WINTER– 18 EXAMINATION

Subject Name: Operating System **Model Answer** Subject Code: 17512

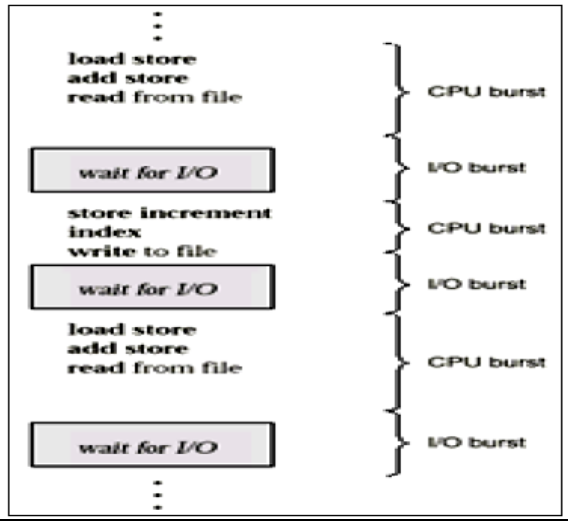
Important Instructions to examiners:

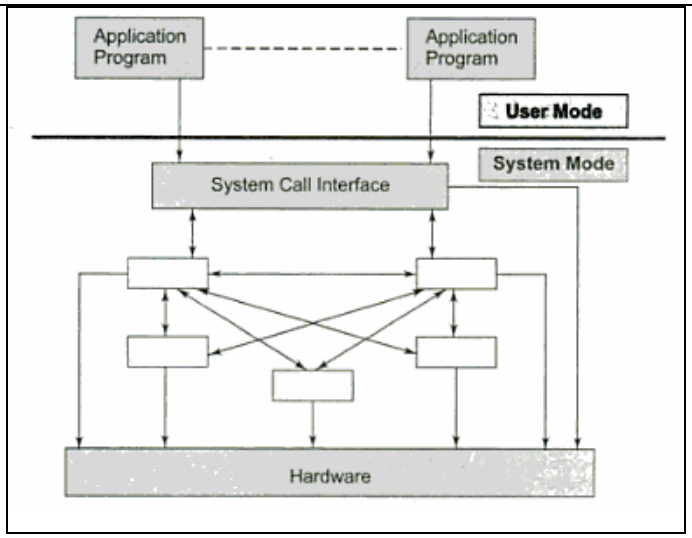
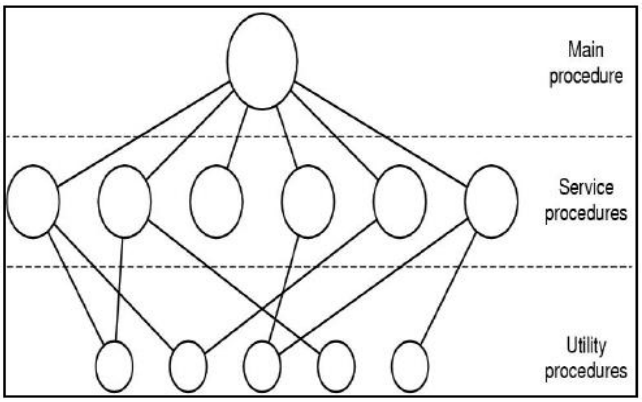
1) The answers should be examined by key words and not as word-to-word as given in the model answer scheme.
2) The model answer and the answer written by candidate may vary but the examiner may try to assess the understanding level of the candidate.
3) The language errors such as grammatical, spelling errors should not be given more Importance (Not applicable for subject English and Communication Skills.
4) While assessing figures, examiner may give credit for principal components indicated in the figure. The figures drawn by candidate and model answer may vary. The examiner may give credit for any equivalent figure drawn.
5) Credits may be given step wise for numerical problems. In some cases, the assumed constant values may vary and there may be some difference in the candidate's answers and model answer.
6) In case of some questions credit may be given by judgement on part of examiner of relevant answer based on candidate's understanding.
7) For programming language papers, credit may be given to any other program based on equivalent concept.

Q. No.	Sub Q. N.	Answer	Marking Scheme
1.		**Attempt any TEN of the following**	**20M**
	a.	**State type of file access method?**	2M
	Ans:	File access methods are: **Sequential Access:** Information in the file is processed in order, one record after the other. **Direct Access:** A file is made up of fixed length logical records that allow programs to read and write records rapidly in no particular order.	1 M for each method
	b.	**State any four criteria in CPU scheduling?**	2M
	Ans:	**CPU utilization: -** In multiprogramming the main objective is to keep CPU as busy as possible. CPU utilization can range from 0 to 100 percent. **Throughput: -** It is the number of processes that are completed per unit time. **Turnaround time:-**The time interval from the time of submission of a process to the time of completion of that process is called as turnaround time. It is calculated as: **Turnaround Time = Waiting Time + Burst Time or End Time – Arrival Time** **Waiting time: -** It is the sum of time periods spent in the ready queue by a process. It is calculated as: **Waiting Time = Start Time – Arrival Time** **Response time:-**The time period from the submission of a request until the first response is produced is called as response time.	1/2 M for each criteria, Any four
	c.	**Define UNIX operating system.**	2M
	Ans:	Unix is a portable, multitasking, multiuser, time sharing operating system originally developed in 1969 by a group of employees at AT & T.	2 M for correct definition

		OR The Unix operating system is a set of programs that act as a link between the computer and the user. The computer programs that allocate the system resources and coordinate all the details of the computer's internals is called the **operating system** or the **kernel**. Users communicate with the kernel through a program known as the **shell**. The shell is a command line interpreter; it translates commands entered by the user and converts them into a language that is understood by the kernel	
	d.	**What is system call?**	2M
	Ans:	System call is an interface between a running program and operating system. It allows user to access services provided by operating system. This system calls are procedures written using C, C++ and assembly language instructions. Each operating system has its own name for each system call. Each system call is associated with a number that identifies itself.	2 M for correct definition
	e.	**Define deadlock.**	2M
	Ans:	A deadlock consists of a set of blocked processes, each holding a resource and waiting to acquire a resource held by another process in the set	2 M for correct definition
	f.	**List any four services provided by operating system.**	2M
	Ans:	**Services provided by operating system are:** 1. Program execution 2. I/O operation 3. File manipulation (File systems) 4. Error detection 5. Communication 6. Resource allocation 7. Job accounting 8. Protection and security	1 M each for service, Any four
	g.	**Define process.**	2M
	Ans:	A process is a program in execution. Process is also called as job, task or unit of work. The execution of a process must progress in a sequential fashion. Process is an active entity.	2 M for correct definition
	h.	**What is the concept of paging?**	2M
	Ans:	Paging refers to the transfer of memory pages from physical memory to disk and vice versa. Virtual memory uses a technique called demand paging for its implementation. Logical address	2 M for correct

		space of a process can be noncontiguous; process is allocated physical memory whenever the latter is available.	definition

	i.	**Describe CPU and I/O burst cycle with suitable diagram.**	2 M
	Ans:	**CPU burst cycle:** It is a time period when process is busy with CPU. **I/O burst cycle:** It is a time period when process is busy in working with I/O resources. 	1 M for definition and 1 M for diagram
	j.	**List different multithreading models.**	2 M
	Ans:	Different multithreading models are: 1. one-to-one 2. one-to-many 3. many-to-many	1 M each, Any two
	k.	**Draw the diagram of monolithic structure of operating system.**	2 M

	Ans:	 **OR** 	2M for correct diagram
	l.	**List any two condition leading to process suspension.**	2M
	Ans:	**The conditions leading to process suspension are:** 1. Insufficient memory space in the system. 2. Running process stops for some correction.	1 M each condition
	m.	**What is booting process?**	2M
	Ans:	Booting is a bootstrapping process that starts operating system when user turns on the computer system. Bootstrapping is the process by which the computer system starts working.	2 M for correct definition
	n.	**List different file allocation method.**	2M
	Ans:	**File allocation methods are:** 1. Contiguous allocation method 2. Linked (Chained) allocation method	1 M for each type, Any two

