MAHARASHTRA STATE BOARD OF TECHNICAL EDUCATION
(Autonomous)
(ISO/IEC – 27001 – 2005 Certified)

WINTER – 2012 EXAMINATION

Subject Code:12178 **Model Answer** **Page No.: 1/32**

Subject Title:- Operating System **QP Code No.:- 12178**
Name of Course & Year/Sem.:- Computer Engineering Vth Semester

Q.No.	MODEL ANSWER	Marking Scheme	Remark
1. a)	Attempt any FIVE of the following: Multiprogramming 1. Multiprogramming is the simple form of parallel processing in which several programs are run at the same time on a processor. 2. Since there is only once processor, there can be no simultaneous execution of different programs. Instead the operating system executes part of one program, then the part of another and so on. 3. Multiprogramming needs to deal with multiple users and therefore they are more complex 4. This operating system needs job scheduling algorithms to choose the next job to execute among several jobs. Multitasking 1. Time sharing is a logical extension of multiprogramming. 2. In multitasking systems the CPU executes multiple jobs by switching among them, but the switching occurs so frequently that the users can interact with each program while it is running. 3. Multitasking systems were developed to provide interactive user of computer system. 4. It uses CPU scheduling and multiprogramming to provide each user with a small portion of a time sharing computer.	**(Four points 1 mark each**)	

Q.No.	MODEL ANSWER	Marking Scheme	Remark
b)	Services provided by operating System 1. Program Creation 2. Program Execution 3. Access to Input/output devices 4. Controlled access to files. 5. Controlled access to any resource 6. Error detection and recovery 7. Accounting. **Program Creation:** Support programs such as editors, translators are not a part of operating system. They use the system facilities in the same way as the user program. Thus services of support program are accessed through operating system. **Program Execution:** Initialization of I/O devices, loading data and instructions in main memory, are the tasks performed by the operating system.. **Access to I/O devices:** Each I/O device requires its own set of instructions and control signals for operations. The operating system takes care of converting the simple read and write instructions from the user into the instruction that the device will understand. **Controlled access o files:** In addition to the particular nature of I/O device (CD, DVD etc)it also deals with the file format of the storage medium. The operating system also provides a protection mechanism to control access to different user files. **Controlled access to any resource:** The operating system provides protection to resources from unauthorized users, resolve conflict arising out of different users claiming the same resource simultaneously. **Error detection and recovery:** For hardware errors such as memory errors, device failure or software errors such as arithmetic overflow, attempt to access the forbidden memory location, the operating system softens the error condition with the least impact on the running of computer system. The response from the user may be termination of the program that caused the error or reporting the error to the user.	List any four services 1 mark each , explain any one 3 marks	

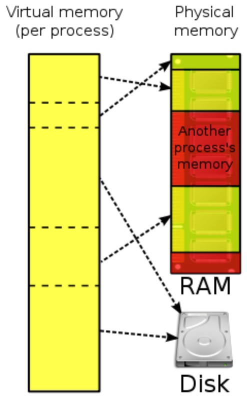
Q.No.	MODEL ANSWER	Marking Scheme	Remark
	Accounting: The operating system can keep the usage statistics for various resources and **Multithreading:** Refers to the ability to an O.S to support multiple threads of execution within a single process. In a multi threaded environment multiple processes and multiple threads can be considered as in case of multiuser O.S. such as UNIX.		
C)	A process is the unit of work in a system. In Process model, all software on the computer is organized into a number of sequential processes. A process includes PC, registers, and variables. Conceptually, each process has its own virtual CPU. In reality, the CPU switches back and forth among processes Process is not the same as program. A process is more than a program code. A process is an 'active' entity as oppose to program which consider to be a 'passive' entity. As we all know that a program is an algorithm expressed in some suitable notation, (e.g., programming language). Being a passive, a program is only a part of process. Process, on the other hand, includes: • Current value of Program Counter (PC) • Contents of the processors registers • Value of the variables • The process stack (SP) which typically contains temporary data such as subroutine parameter, return address, and temporary variables. • A data section that contains global variables.	Process explanation 2 marks and program 2 marks	
d)	FJFS **Waiting time** : the waiting time under FCFS vary if process CPU burst times vary greatly , SJF the waiting time is minimum because it moves short processes before the long processes. **Turnaround time**: large fluctuations in	All there compared 3 marks example(any) 1 mark	

<table>
<tr><th>Q.No.</th><th>MODEL ANSWER</th><th>Marking Scheme</th><th>Remark</th></tr>
<tr><td></td><td>average turnaround time are possible in FCFS systems. SJF always gives minimum turnaround Time
FCFS schedule:
Job 1: 24 units, Job 2: 3 units, Job 3: 3 units
| Job 1 | Job 2 | Job 3 |
0 24 27 30
Total waiting time: 0 + 24 + 27 = 51
Average waiting time: 51/3 = 17
Total turnaround time: 24 + 27 + 30 = 81
Average turnaround time: 81/3 = 27

SJF schedule:
Job 1: 24 units, Job 2: 3 units, Job 3: 3 units
| Job 2 | Job 3 | Job 1 |
0 3 6 30
Total waiting time: 6 + 0 + 3 = 9
Average waiting time: 3
Total turnaround time: 30 + 3 + 6 = 39
Average turnaround time: 39/3 = 13
SJF always gives minimum waiting time and turnaround time
Throughput :for FCFS throughput is potentially low and for SJF throughput is potentially high.</td><td></td><td></td></tr>
<tr><td>e)</td><td>**Benefits of Multithreading**
Responsiveness: Multithreading allows a process to keep running even if some threads within the process are stalled, working on a lengthy task, or awaiting user interaction. Using a digital alarm clock as an example of a process, the thread of keeping track of time, continues while an alarm is sounding while another awaits it's time to activate.
Cost Effective: Memory and resource allocation to process creation remains costly where as threads share the resources allocated to the process they reside in making it less costly to make threads or move them from on process to another.
Resource Distribution: The inherit property of sharing memory and resources of the</td><td>Listing 1 marks , explanation 3 marks.</td><td></td></tr>
</table>

