marks** **Diagram: 2.5 Marks** The fundamental idea behind a virtual machine is to abstract the hardware of a single computer (the CPU, memory, disk drives, network interface cards, and so forth) into several different execution environments. Thereby creating the illusion that each separate execution environment is running its own private computer.  	5	CO1	L2
6 Is process scheduling is necessary? Discuss the three types of schedulers with a neat diagram and explain the difference between them with respect to process state. **Process scheduling Definition: 2 Marks** **Explanation: 5 Marks** **Diagram: 3 Marks** As processes enter the system, they are put into a job queue, which consists of all processes in the system. • The processes that are residing in main memory and are ready and waiting to execute are kept on a list called the ready queue. This queue is generally stored as a linked list	10	CO1	L3



Types of Schedulers

- A long-term scheduler or Job scheduler
- The short-term scheduler, or CPU Scheduler
- The medium-term scheduler

A **long-term scheduler or Job scheduler** – selects jobs from the job pool (of secondary memory, disk) and loads them into the memory.

If more processes are submitted, than that can be executed immediately, such processes will be in secondary memory. It runs infrequently, and can take time to select the next process.

• **The short-term scheduler, or CPU Scheduler** – selects job from memory and assigns the CPU to it. It must select the new process for CPU frequently.

• **The medium-term scheduler** - selects the process in ready queue and reintroduced into the memory.

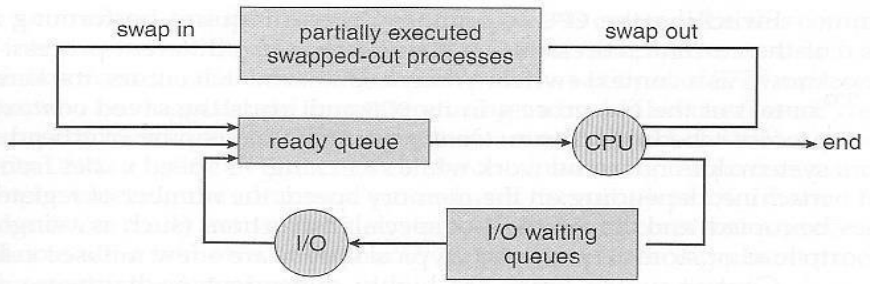


Figure 3.8 Addition of medium-term scheduling to the queueing diagram.