3. Indexed allocation method.

2. **Attempt any FOUR of the following** **16M**

a **Differentiate between multitasking and multiprogramming.** 4M

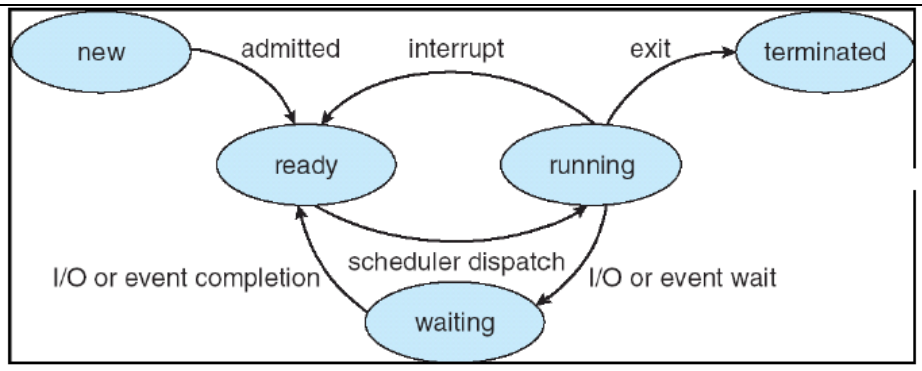
Ans:

Sr No.	Multitasking	Multiprogramming
1	The process of executing multiple numbers of tasks simultaneously or concurrently is known as multitasking.	The process of executing multiple numbers of programs simultaneously or concurrently is known as multiprogramming.
2	Task is smallest unit of operation.	Program is larger unit of operation.
3	Tasks do not contain programs.	Program may contain multiple numbers of tasks.
4	User interaction is provided.	There is no user interaction to individual program.
5	In multitasking there are two types of labels, i.e. Foreground and Background.	In multiprogramming there are only programs.
6	It utilizes CPU efficiently.	It utilizes CPU as well as I/O devices efficiently.

1 M for each point, Any four

b **Draw process state diagram and describe each state.** 4M

Ans:



2 M for diagram and 2 M for explanation

		New: The process being created is available in the new state. It is the new state because the system is not permitted it to enter the ready state due to limited memory available in the ready queue. If some memory becomes available, then the process from the new state will go to ready state. **Ready State:** The process which is not waiting for any external event such as I/O operation and which is not running is said to be in ready state. It is not in the running state because some other process is already running. It is waiting for its turn to go to the running state. **Running State:** The process which is currently running and has control of the CPU is known as the process in running state. In single user system, there is only one process which is in the running state. In multiuser system, there are multiple processes which are in the running state. **Blocked State:** The process that is currently waiting for external event such as an I/O operation is said to be in blocked state. After the completion of I/O operation, the process from blocked state enters in the ready state and from the ready state when the process turn will come it will again go to running state. **Terminated State:** The process whose operation is completed, it will go the terminated state from the running state. In halted state, the memory occupied by the process is released.	
	c	**Explain any four file related system calls.**	4M
	Ans:	**System calls Related to File management:** **Create file:** This system call is used to create a new file in any application. **Delete file:** This system call is used to delete a specific file from the given directory. **Open file:** This system call is used to open an existing file. **Close file:** This system call is used to close an already opened file. **Read file:** This system call is used to read a file which is already opened. By using this system call any information which is desired can be read from the file. **Write file:** This system call is used to write some data to a file. **Reposition file:** It means to change the position of a file pointer within the file. If there are many numbers of pages in a file then we can move forward or backward to any position within that file. **Get file attribute:** A file has certain attributes like file_name, file_type, file_size, date of creation,	1 M for each system call, Any four

		modification, access permission etc. By using this system call we can check any of the attributes of a particular file. **Set file attribute:** With the help of this system call, any of the attributes which can be modified or changed can be "SET".	
	d	**Describe any four condition for deadlock.**	4M
	Ans:	**1. Mutual exclusion:** Only one process at a time can use non-sharable resource. **2. Hold and wait:** A process is holding at least one resource and is waiting to acquire additional resources held by other processes. **3. No pre-emption:** A resource can be released only voluntarily by the process holding it after that process completes its task. **4. Circular wait:** There exists a set {P0, P1, …, P0} of waiting processes such that P0 is waiting for a resource that is held by P1, P1 is waiting for a resource that is held by P2, …, Pn–1 is waiting for a resource that is held by Pn, and Pn is waiting for a resource that is held by P0.	1 M for each condition
	e	**Explain the features of UNIX operating system.**	4M
	Ans:	**Features of UNIX are:** **1. Multiuser capability:** In multiuser system, the same computer resources that is hard disk, memory etc. is accessible to many users. The users are given separate terminals from where they can operate. All terminals are connected to the main computer whose resources are shared by all users. Such a step is economical, when a data needs to be shared among multiple users. **2. Multitasking capability:** UNIX supports multitasking; it means that it can do multiple tasks at the same time. It uses the technique of background and foreground processes to achieve multitasking. For example: A user is listening to the music and also typing a program when the music is playing. The CPU time is shared among multiple processes. **3. Communication:** UNIX has excellent support for networking because of which the users of UNIX can easily exchange information even when they are geographically apart. **4. Security:** UNIX has three provisions for protecting data. First can be assigning login ID and password. Second can be by granting permissions to the file. Third is files can be encrypted. **5. Portability:** UNIX can be ported to almost any computer architecture because it is coded in 'C'.	1 M for each features, Any four

	f	**Describe working of sequential and direct access method.**	4M
	Ans:	 **Sequential Access:** Information in the file is processed in order, one record after the other. This is by far the most common mode of access of files. For example, computer editors usually access files in this fashion. A read operation reads the next portion of the file and automatically advances the file pointer. Similarly, a write appends to the end of the file and the file pointer. Similarly, a write appends to the end of the file and the file pointer. Similarly, a write appends to the end of the end of the file and advances to the end of the newly written material (the new end of file). Such a file can be reset to the beginning, and, on some systems, a program may be able to skip forward or backward n records, for some integer n. This scheme is known as sequential access to a file. Sequential access is based on a tape model of a file. **Direct Access:** A file is made up of fixed-length logical records that allow programs to read and write records rapidly in no particular order. Thus, we may read block 14, then read block 53, and then write block 7. There are no restrictions on the order of reading or writing for a direct-access file. The direct-access method is based on a disk model of a file, since disks allow random access to any file block. Direct-access files are of great use for immediate access to large amounts of information. Databases are often of this type. For the direct-access method, the file operations must be modified to include the block number as a parameter. The block number provided by the user to the OS is normally a relative block number.	2 M for sequential access and 2 M for direct access
3		**Attempt any FOUR of the following:**	**16 M**
	a	**Describe real time system and state any two example of its application.**	4M
	Ans:	Real time system has well defined fixed time constraints. Processing should be done within the Defined constraints. A primary objective of real-time systems is to provide quick event response time and thus meet the scheduling deadlines. User convenience and resource utilization are of secondary concern to real-time system designers. In Real time systems, processor is allocated to the highest priority process among those that are ready to execute. Higher priority processes preempt execution of the lower priority processes. This form is called as 'priority –based preemptive scheduling'.	Description: 2 marks, Any two example: 1 mark each

The primary functions of the real time operating system are to:

1. Manage the processor and other system resources to meet the requirements of an application.
2. Synchronize with and respond to the system events.
3. Move the data efficiently among processes and to perform coordination among these processes.

Types of real time system:

1. Hard real time:-

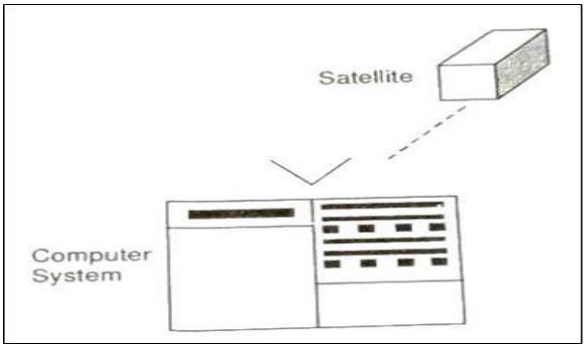
Hard real time means strict about adherence to each task deadline. When an event occurs, it should be serviced within the predictable time at all times in a given hard real time system. **Example: -**video transmission, each picture frame and audio must be transferred at fixed rate.

2. Soft real time:-

Soft real time means that only the precedence and sequence for the task operations are defined, interrupt latencies and context switching latencies are small. There can be few deviations between expected latencies of the tasks and observed time constraints and a few deadline misses are accepted.

Example: - Mobile phone, digital cameras and orchestra playing robots.