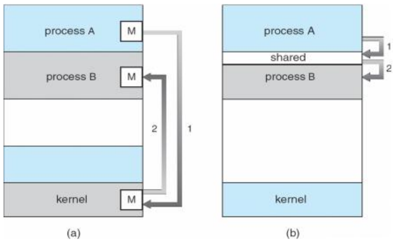
Q.No.	MODEL ANSWER	Marking Scheme	Remark
	parent process fosters the ability of having multiple treads occupying the same address space. **Cross-Processor Distribution:** The benefits of multithreading are multiplied as the number of available processors increase opposite to single threading where only one processor is used. In a multiprocessor architecture, running of threads can distribute across multiple processors in parallel thereby increasing efficiency.		
f)	Paging divides the computer's primary memory into fixed-size units called page frames, and the program's address space into pages of the same size. The hardware memory management unit maps pages to frames. The physical memory can be allocated on a page basis while the address space appears contiguous. Segmentation is the only memory management technique that does not provide the user's program with a 'linear and contiguous address space.".Segments are areas of memory that usually correspond to a logical grouping of information such as a code procedure or a data array. Segments require hardware support in the form of a segment table which usually contains the physical address of the segment in memory, its size, and other data such as access protection bits and status	Paging 2 marks and segmentation 2 marks	
g)	Swapping is a simple memory/process management technique used by the operating system(os) to increase the utilization of the processor by moving some blocked process from the main memory to the secondary memory(hard disk);thus forming a queue of temporarily suspended process and the execution continues with the newly arrived process.After performing the swapping process,the operating system has two options in selecting a process for	Explanation 3 marks , use 1 mark	

Q.No.	MODEL ANSWER	Marking Scheme	Remark
	execution. Assume a multiprogramming environment with Round-Robin CPU scheduling algorithm. When a quantum expires memory manager will start to swap out the process that just finished, and swap in another price to the memory space that has been freed. In the meantime, CPU scheduler will allocate a time slice to some other process in memory. when each process finishes its quantum it will be swapped back with another process. Ideally, memory manager can swap process fast enough so that there are always process in memory, ready to execute, when CPU scheduler wants to reschedule the CPU. The quantum must also be sufficiently large that reasonable amounts of computing are done between swaps. Swapping can be implemented in various ways. For example, swapping can be priority based.		
2. a)	**Attempt any FOUR of the Following:** Virtual Memory Virtual memory makes application programming easier by hiding fragmentation of physical memory; by delegating to the kernel the burden of managing the memory hierarchy (eliminating the need for the program to handle overlays explicitly); and, when each process is run in its own dedicated address space, by obviating the need to relocate program code or to access memory with relative addressing.	Explanation 3 marks diagram 1 mark.	

Q,No.	MODEL ANSWER	Marking Scheme	Remark
b)	 Virtual memory (per process) Physical memory Another process's memory RAM Disk **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 a. 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 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. b. 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	Explanation 3 marks diagram 1 mark.	

<table>
<tr><th>Q.No.</th><th>MODEL ANSWER</th><th>Marking Scheme</th><th>Remark</th></tr>
<tr><td></td><td>different computer connected by a network.

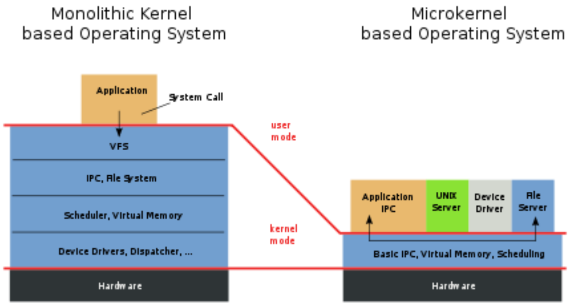
Communication requires sending and receiving messages through the kernel. The processes that want to communicate with each other must have a communication link between them. Between each pair of processes exactly one communication link



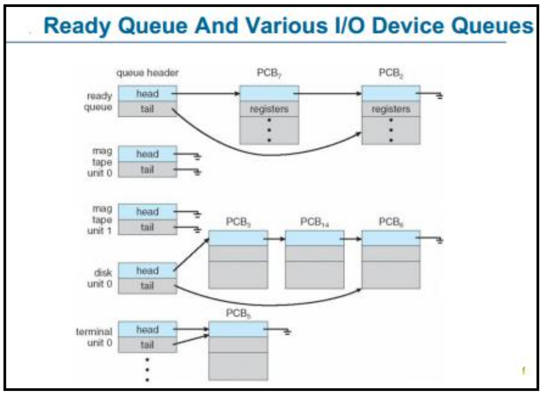
process A
M
process B
M
kernel
M
2
1
(a)
process A
shared
process B
kernel
1
2
(b)

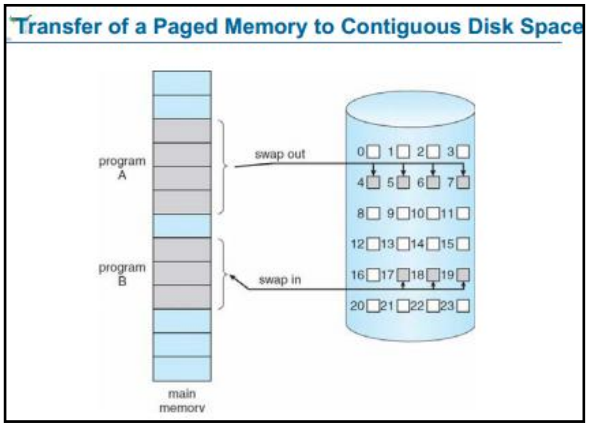
</td><td></td><td></td></tr>
<tr><td>c)</td><td>I/O system management
1. I/O system management hides the peculiarities of specifis hardware devices from the user . Only the device driver knows the peculiarities of the specific device to which it is assigned.