Satellite Application of real time OS: Satellite application of real time OS The satellite connected to the computer system sends the digital samples at the rate of 1000 samples per second. The computer system has an application program that stores these samples in a file. The sample sent by the satellite arrives every millisecond to the application. So computer must store or respond the sample in less than 1 millisecond. If the computer does not respond to the sample within this time, the sample will lost.



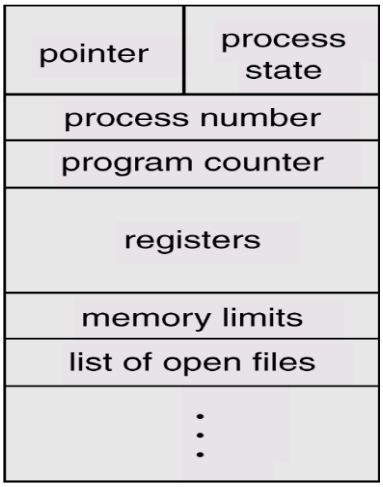
Applications:

1. Flight Control System
2. Simulations
3. Industrial control
4. Military applications

	b.	**What is process management? State any four functions of process management.**	4 M
	Ans:	**Process Management:** The operating system manages many kinds of activities ranging from user programs to System programs like printer spooler, name servers, file server etc. Each of these activities is encapsulated in a process. A process includes the complete execution context (code, data, PC, registers, OS resources in use etc.). The major activities/functions of an operating system in regard to process management are: 1. Creation and deletion of user and system processes. 2. Suspension and resumption of processes. 3. A mechanism for process synchronization. 4. A mechanism for process communication. 5. A mechanism for deadlock handling.	Description: 2 marks, Any four functions: ½ mark each
	c	**Draw and explain process control block in detail**	4M
	Ans:	Each process is represented as a process control block (PCB) in the operating system. It contains information associated with specific process. In general, a PCB may contain information regarding: 1. **Process Number:** Each process is identified by its process number, called process identification number (PID). 2. **Priority:** Each process is assigned a certain level of priority that corresponds to the relative importance of the event that it services. 3. **Process State:** This information is about the current state of the process. I.e. whether process is in new, ready, running, waiting or terminated state. 4. **Program Counter:** This contains the address of the next instruction to be executed for this process. 5. **CPU Registers:** CPU registers vary in number and type, depending upon the computer architectures. These include index registers, stack pointers and general purpose registers etc. When an interrupt occurred, information about the current status of the old process is saved in registers along with the program counters. This information is necessary to allow the process to be continued correctly after the completion of an interrupted process. 6. **CPU Scheduling Information:** This information includes a process priority, pointers to	Diagram: 2 marks, Description: 2 marks

scheduling queues and any other scheduling parameters.

7. **Memory Management Information:** This information may include such information as the value of base and limit registers, the page table or the segment table depending upon the memory system used by operating system.
8. **Accounting:** This includes actual CPU time used in executing a process in order to charge individual user for processor time.
9. **I/O Status:** It includes outstanding I/O request, allocated devices information, pending operation and so on.
10. **File Management:** It includes information about all open files, access rights etc.



	d.	**Explain the pre-emptive and non-preemptive type of scheduling.**	4M
	Ans:	**Pre-emptive Scheduling**:-Even if CPU is allocated to one process, CPU can be preempted to other process if other process is having higher priority or some other fulfilling criteria. • Throughput is less • Only the processes having higher priority are scheduled. • It doesn't treat all processes as equal. • Algorithm design is complex. Circumstances for preemptive • Process switch from running to ready state • Process switch from waiting to ready state **For e.g.: Round Robin, Priority algorithms** **Non-Preemptive Scheduling** Once the CPU has been allocated to a process the process keeps the CPU until it releases CPU either by terminating or by switching to waiting state.	Preempti-ve Scheduli-ng: 2 marks, Non preempti-ve: 2 marks

- Throughput is high.
- It is not suitable for RTS.
- Processes having any priority can get scheduled.
- It treats all process as equal.
- Algorithm design is simple.

Circumstances for Non preemptive

- Process switches from running to waiting state
- Process terminates

For e.g.: FCFS algorithm ~~It is suitable for RTS.~~, SJF (non-preemtive)

e. **Give the comparison between UNIX and LINUX operating system. (Any four points).** 4M

Ans: Any four points: 1 mark each

	LINUX	**UNIX**
What is it?	Linux is an example of Open Source software development and Free Operating System (OS).	Unix is an operating system that is very popular in universities, companies, big enterprises etc.
Cost	Linux can be freely distributed, downloaded freely, distributed through magazines, Books etc. There are priced versions for Linux also, but they are normally cheaper than Windows.	Different flavors of Unix have different cost structures according to vendors
User	Everyone. From home users to developers and computer enthusiasts alike.	Unix operating systems were developed mainly for mainframes, servers and workstations except OSX, Which is designed for everyone. The Unix environment and the client-server program model were essential elements in the development of the Internet
Manufacturer	Linux kernel is developed by the community. Linus Torvalds oversees things.	Three bigest distributions are Solaris (Oracle), AIX (IBM) & HP-UX Hewlett Packard. And Apple Makes OSX, an unix based os..
Usage	Linux can be installed on a wide variety of computer hardware, ranging from mobile phones, tablet computers and video game consoles, to mainframes	The UNIX operating system is used in internet servers, workstations & PCs. Backbone of the majority of finance infastructure and many 24x365

	and supercomputers.	high availability solutions.
Development and Distribution	Linux is developed by Open Source development i.e. through sharing and collaboration of code and features through forums etc and it is distributed by various vendors.	Unix systems are divided into various other flavors, mostly developed by AT&T as well as various commercial vendors and non-profit organizations.
GUI	Linux typically provides two GUIs, KDE and Gnome. But there are millions of alternatives such as LXDE, Xfce, Unity, Mate, twm, ect.	Initially Unix was a command based OS, but later a GUI was created called Common Desktop Environment. Most distributions now ship with Gnome.
File system support	Ext2, Ext3, Ext4, Jfs, ReiserFS, Xfs, Btrfs, FAT, FAT32, NTFS	jfs, gpfs, hfs, hfs+, ufs, xfs, zfs format
Text mode interface	BASH (Bourne Again SHell) is the Linux default shell. It can support multiple command interpreters.	Originally the Bourne Shell. Now it's compatible with many others including BASH, Korn & C.
Price	Free but support is available for a price.	Some free for development use (Solaris) but support is available for a price.
Security	Linux has had about 60-100 viruses listed till date. None of them actively is spreading nowadays.	A rough estimate of UNIX viruses is between 85 -120 viruses reported till date.
Threat detection and solution	In case of Linux, threat detection and solution is very fast, as Linux is mainly community driven and whenever any Linux user posts any kind of threat, several developers start working on it from different parts of the world	Because of the proprietary nature of the original Unix, users have to wait for a while, to get the proper bug fixing patch. But these are not as common.
Processors	Dozens of different kinds.	x86/x64, Sparc, Power, Itanium, PA-RISC, PowerPC and many others.
Examples	Ubuntu, Fedora, Red Hat, Debian, Archlinux, Android etc.	OS X, Solaris, All Linux
Architectures	Originally developed for Intel's x86 hardware, ports available for over two dozen CPU types including ARM	is available on PA-RISC and Itanium machines. Solaris also available for x86/x64 based systems. OSX is PowerPC(10.0-10.5)/x86(10.4)/x64(10.5-10.8)

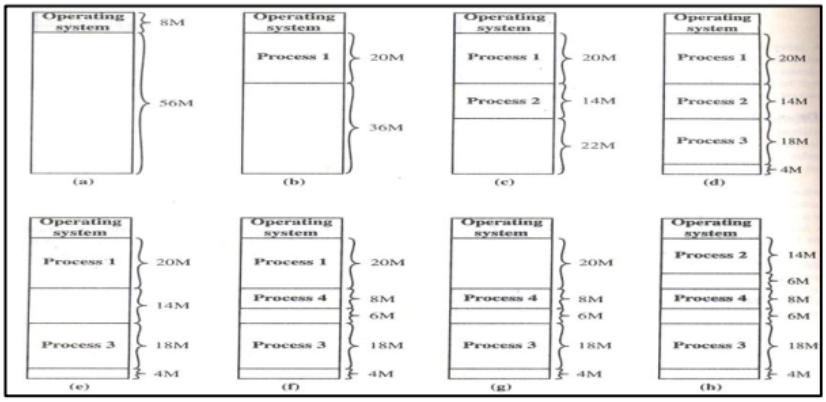
			Inception	Inspired by MINIX (a Unix-like system) and eventually after adding many features of GUI, Drivers etc, Linus Torvalds developed the framework of the OS that became LINUX in 1992. The LINUX kernel was released on 17th September, 1991	In 1969, it was developed by a group of AT&T employees at Bell Labs and Dennis Ritchie. It was written in "C" language and was designed to be a portable, multi-tasking and multi-user system in a time-sharing configuration	
	f.	**Explain static and dynamic memory partitioning method.**				4M
	Ans:	**Static Memory Partitioning:** Main memory is divided into multiple partitions of fixed size at the time of system generation. A process may be loaded into a partition of equal size or greater size. Partitions can be of equal size or unequal size Operating System 8 M 8 M 8 M 8 M 8 M 8 M 8 M 8 M **Advantages:** • Simple to implement • It requires minimal operating system software and processing overhead as partitions are fixed at the time of system generation. **Disadvantages:** • Memory wastage • Inefficient use of memory due to internal fragmentation. • Maximum number of active processes is fixed. **Dynamic Memory partitioning:** When a process enters in main memory, it is allocated exact size that is required by that process. So in this method, partitions can vary in size depending on memory space required by a process entering in main memory. Operating system maintains a table indicating which parts of memory are available and which are occupied. When new process arrives and it needs space, system searches for available memory space in mainmemory. If it is available then memory is allocated to the process by creating a partition in memory. Like this depending on size of process and available memory, partitions take place in main				Explanat-ion of Static Partition-ing: 2 marks, Explanat-ion of Dynamic Partition-ing: 2 marks

memory.

For example:-Consider following table with process and memory space.

Process	Memory space
P1	20 M
P2	14 M
P3	18 M
P4	8 M
P5	10 M

Process of memory allocation:-



Total memory size is 64M.from this 8M partition is occupied by operating system and remaining can be partitioned as per the size of the process

Advantages:

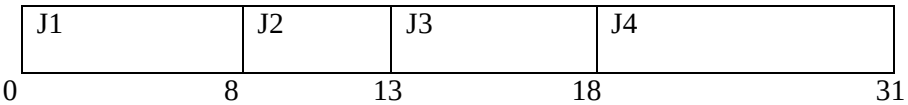
- No internal fragmentation.
- More efficient use of main memory.

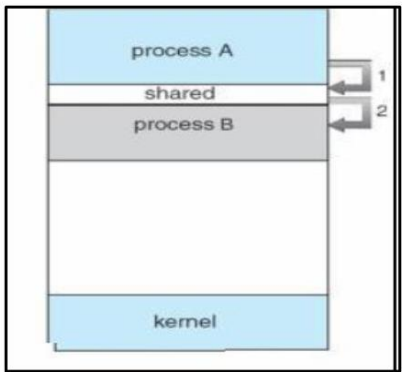
Disadvantages:

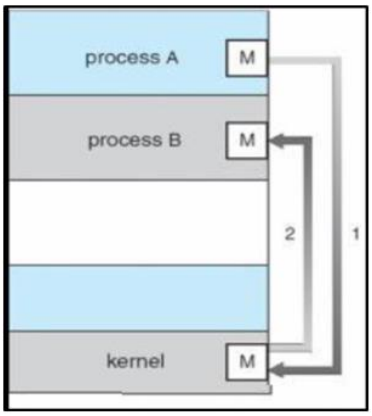
- It suffers from external fragmentation.
- It needs compaction

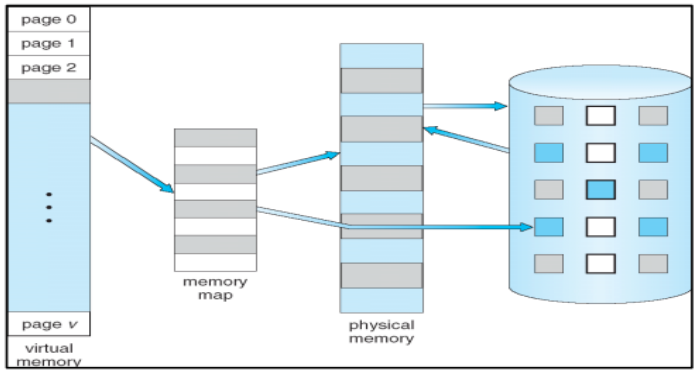
4		**Attempt any FOUR of the following:**	**16M**
	a.	**Describe evolution of operating system.**	4M
	Ans:	**Generations of operating system** 1. The 1940's - First Generations 2. The 1950's - Second Generation 3. The 1960's - Third Generation 4. The 1980's-The Fourth Generation **First generation 1945 – 1955 - vacuum tubes, plug boards:** The earliest electronic digital computers had no operating systems. Machines of the time were so	{**Note - marks shall be given for generations or types of operating system**}

		primitive that programs were often entered one bit at time on rows of mechanical switches (plug boards). Programming languages were unknown (not even assembly languages). **The 1950's - Second Generation:** Second generation 1955 – 1965 - transistors, batch systems. By the early 1950's, the routine had improved somewhat with the introduction of punch cards. The General Motors Research Laboratories implemented the first operating systems in early 1950's for their IBM 701. The system of the 50's generally ran one job at a time. These were called single-stream batch processing systems because programs and data were submitted in groups or batches. **The 1960's - Third Generation:** Third generation 1965 – 1980 - ICs and multiprogramming. The systems of the 1960's were also batch processing systems, but they were able to take better advantage of the computer's resources by running several jobs at once. So operating systems designers developed the concept of multiprogramming in which several jobs are in main memory at once; a processor is switched from job to job as needed to keep several jobs advancing while keeping the peripheral devices in use. **The Fourth Generation Fourth generation 1980:** Present personal computers with the development of LSI (Large Scale Integration) circuits, chips, operating system entered in the system entered in the personal computer and the workstation age. Microprocessor technology evolved to the point that it becomes possible to build desktop computers as powerful as the mainframes of the 1970s. **OR** Description of batch, Multi programmed Multitasking, Timesharing, Desktop, Distributed Systems, Clustered System, Real Time system. **Batch Systems:** Main function of a batch processing system is to automatically keep executing the jobs in a batch. **Multiprogramming:** It executes multiple programs simultaneously by a single processor. **Multitasking:** Multitasking is a logical extension of multiprogramming. Multiple jobs are executed by the CPU switching between them, but the switches occur so frequently that the users may interact with each program while it is running. **Time-Sharing Systems**–Interactive Computing: In time sharing system, the CPU executes multiple jobs by switching among them. **Desktop Systems: Personal computers** – computer system dedicated to a single user. **Distributed system:** Distributed system or distributed data processing is the system in which processors, data and other aspects of a data processing system may be dispersed within on organization. **Clustered system:** It is a group of connected computers working together as one unit.	Explanation of four generations: 1 mark each

		Real Time system: A Real Time system is used when there are rigid time requirement on the operation of a processor or the flow of data and thus is often used as a control device in a dedicated application.	
	b.	**Explain memory management in detail.**	**4M**
	Ans:	**Main-Memory Management:** Primary-Memory or Main-Memory is a large array of words or bytes. Each word or byte has its own address. Main-memory provides storage that can be access directly by the CPU. That is to say for a program to be executed, it must in the main memory. The major activities of an operating in regard to memory-management are: • Keep track of which part of memory are currently being used and by whom. • Decide which process are loaded into memory when memory space becomes available • Allocate and De-allocate memory space as needed. **Secondary-Storage Management:** Systems have several levels of storage, including primary storage, secondary storage and cache storage. Instructions and data must be placed in primary storage or cache to be referenced by a running program. Because main memory is too small to accommodate all data and programs, and its data are lost when power is lost, the computer system must provide secondary storage to back up main memory. Secondary storage consists of tapes, Disks, and other media designed to hold information that will eventually be accessed in primary storage (primary, secondary, cache) is ordinarily divided into bytes or words consisting of a fixed number of bytes. Each location in storage has an address; the set of all addresses available to a program is called an address space. The three major activities of an operating system in regard to secondary storage management are: • Managing the free space available on the secondary-storage device • Allocation of storage space when new files have to be written. • Scheduling the requests for memory access.	Correct Explanation: 4 marks
	c.	**Consider the following four jobs.** Find average waiting time for (i)FCFS (ii)SJF Job | Burst Time J1 | 8 J2 | 5 J3 | 5 J4 | 13	4M
	Ans:	**(i) FCFS**  	For each scheduling, Gantt chart: 1 mark;