<table><tr><td>User level</td></tr><tr><td>Device Independent OS Software</td></tr><tr><td>Device drivers</td></tr><tr><td>Interrupt Handlers</td></tr><tr><td>Hardware</td></tr></table>
2. The diagram shows the users are provided with the user friendly I/O software through which they can access any device connected to the computer
3. For every device , a driver is available ,operating system provides the platform to install these device drivers.
4. Device drivers communicate directly with the hardware by using interrupt</td><td>Explanation 3 marks diagram 1 mark.</td><td></td></tr>
</table>

<table>
<tr><th>Q.No.</th><th>MODEL ANSWER</th><th>Marking Scheme</th><th>Remark</th></tr>
<tr><td></td><td>handlers.</td><td></td><td></td></tr>
<tr><td>d)</td><td>Clustered System: Cluster is a group of interconnected, whole computers working together as a unified computing source that can create the illusion of being one machine. Each computer in a cluster is typically referred to as a node.
<b>Clustering</b> (means gather together) allows two or more system to shear storage closely linked via a local area network .
Asymmetric Cluster (at least two servers: One is on a standby mode while the other is monitoring the other one. If one stops other will work).
Symmetric Cluster (all work with one: The work together and monitor each other).</td><td>Explanation 4 marks</td><td></td></tr>
<tr><td>e)</td><td>A <b>microkernel</b> (also known as μ-kernel) is the near-minimum amount of software that can provide the mechanisms needed to implement an operating system (OS). These mechanisms include low-level address space management, thread management, and inter-process communication (IPC). If the hardware provides multiple rings or CPU modes, the microkernel is the only software executing at the most privileged level (generally referred to assupervisor or kernel mode)

Monolithic Kernel based Operating System
Microkernel based Operating System
Application
System Call
VFS
IPC, File System
Scheduler, Virtual Memory
Device Drivers, Dispatcher, ...
Hardware
user mode
kernel mode
Application IPC
UNIX Server
Device Driver
File Server
Basic IPC, Virtual Memory, Scheduling
Hardware
</td><td>Explanation 4 marks.</td><td></td></tr>
<tr><td>f)</td><td><b>System Calls:</b> System calls are programming interface to the services provided by the operating system.
Implementation:</td><td>Explanation 4 marks</td><td></td></tr>
</table>

Q.No.	MODEL ANSWER	Marking Scheme	Remark
	1. Number to number the system calls, each system call associated with a particular number. 2. System call interface maintains a table indexed according to these numbers. 3. The system call interface invokes intended system call in operating system kernel & returns status of the system call and any return values. 4. The caller need to know nothing about how the system call is implemented. Just needs to obey API and understand what OS will do as a result call. 5. Most details of operating system interface hidden from programmers by API. It is managed by run-time support library. 6. System calls can be roughly grouped into the following major categories. a. Process or Job control b. File Management c. Device Management d. Information Maintenance System calls related to process control: End, Abort Load, Execute Create process, Terminate process Ready process, Dispatch process Suspend, Resume Get Process attribute, set attribute Wait for time Wait event, signal event System calls Related to File management: Create file, delete file Open file , Close file Create directory Read, write, Reposition Get file attribute , set file attribute Create a link Change the working directory System calls Related to Device Management: Request a device, Release a device Read, Write, Reposition Get device attribute, set device attribute System calls Related to Information Maintenance: Get Time or Date, Set Time or Date Get System data, Set system data Get process, file or device attributes Set process, file or Device attributes.		

Q.No.	MODEL ANSWER	Marking Scheme	Remark
3.	**Attempt any four the following**	**(4 marks)**	
a)	**Real time operating system** It is a special purpose operating system. A real time system is used when rigid time requirements have been placed on the operation of processor or the flow of data ,thus it is often used as a control device in a dedicated application. Fixed constraint Processing must be done in within defined constraint or system fail Hard real time os. Critical task to be completed within given time Soft real time o.s less stringent than hard real time os. Some extra time will be given to complete task	(Description of real time o.s.2 marks, hard & soft real time o.s 2 marks)	
b) i)	 Ready Queue And Various I/O Device Queues queue header PCB$_7$ PCB$_2$ ready queue head tail registers registers mag tape unit 0 head tail mag tape unit 1 head tail PCB$_3$ PCB$_{14}$ PCB$_6$ disk unit 0 head tail PCB$_5$ terminal unit 0 head tail	(Diagram 1 marks)	

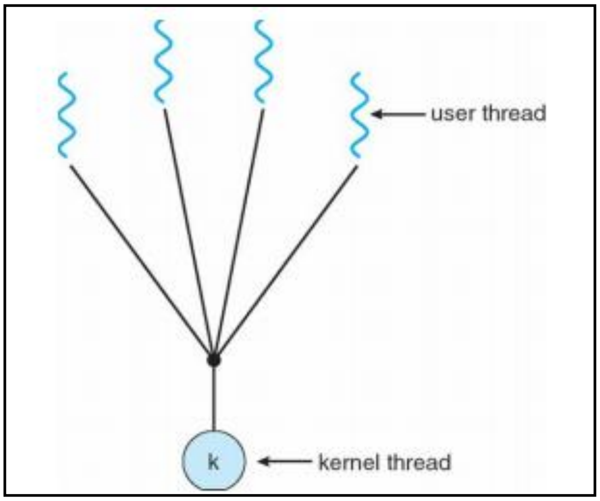
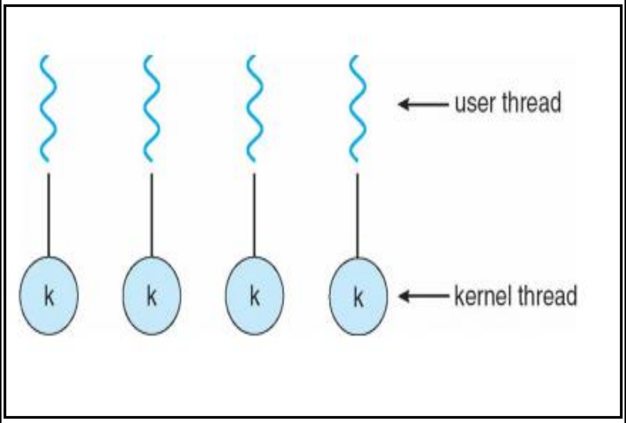
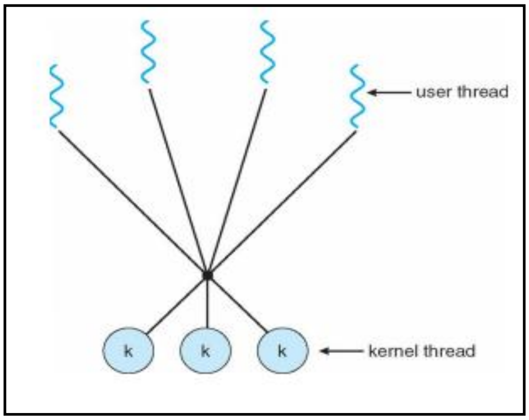
Q.No.	MODEL ANSWER	Marking Scheme	Remark
	scheduling queue - ready queue - job queue - device queue	Description of queues 1 Mark	
ii)	context switch - Switch the cpu to another process requires saving the state of old process and loading the saved state for new process. This time is known as a context switch. The context switch represented in PCB. - Saves context of old process in its PCB and loads context of new process schedule to run. - Pure overhead - Depend on hardware support	Any 4 correct points 2 Marks	
c)	A process requests resources , if resources are not available at that time , process enters a wait state. Waiting process may never again change state because resources they have requested are held by other waiting process. This situation is called deadlock. Necessary condition(1 points for each condition) Mutual exclusion Hold and wait No preemption Circular wait	Definition 1 marks, for 4 conditions 3 marks	
d)	 Transfer of a Paged Memory to Contiguous Disk Space program A swap out program B swap in main memory 0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 Demand Paging	Diagram 2 marks correct 4 points 2 marks	

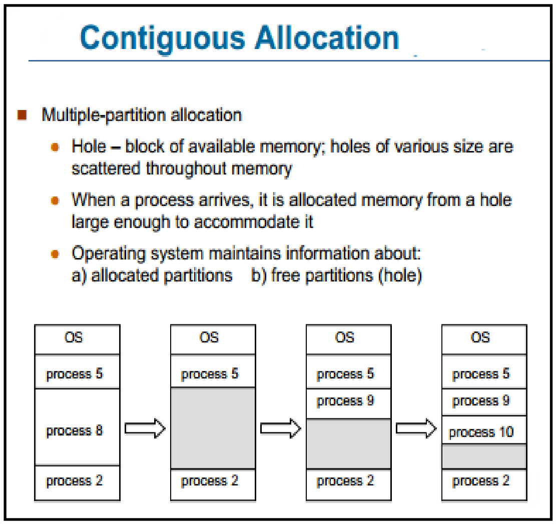
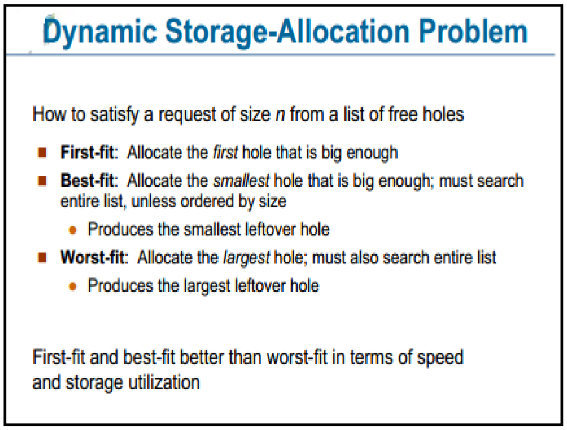
Q.No	MODEL ANSWER	Marking Scheme	Remark
	Similar to paging with swapping. Pages brought into main memory as per demand from disk storage Valid invalid bit Page table Secondary memory		
e)	A file is named for convenience of its human users and is referred to by its name **Name** symbolic file name, human readable **Identifier** unique tag, number identifies file in system **Type** information to support different types **Location** location of file on device **Size** file size **Protection** access control information **Time ,date and user identification** info for creation , last modification and last use	Any 4 attribute 4 Marks	
f)	i) Preemptive 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. Circumstances for preemptive • process switch from running to ready state • process switch from waiting to ready state incurs cost	Relevant 2 points carries 2 Marks	
	ii)Non preemptive scheduling Once the cpu ha been allocated to a process the process keeps the cpu until releases cpu either by terminating or by switching to waiting state circumstances for Non preemptive • When process switches from running to waiting state • When process terminates	2 Marks for 2 points	