		` **Waiting Time for each process:** J1=0 J2=8 J3=13 J4=18 **Average waiting time**= (0+8+13+18)/4=39/4=9.75 **(ii) SJF** J2 | J3 | J1 | J4 0 5 10 18 31 **Waiting Time for each process:** J1=10 J2=0 J3=5 J4=18 **Average waiting time**= (10+0+5+18)/4=33/4=8.25	waiting time for each process calculation : ½ marks; average Waiting time: ½ mark
	d.	**Draw and explain inter-process communication model**	4M
	Ans:	**Inter-process communication:** Cooperating processes require an Inter-process communication (IPC) mechanism that will allow them to exchange data and information. There are two models of IPC: **1. Shared memory**  In this a region of the memory residing in an address space of a process creating a shared memory segment can be accessed by all processes who want to communicate with other processes. All the processes using the shared memory segment should attach to the address space of the shared memory. All the processes can exchange information by reading and/or	For each model, diagram: 1 mark; explanation: 1 mark

		writing data in shared memory segment. The form of data and location are determined by these processes who want to communicate with each other. These processes are not under the control of the operating system. The processes are also responsible for ensuring that they are not writing to the same location simultaneously. After establishing shared memory segment, all accesses to the shared memory segment are treated as routine memory access and without assistance of kernel. 2. **Message Passing:** In this model, communication takes place by exchanging messages between cooperating processes. It allows processes to communicate and synchronize their action without sharing the same address space. It is particularly useful in a distributed environment when communication process may reside on a different computer connected by a network. Communication requires sending and receiving messages through the kernel. The processes that want to communicate with each other must have a communication link between them. Between each pair of processes exactly one communication link.  	
	e.	**Describe virtual memory management.**	4M
	Ans:	Virtual memory is the separation of user logical memory from physical memory. This separation allows an extremely large virtual memory to be provided for programmers when only a smaller physical memory is available. Virtual memory makes the task of programming much easier, because the programmer no longer needs to worry about the amount of physical memory available, or about what code can be placed in overlays, but can concentrate instead on the problem to be programmed. It is the process of increasing the apparent size of a computer's RAM by using a section of the hard disk storage as an extension of RAM. As computers have RAM of capacity 64 or 128 MB to be used by the CPU resources which is not sufficient to run all applications that are used by most users in their expected way and all at once.	Correct Explanation: 4 marks

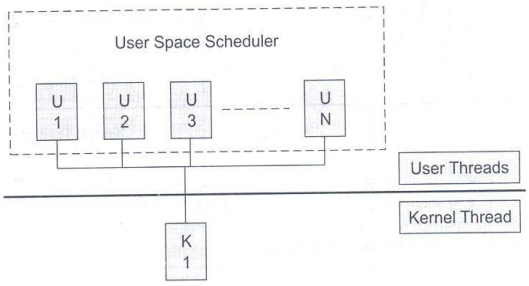


Example:

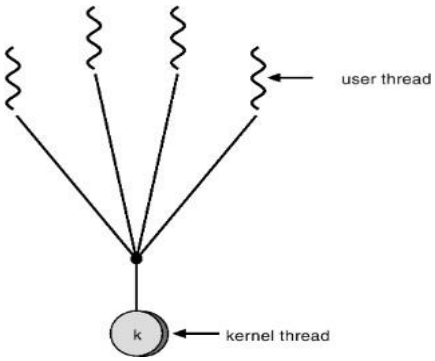
For example, an e-mail program, a web browser and a word processor is loaded into RAM simultaneously; the 64 MB space is not enough to store all these programs. Without a virtual memory, a message "You cannot load any more applications. Please close an application to load a new one." would be displayed. By using a virtual memory, a computer can look for empty areas of RAM which is not being used currently and copies them on to the hard disk device. Thus RAM is freed to load new applications.

Actually it is done automatically, the user do not even know that it is happening, and the user feels like RAM has unlimited space even though the RAM capacity is 32 MB. It is a process of increasing computer's RAM by using a section of the hard disk storage as an extension of RAM.

	f.	**Explain multithreading model with diagram.**	4M
	Ans:	Multithreading models:- 1. Many-to-One 2. One-to-One 3. Many-to-Many **1. Many-to-One:** - This model maps many user level threads to one kernel level thread. Thread management is done by thread library in user space. **Advantages:-** • It is an efficient model as threads are managed by thread library in user space. **Disadvantages:-** • Only one thread can access the kernel at a time, so multiple threads are unable to run in parallel on microprocessor. • If a thread makes a blocking system call then the entire process will be block. Example: - Green threads – a thread library available for Solaris use many-to-one model.	List: 1 mark; Correct Explanation of each model: ½ mark each; Diagram of each model: ½ mark



OR



2. One-to-One: It maps each user level thread to a kernel level thread. Even one thread makes a blocking call; other thread can run with the kernel thread.

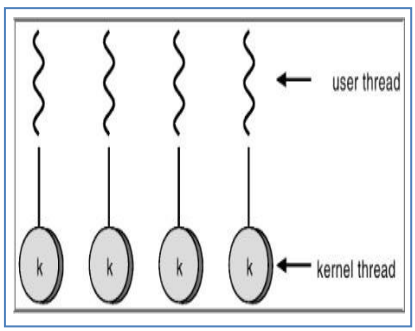
Advantages:-

- It allows multiple threads to run in parallel on multiprocessors.

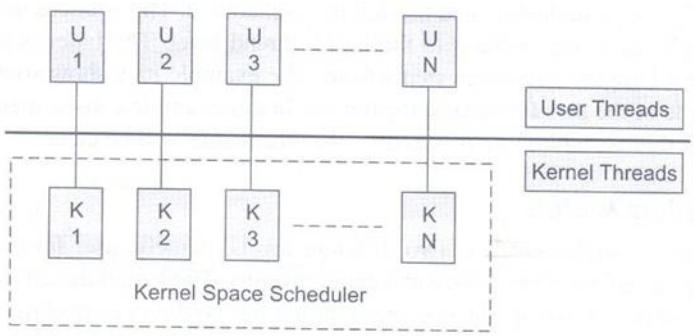
Disadvantages:-

- Creating a user thread requires creating the corresponding kernel thread. Creating kernel thread may affect the performance of an application.

Example: - Linux, Windows OS including Win 95, 98, NT 2000, and XP implement the one-to-one model.

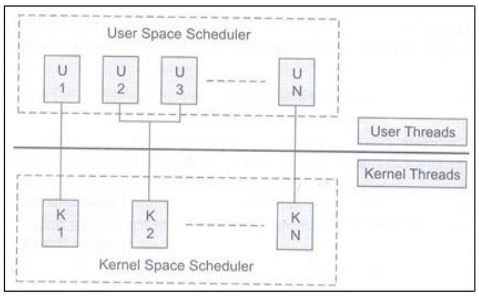


OR

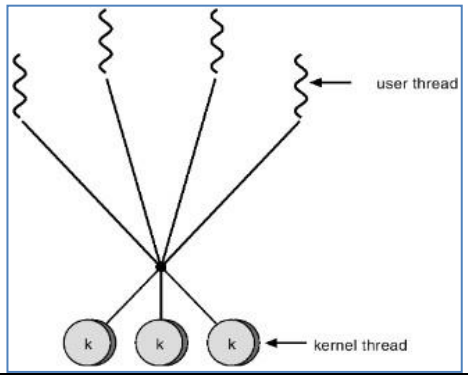


3. Many-to-Many:

This model maps many user level threads to a smaller or equal number of kernel threads. Number of kernel threads may be specific to either a particular application or particular machine.



OR



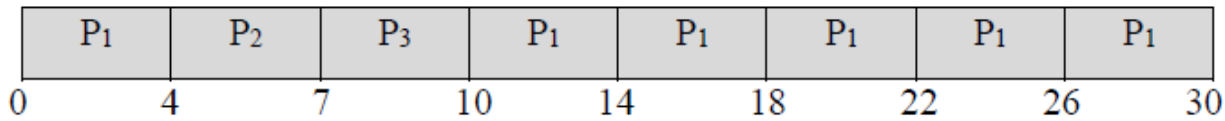
5		**Attempt any FOUR of the following**	**16 M**
	a	**Describe Round Robin Algorithm with suitable example**	4 M
	Ans:	• The Round–Robin (RR) scheduling algorithm is designed especially for time sharing systems. • It is similar to FCFS scheduling, but preemption is added to enable the system to switch between processes. • A small unit of time, called as time quantum or time slice, is defined. A time quantum is generally from 10 to 100 milliseconds in length.	Description: 2 Marks, Example: 2 marks

- The ready queue is treated as a circular queue. The CPU scheduler goes around the ready queue, allocating the CPU to each process for a time interval of up to 1 time quantum.
- To implement RR scheduling, we keep the ready queue as a FIFO queue of processes. New processes are added to the tail of the ready queue. The CPU scheduler picks the first process from the ready queue, sets a timer to interrupt after 1 time quantum, and dispatches the process.
- One of two things will then happen. The process may have a CPU burst of less than 1 time quantum. In this case, the process itself will release the CPU voluntarily. The scheduler will then process to the next process in the ready queue. Otherwise, if the CPU burst of the currently running process is longer than 1 time quantum, the timer will go off and will cause an interrupt to the operating system. A context switch will be executed, and the process will be put at the tail of the ready queue. The CPU scheduler will then select the next process in the ready queue.
- The average waiting time under the RR policy is often long.
- Consider the following set of processes that arrive at time 0, with the length of the CPU burst given in milliseconds:

Process	Burst Time
P1	24
P2	3
P3	3

If we use a time quantum of 4 milliseconds, then process P1 gets the first 4 milliseconds. Since it requires another 20 milliseconds, it is preempted after the first time quantum. And the CPU is given to the next process in the queue, process P2. process P2 does not need 4 milliseconds, so it quits before its time quantum expires. The CPU is then given to the next process, process P3. Once each process has received 1 time quantum, the CPU is returns to process P1 for an additional time quantum.

The resulting RR schedule is as follows:



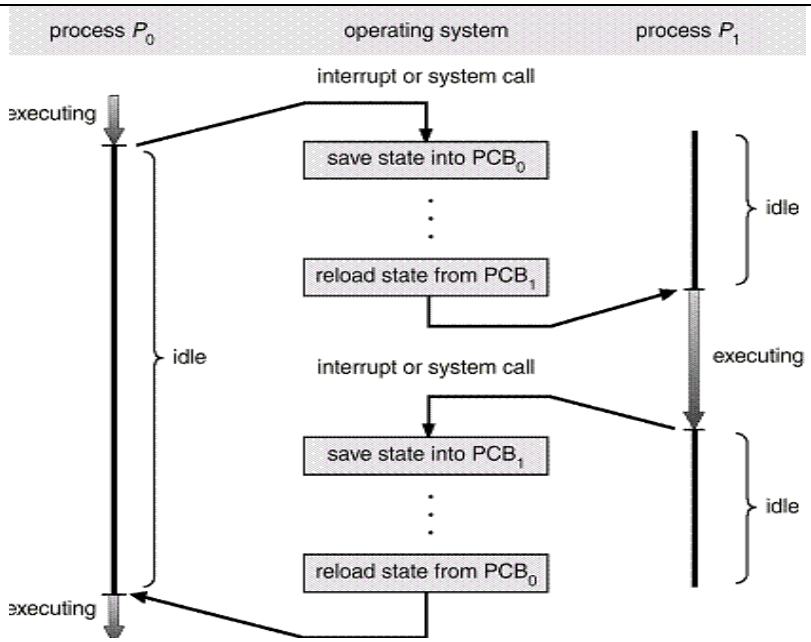
b **Explain secondary Storage Management** 4 M

Ans:

- Systems have several levels of storage, including primary storage, secondary storage and cache storage.
- Instructions and data must be placed in primary storage or cache to be referenced by a running program. Because main memory is too small to accommodate all data and programs, and its data