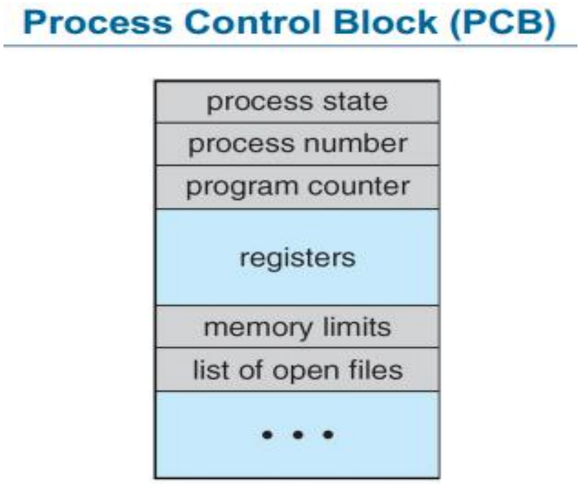
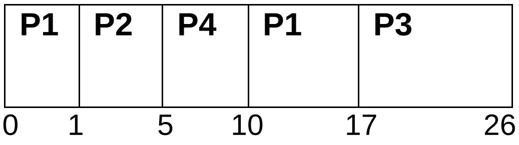
Q.No.	MODEL ANSWER	Marking Scheme	Remark
4. a)	Attempt any four of the following System components 1. Process management 2. Main memory management 3. File management 4. I/O system management 5. Secondary storage management 6. Networking 7. Protection system 8. Command interpreter 1. Process management • Creating and Deleting both user and system processes • Suspending and resuming processes • Providing mechanisms for process synchronization • Providing mechanisms for process communication • Providing mechanisms for process deadlock handling 2. Main memory management • Keeping track of which part of memory are currently being used and by whom • Deciding which processes are to be loaded in to memory when memory space becomes available • Allocating & De allocating space as needed 3. File management • Creating and Deleting files and directories • Mapping files onto secondary storage • Backing up files on stable storage media 4. IO system management • It is consist of memory management components, a general device driver interface and drivers for specific hardware	Listing components carries 1 mark For description 2 component carries 3 marks each carries 1 ½ marks	

Q.No.	MODEL ANSWER	Marking Scheme	Remark
	devices 5. Secondary storage management • Free space management • Storage allocation • Disk scheduling 6. Networking • The network may be fully or partially connected • Network design consider message routing and connection strategies and problem of contention and security		
b)	An operating system is a program that manages the computer hardware. It provides basic for application programs and act as an intermediary between a user of a computer and computer hardware. Any four features of Mobile Phone Operating System OS Features • Features – Multitasking – Scheduling – Memory Allocation – File System Interface – Keypad Interface – I/O Interface – Protection and Security – Multimedia features – • Design and capabilities of a Mobile OS (Operating System) is very different than a general purpose OS running on desktop machines: – mobile devices have constraints and restrictions on their physical characteristic such as screen size, memory, processing power and etc. – Scarce availability of battery	Definition 2 Marks & Features 2 Marks	Mobile Phone Operating system is out of syllabus. So any relevant answer should be given 2 Marks.

Q.No.	MODEL ANSWER	Marking Scheme	Remark
	– power – Limited amount of computing and communication capabilities • Thus, they need different types of operating systems depending on the capabilities they support. e.g. a PDA OS is different from a Smartphone OS. Operating System is a piece of software responsible for management of operations, control, coordinate the use of the hardware among the various application programs, and sharing the resources of a device • There are many mobile operating systems. The followings are the most important ones: – Java ME Platform – Palm OS – Symbian OS – Linux OS – Windows Mobile OS – BlackBerry OS – iPhone OS – Google Android Platform		
c)	Multithreading model System provides support to both user and kernel threads , resulting in different types of multithreading models Many to one model One to one model Many to many model	Diagram along with description 4 Marks	

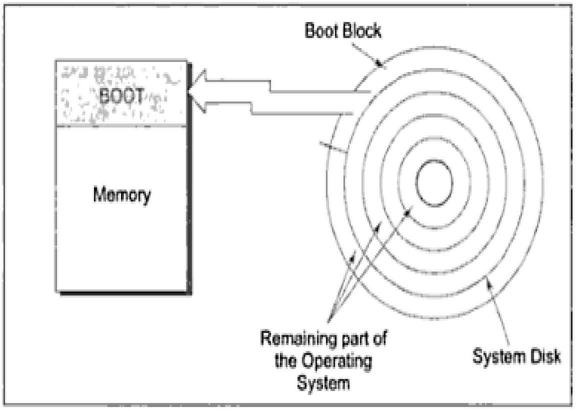
Q.No.	MODEL ANSWER	Marking Scheme	Remark
	i) Many to one  ii) One to One  Iii) Many to Many  		

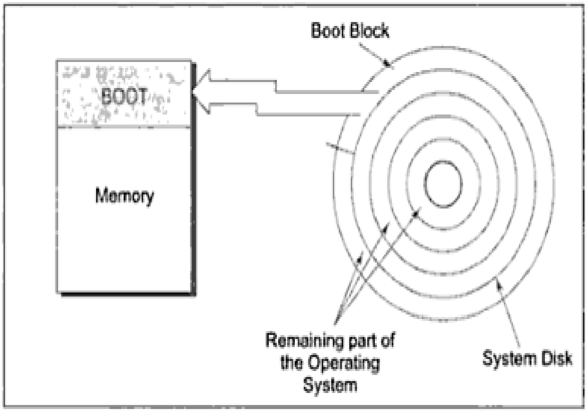
Q.No.	MODEL ANSWER	Marking Scheme	Remark
d)	Memory allocation • Contiguous allocation • Linked allocation • Indexed allocation  Contiguous Allocation ■ Multiple-partition allocation • Hole – block of available memory; holes of various size are scattered throughout memory • When a process arrives, it is allocated memory from a hole large enough to accommodate it • Operating system maintains information about: a) allocated partitions b) free partitions (hole) OS process 5 process 8 process 2 OS process 5 process 2 OS process 5 process 9 process 2 OS process 5 process 9 process 10 process 2  Dynamic Storage-Allocation Problem How to satisfy a request of size *n* from a list of free holes ■ **First-fit**: Allocate the *first* hole that is big enough ■ **Best-fit**: Allocate the *smallest* hole that is big enough; must search entire list, unless ordered by size • Produces the smallest leftover hole ■ **Worst-fit**: Allocate the *largest* hole; must also search entire list • Produces the largest leftover hole First-fit and best-fit better than worst-fit in terms of speed and storage utilization	Explanation of any one method along with diagram 4 Marks	