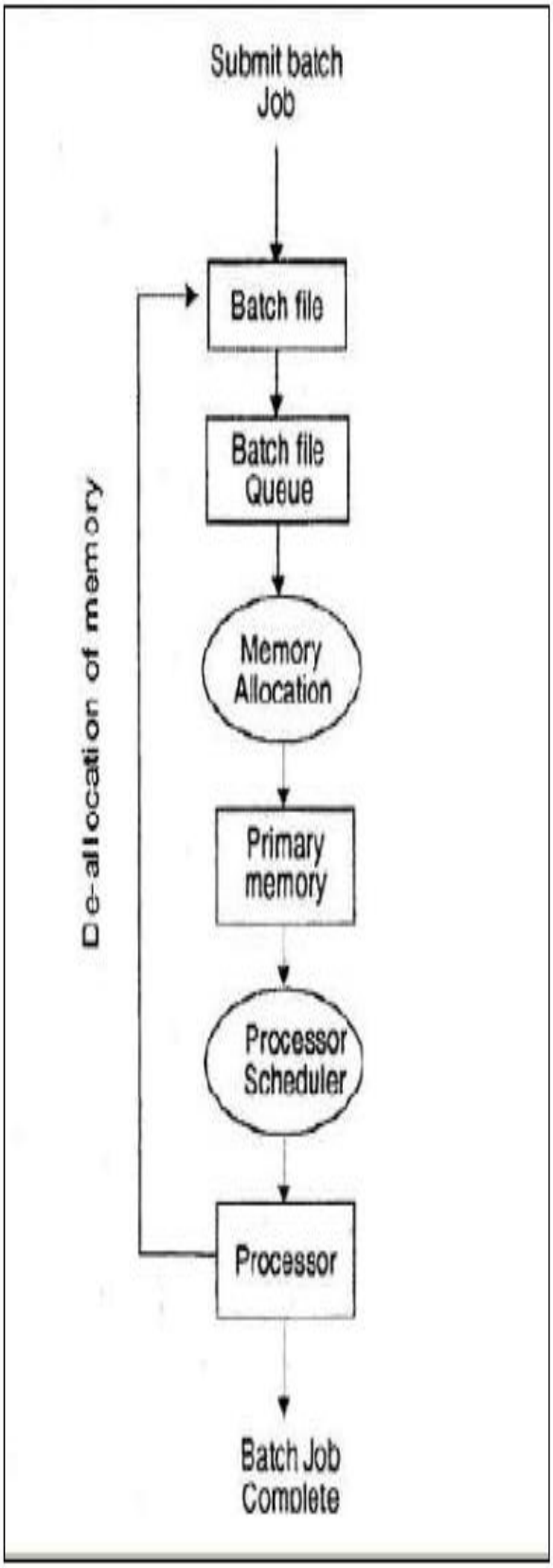
Any relevant explanation: 4 marks

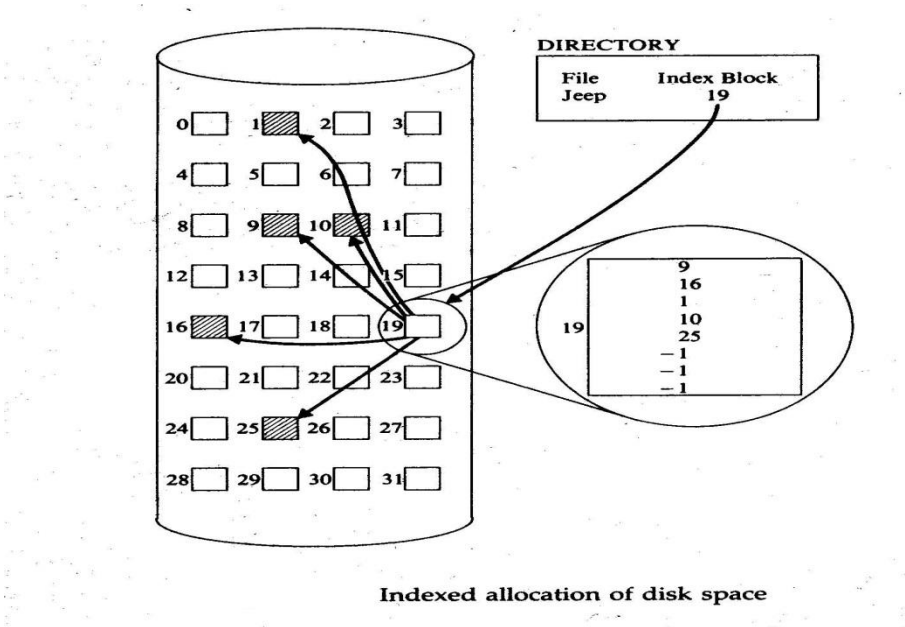
		are lost when power is lost, the computer system must provide secondary storage to back up main memory. • Secondary storage consists of tapes, disks, and other media designed to hold information that will eventually be accessed in primary storage (primary, secondary, cache) is ordinarily divided into bytes or words consisting of a fixed number of bytes. • Each location in storage has an address; the set of all addresses available to a program is called an address space. Activities of an operating system in regard to secondary storage management are: • Managing the free space available on the secondary-storage device • Allocation of storage space when new files have to be written. • Scheduling the requests for memory access.	
	c	**How context switching is done?**	4 M
		• Switching the CPU to another process requires saving the state of current process and • loading the saved state for new process. This process is known as a context switch. • The context of a process is represented in the PCB of the process, it includes the value of the CPU registers the process state and memory management information. • When a context switch occurs, the kernel saves the context of the old process in its PCB and loads the saved context of the new process scheduled to run. • The context switching time is an overhead time. During switching time system does not do any useful work. Context switch times are highly dependent on hardware support. Its speed varies from machine to machine, depending on the memory speed, the number of registers that must be copied and the existence of special instructions. Typically, the speed ranges from 1 to 1000 microseconds. Context-switch times are highly dependent on hardware support. • Some hardware systems employ two or more sets of processor registers to reduce the amount of context switching time. When the process is switched, the following information is stored. o Program Counter o Scheduling Information o Base and limit register value o Currently used register o Changed State o I/O State o Accounting	Description: 2 marks, Diagram: 2 marks

			
	d	**Explain Batch Monitoring functions.**	4 M
	Ans:	• A batch operating system normally reads a stream of separate jobs (from a card reader. For example,), each with its own control cards that predefine to prevent errors and improper use of the computer. It is concerned with the operation and control if I/O devices. • A batch system is one in which jobs are bundled together with the instruction necessary to allow them to be processed without intervention. Often jobs of a similar nature can be bundled together to further increase economy. • Common input devices were card readers and tape drives. The basic physical layout of the memory of batch job computer is shown in fig. • The OS was simple; its major task was to transfer control from one job to the next. The job was submitted to the computer operator in form of punch cards. At some later time, the output appeared. • The OS was always resident in memory. Often magnetic tapes and drums were used to store intermediate data and compiled programs. • Example: Payroll system, stock control and billing systems.	Explanation:2 marks, Diagram 2 marks


operating system
user program area

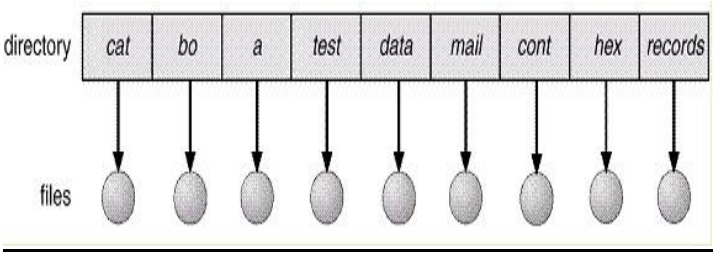
OR

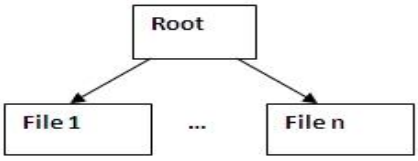

Submit batch Job
Batch file
Batch file Queue
Memory Allocation
Primary memory
Processor Scheduler
Processor
Batch Job Complete
De-allocation of memory

	e	**Describe indexed allocation method with advantages and disadvantages**	4 M
	Ans:	In this method, each file has its own index block. This index block is an array of disk block addresses. When a file is created, an index block and other disk blocks according to the file size are allocated to that file. Pointer to each allocated block is stored in the index block of that file. Directory entry contains file name and address of index block. When any block is allocated to the file, its address is updated in the index block. Any free disk block can be allocated to the file. Each ith entry in the index block points to the itch block of the file. To find and read the ith block, we use the pointer in the ith index block entry. It supports direct access without suffering from external fragmentation. **Advantages of Indexed Allocation**: • Does not suffers from external fragmentation • Support both sequential and direct access to the file. • Indexing of free space can be done by mean of the bit map. • Entire block is available for data as no space is occupied by pointers. **Disadvantages of Indexed Allocation:** • It required lot of space for keeping pointers. Wasted space. • Storing many addresses i.e. pointers becomes an overhead • Indexed allocation is more complex and time consuming.  Indexed allocation of disk space	Description: 01, Advantages: 01, Disadvantages: 01, Diagram: 01
	f	**Explain single level directory structure.**	4 M
	Ans:	• It is the simplest form of directory system is having one directory containing all the files, and each file must have a unique name. Sometimes it is called the **root directory.**	Explanation: 2

- A single-level directory has significant limitations, however, when the number of files increases or when there is more than one user.
- Software design is simple.
- Easy to support and understand. The advantages of this scheme are its simplicity and the ability to locate files quickly - there is only one place to look, after all.
- Since all files are in the same directory, they must have unique names. If there are two users who call their data file "test", then the unique-name rule is violated. Even with a single-user, as the number of files increases, it becomes difficult to remember the names of all the files in order to create only files with unique names

marks, Diagram: 2 marks



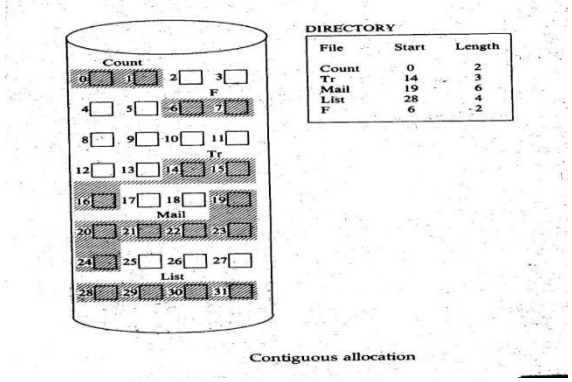
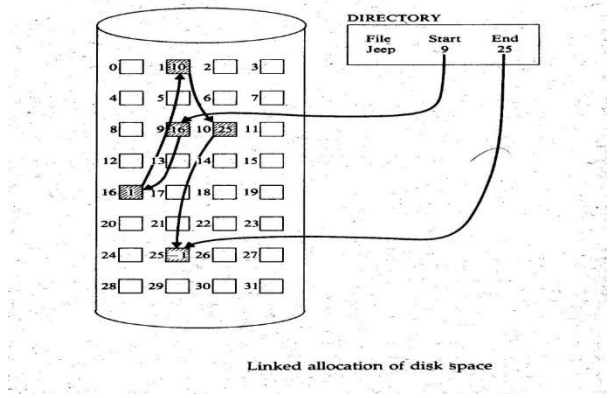


6		**Attempt any four of the following**	**16 M**
	a	**Differentiate between contiguous and linked memory allocation method**	4 M

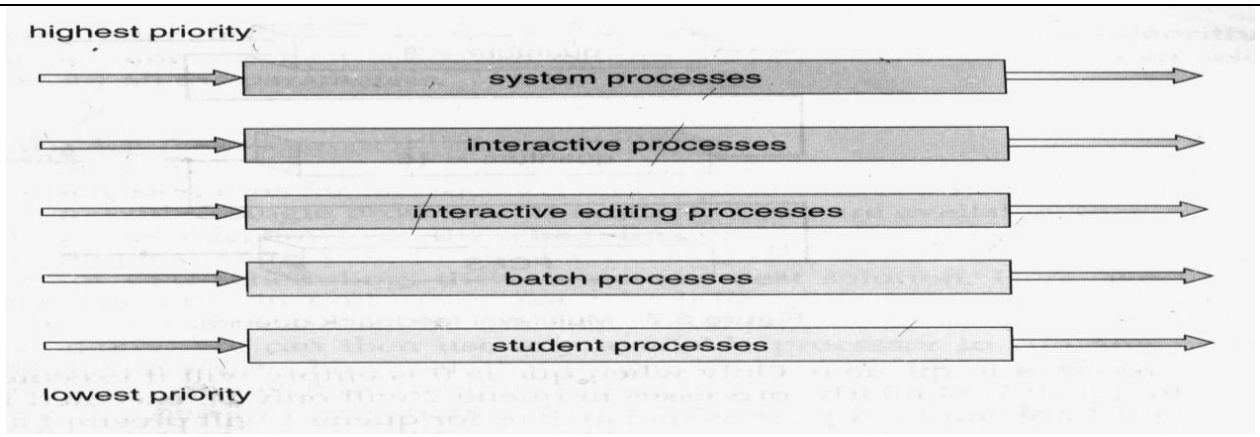
Ans:

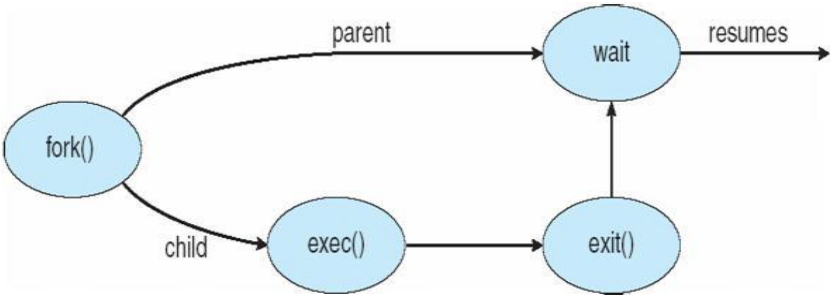
Contiguous memory allocation	Linked memory allocation
Contiguous allocation requires that each file occupy a set of contiguous blocks on the disk.	Data structures are connected by a series of nodes.
Directory contains filename, starting block number and offset	Directory contains filename, pointer to starting block and ending block
Suffers from external fragmentation	There is no external fragmentation
Very difficult to find contiguous blocks of space	Any free blocks can be added to a chain.
Support direct access	Cannot support direct access
Size of file declared at start	Size of file need not be declared at start.
Insertions and deletions are difficult.	Insertions and deletions can be done easily.

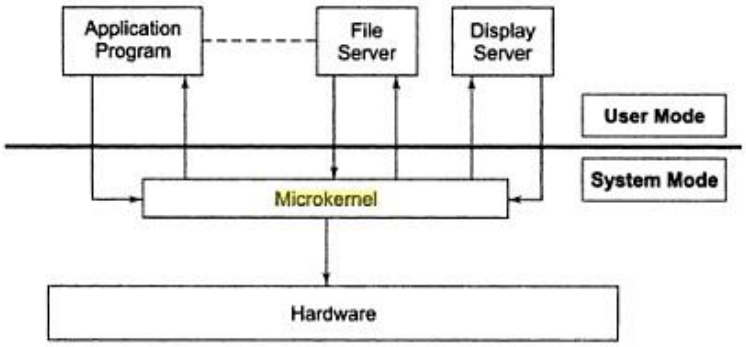
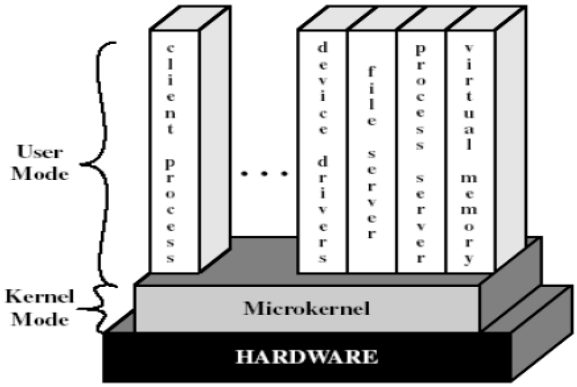
Any four points: one marks for each point

		Pointers not used.  Contiguous allocation	Memory space required for pointer  Linked allocation of disk space
	b	**Write steps for Bankers algorithm to avoid deadlock**	4 M
	Ans:	**Banker's Algorithm:** This algorithm calculates resources allocated, required and available before allocating resources to any process to avoid deadlock. It contains two matrices on a dynamic basis. Matrix A contains resources allocated to different processes at a given time. Matrix B maintains the resources which are still required by different processes at the same time. **Algorithm F: Free resources** **Step 1:** When a process requests for a resource, the OS allocates it on a trial basis. **Step 2:** After trial allocation, the OS updates all the matrices and vectors. This updating can be done by the OS in a separate work area in the memory. **Step 3:** It compares F vector with each row of matrix B on a vector to vector basis. **Step 4:** If F is smaller than each of the row in Matrix B i.e. even if all free resources are allocated to any process in Matrix B and not a single process can completes its task then OS concludes that the system is in unstable state. **Step 5:** If F is greater than any row for a process in Matrix B the OS allocates all required resources for that process on a trial basis. It assumes that after completion of process, it will release all the recourses allocated to it. These resources can be added to the free vector. **Step 6:** After execution of a process, it removes the row indicating executed process from both matrices. **Step 7:** This algorithm will repeat the procedure step 3 for each process from the matrices and finds that all processes can complete execution without entering unsafe state. For each request for any resource by a process OS goes through all these trials of imaginary allocation and updation. After this if the system remains in the safe state, and then changes can be made in actual matrices.	Any relevant steps: 4 m
	c	**Explain with suitable example how semaphore help to overcome critical section problem**	4 M

	Ans:	Semaphore is a synchronization tool. A semaphore S is an integer variable which is initialized and accessed by only two standard operations: wait () and signal ().All the modifications to the integer value of semaphore in wait () and signal () operations can be done only by one process at a time. Working of semaphore to solve synchronization problem:- Consider two concurrently running processes P1 and P2.P1 contains statement S1 and P2 contains statement S2.When we want to execute statement S2 only after execution of statement S1, then we can implement it by sharing a common semaphore synch between two processes. Semaphore synch is initialized to 0.to execute the sequence modify code for process P1 and P2. **Process P1 contains:** S1; signal (synch); **Process P2 contains:-** wait (synch); S2; As synch is initialized to 0, Process P2 will wait and process P1 will execute. Once process P1 completes execution of statement S1, it performs signal () operation that increments synch value. Then wait () operation checks the incremented value and starts execution of statement S2 from Process P2.	Explanation: 2 marks, Example: 2 marks
	d	**Explain multilevel queue scheduling**	4 M
	Ans:	Multilevel queue scheduling classifies processes into different groups. It partitions the ready queue into several separate queues. The processes are permanently assigned to one queue based on some properties such as memory size, priority, process type, etc. Each queue has its own scheduling algorithm. In a system there are foreground processes and background processes. So system can divide processes into two queues: one for background and other for foreground. Foreground queue can be scheduled with Round Robin algorithm where as background queue can be scheduled by First Come First Serve algorithm. Scheduling is done for all the processes inside the queue as well as for all separate queues. **Example:** Consider all the processes in the system are divided into four groups: system, interactive, interactive editing, batch and student processes queue. Each queue contains processes. CPU is first scheduled for all queues on may be priority, total burst time or process type. (Any relevant diagram shall be considered)	Explanation: 2 marks, Diagram: 2 marks



	e	**Why is process creation necessary? State the role of fork process in the context.**	4 M
	Ans:	**Process creation:** When a new process is to be added to those currently being managed, the operating system builds the data structures that are used to manage the process and allocates address space in main memory to the process. This is the creation of a new process. Create Process Operating system creates a new process with the specified or default attributes and identifier. A process may create several new sub processes. Syntax for creating new process is: CREATE (processed, attributes) Process creation in UNIX and Linux are done through fork() system calls. When the operating system creates a process at the explicit request of another process, the action is referred to as process spawning. When one process spawns another, the former is referred to as the parent process, and the spawned process is referred to as the child process. The parent may have to partition its resources among its children, or it may be able to share some resources among several of its children. A sub-process may be able to obtain its resources directly from the operating system. exec system call used after a fork to replace the process' memory space with a new program  	Explanation:2 marks; Role of fork: 2 marks
	f	**Describe microkernel structure of operating system**	4 M
	Ans:	In this system, kernel provides only the most essential operating system functions like process management, communication primitives and low level memory management. System programs and user	Explanation: 2

		level programs implemented outside the kernel, provides the remaining operating system services. These programs are known as servers. Due to separation of functionality of kernel, size of the kernel is reduced. This reduced kernel is called as microkernel. The application programs and various servers communicate with each other using messages that passed through microkernel. The microkernel validates the messages and passes them between the various modules of the operating system and permits access to the hardware.  **OR**  	marks, Diagram: 2 marks