Q.No.	MODEL ANSWER	Marking Scheme	Remark
e)	Process control block each process is represented in the operating system by a process control block(PCB). Process state Program counter CPU registers CPU Scheduling Information Memory management information  Process Control Block (PCB) process state process number program counter registers memory limits list of open files • • •	Diagram 2 Marks, Any four points 2 Marks	
f)	Waiting time for P1 = 0 + 10 – 1 = 9ms P2 = 1-1 = 0ms P3 = 17 – 2 = 15ms P4 = 5-3 = 2ms Average waiting time = (9 + 0 + 15 + 2)/4 = 6.5ms **Gantt chart for Preemptive SJF** 	1 Mark for Gantt Chart, 3 Marks for calculation of avg. waiting time	

Q.No.	MODEL ANSWER	Marking Scheme	Remark
Q5 a)	Attempt any FOUR of the following: Describe evolution of Operating System. Batch Multi programmed Multitasking Time sharing Desktop Multiprocessor Systems Distributed Systems Clustered System Real Time system **Batch Systems** Several jobs are kept in main memory at the same time, and the CPU is multiplexed among them. S **Multiprogramming** I/O routine supplied by the system. Memory management – the system must allocate the memory to several jobs. CPU scheduling – the system must choose among several jobs ready to run. Allocation of devices. **Multitasking** Multitasking is a logical extension of multiprogramming. Multiple jobs are executed by the CPU switching between them,but the switches occur so frewuently that the users may interact with each program while it is running. **Time-Sharing Systems–Interactive Computing** The CPU is multiplexed among several jobs that are kept in memory and on disk (the CPU is allocated to a job only if the job is in memory). A job swapped in and out of memory to the disk. On-line communication between the user and the system is provided; when the operating system finishes the execution of one command, it seeks the next “control statement” from the user’s keyboard. On-line system must be available for users to access data and code.	List of evolution of operating system 1 mark Description 3 marks	

Q.No.	MODEL ANSWER	Marking Scheme	Remark
	Desktop Systems *Personal computers* – computer system dedicated to a single user. I/O devices – keyboards, mice, display screens, small printers. User convenience and responsiveness. Can adopt technology developed for larger operating system' often individuals have sole use of computer and do not need advanced CPU utilization of protection eatures. May run several different types of operating systems (Windows, MacOS, UNIX, Linux) **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. A DDP system involves a partitioning of the computing function and may also involve a distributed of databases, device control and interaction (network) control. **Real Time system** A Real Time system is used when there are regid 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. A Real Time system is considered to function correctly only if it returns the correct result within any time constraint. Hard real-time system Soft real-time system		

Q.No.	MODEL ANSWER	Marking Scheme	Remark
b)	Explain System Booting in detail. The loading of the operating system is achieved by a special program called BOOT. Generally this program is stored in one (or two) sectors on the disk with a pre-determined address. This portion is normally called 'BOOT Block' as shown in fig. The ROM normally contains a minimum program. When one turns the computer 'ON', the control is transferred to this program automatically by the hardware itself. This program in ROM loads the BOOT program in pre-determined memory locations. The beauty is to keep BOOT program as small as possible, so that the hardware can manage to loac it easily and in a very few instructions. This BOOT program in turn contains to read the rest of the Operating System into the memory. This is depicted in figures. The mechanism gives an impression of pulling oneself up. Therefore, the nomenclature bootstrapping or its short form booting  Boot Block BOOT Memory Remaining part of the Operating System System Disk	Explanation 3 marks Any one diagram 1 mark	

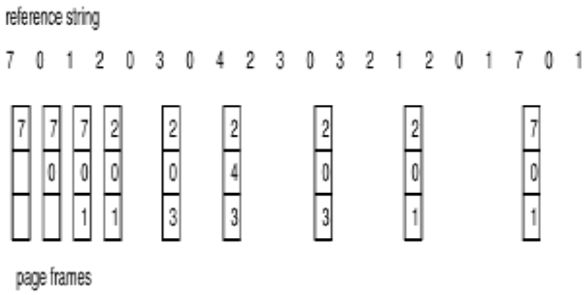
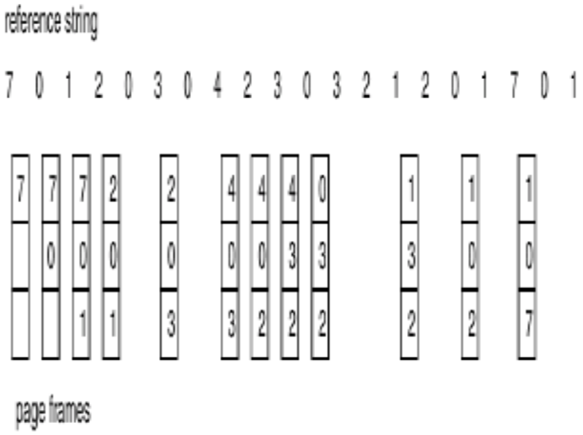
Q.No.	MODEL ANSWER	Marking Scheme	Remark
	 Boot Block BOOT Memory Remaining part of the Operating System System Disk		
c)	Differentiate Between Short-term Scheduler and Long term Scheduler **Schedulers** Long-term scheduler (or job scheduler) – selects which processes should be brought into the ready queue. Long-term scheduler is invoked very infrequently(seconds, minutes) _ (may be slow). The long-term scheduler controls the *degree of multiprogramming* Short-term scheduler (or CPU scheduler) – selects which process should be executed next and allocates CPU. Short-term scheduler is invoked very frequently(milliseconds) _ (must be fast). Long-term scheduler is invoked very infrequently(seconds, minutes) _ (may be slow).	Each differentiation 1 mark. Minimum 4 differentiation.	

Q.No.	MODEL ANSWER	Marking Scheme	Remark
d)	Explain Banker's Algorithm for deadlock prevention? **Banker's Algorithm** Multiple instances. Each process must a priori claim maximum use. When a process requests a resource it may have to wait. When a process gets all its resources it must return them in a finite amount of time. 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 2 marks Data structure 1 marks Example 1 mark	

Q.No.	MODEL ANSWER	Marking Scheme	Remark
	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 updation 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 o 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 tat 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. 2. **Resource request algorithm:** Let Requesti be the request vector for the process Pi. If Requesti[j] == k, then process Pi wants k instances of resource type Rj. When a request for resources is made by process Pi, the following actions are taken: 1. If Requesti <= Needj, go to step 2. Otherwise raise an error condition, since the process has executed its maximum claim. 2. If Requesti <= Available, go to step 3. Otherwise pi must wait, since the resources are not available. 3. Have the system pretend to have allocated te requested resources to process Pi by		

	modifying the state as follows. Available – Available- Request ; Allocationi = Allocationi + Request; Needi = Needi – requesti; If the resulting resource allocation state is safe, the transaction is completed and process Pi is allocated its resources. If the new state is unsafe, the Pi must wait for Requesti and the old resource allocation state is restored. **Data Structures for the Banker's Algorithm** ***Available****:* Vector of length *m*. If available [*j*] = *k*, there are *k* instances of resource type *Rj* available. ***Max:*** *n x m* matrix. If *Max* [*i,j*] = *k*, then process *Pi* may request at most *k* instances of resource type *Rj*. ***Allocation****:* *n* x *m* matrix. If Allocation[*i,j*] = *k* then *Pi* is currently allocated *k* instances of *Rj.* ***Need:*** *n* x *m* matrix. If *Need*[*i,j*] = *k*, then *Pi* may need *k* more instances of *Rj* to complete its task. *Need* [*i,j] = Max*[*i,j*] – *Allocation* [*i,j*]. Let *n* = number of processes, and *m* = number of resources types. One Example of Banker's Algorithm		
e)	What are different free space management techniques? Describe any one in detail? 1.Bit Vector The free-space list is implemented as a bit map or bit vector. Each block is represented by 1 bit. If the block is free, the bit is 1; if the block is allocated, the bit is 0. 2. Linked List Another approach is to link together all the free disk blocks, keeping a pointer to the first free block in a special location on the disk and caching it in memory. 3. Grouping Store the addresses of n free blocks in the first free block. The first n-1 of these blocks are actually free. The last block contains the addresses of another n free blocks, and so on.	List 1 mark Explanation 3 mark	

Q.No.	MODEL ANSWER	Marking Scheme	Remark
	4. counting Keep the address of the first free block and the number n of free contiguous blocks that follow the first block. Each entry requires more space than would a simple disk address, the overall list will be shorter, as long as the count is generally greater than 1.		
f)	State and explain criterias used in differentiating CPU scheduling? **CPU utilization** – keep the CPU as busy as possible. **Throughput** – no of processes that complete their execution per time unit. **Turnaround time** – amount of time to execute a particular process. The interval from the time of submission of a process to the time of completion is the turnaround time. **Waiting time** – amount of time a process has been waiting in the ready queue **Response time** – amount of time it takes from when a request was submitted until the first response is produced, **not** output (for time-sharing environment)	List 1 mark Explanation 3 marks	
Q.6	Attempt any FOUR of the following:		
a)	Explain any two page replacement algorithm with their advantages? First-In-First-Out (FIFO) Algorithm A FIFO replacement associates with each page the time when that page was bought into memory. When the page must be replaced we replace the page at the, the oldest page is chosen. We replace the page at the head of the queue. When a page is brought into the memory, we insert it at the tail of the queue.	List 1 mark Explanation of each with advantages 3 marks	

Q.No.	MODEL ANSWER	Marking Scheme	Remark
	 reference string 7 0 1 2 0 3 0 4 2 3 0 3 2 1 2 0 1 7 0 1 page frames Advantages: Easy to understand and execute. **Least Recently Used (LRU) Algorithm** Counter implementation Every page entry has a counter; every time page is referenced through this entry, copy the clock into the counter. When a page needs to be changed, look at the counters to determine which are to change  reference string 7 0 1 2 0 3 0 4 2 3 0 3 2 1 2 0 1 7 0 1 page frames		

Q.No.	MODEL ANSWER	Marking Scheme	Remark

Stack implementation – keep a stack of page numbers in a double link form:

Page referenced:

- move it to the top
- requires 6 pointers to be changed
- No search for replacement

Advantage:
Less number of faults as compared to FIFO

b) Describe the round robin scheduling algorithm with suitable example?

Explanation 3 marks
Example 1 mark

Round Robin (RR)

Design for time sharing systems.

Each process gets a small unit of CPU time (*time quantum*), usually 10-100 milliseconds. After this time has elapsed, the process is preempted and added to the end of the ready queue.

If there are *n* processes in the ready queue and the time quantum is *q*, then each process gets 1/*n* of the CPU time in chunks of at most *q* time units at once. No process waits more than (*n*-1)*q* time units.

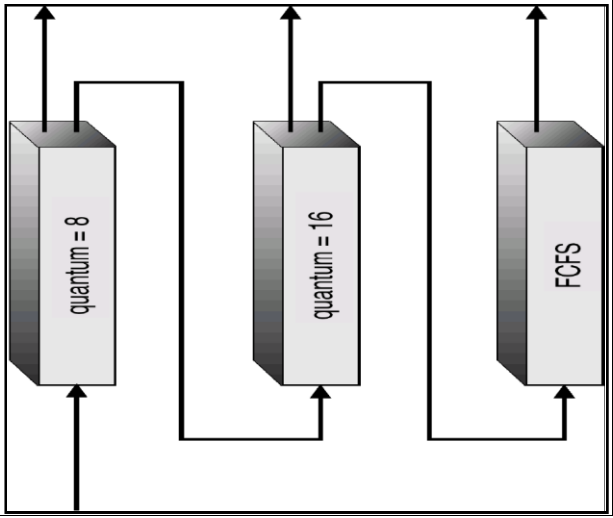
Any one example

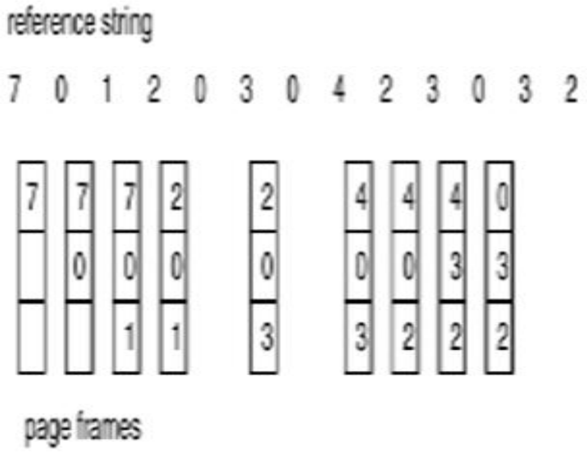
Process	Burst Time
P1	24
P2	3
P3	3

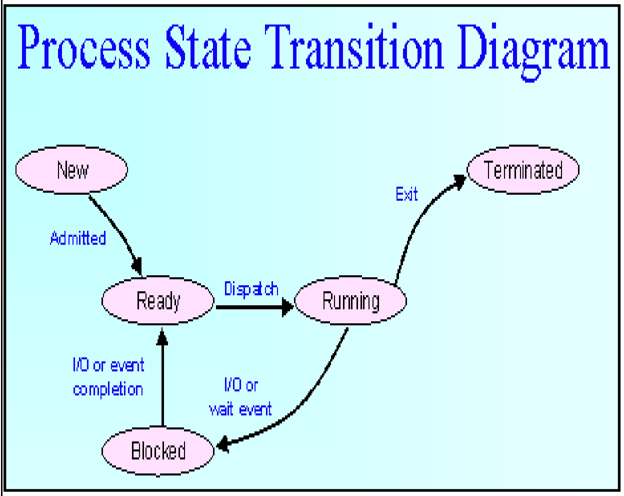
P1	P2	P3	P1	P1	P1	P1	P1

0 4 7 10 14 18 22 26 30

The average waiting time is 17/3=5.66 milliseconds.

Q.No.	MODEL ANSWER	Marking Scheme	Remark
c)	Explain Multilevel feedback-queue scheduling algorithm in detail? Multilevel Feedback Queue A process can move between the various queues; aging can be implemented this way. Multilevel-feedback-queue scheduler defined by the following parameters: • number of queues • scheduling algorithms for each queue • method used to determine when to upgrade a process • method used to determine when to demote a process • method used to determine which queue a process will enter when that process needs service Example of Multilevel Feedback Queue Three queues: *Q*0 – time quantum 8 milliseconds *Q*1 – time quantum 16 milliseconds *Q*2 – FCFS Scheduling A new job enters queue *Q0* which is served FCFS. When it gains CPU, job receives 8 milliseconds. If it does not finish in 8 milliseconds, job is moved to queue *Q*1. At *Q*1 job is again served FCFS and receives 16 additional milliseconds. If it still does not complete, it is preempted and moved to queue *Q*2.  	Explanation with diagram 3 marks Example 1 mark	

Q.No.	MODEL ANSWER	Marking Scheme	Remark
d)	Explain LRV page replacement algorithm for the reference string:7 0 1 2 0 3 0 4 2 3 0 3 2. Find out page fault and list drawback consider page frames=3. **Least Recently Used (LRU) Algorithm** Counter implementation - Every page entry has a counter; every time page is referenced through this entry, copy the clock into the counter. - When a page needs to be changed, look at the counters to determine which are to change. The result of applying LRU replacement to our example string is shown in fig. The LRU algorithm produces 9 page faults.  reference string 7 0 1 2 0 3 0 4 2 3 0 3 2 7 7 0 7 0 1 2 0 1 2 0 3 4 0 3 4 0 2 4 3 2 0 3 2 page frames	Explanation 3 marks Example 1 mark	Instead of LRV it should be LRU.
e)	Explain different process states with the help of state diagram? **Process State** As a process executes, it changes *state* **New**: The process is being created. **Running**: Instructions are being executed. **Waiting**: The process is waiting for some event to occur. **Ready**: The process is waiting to be assigned to a process. **Terminated**: The process has finished execution.	Explanation 3 marks diagram 1 mark	

Q.No.	MODEL ANSWER	Marking Scheme	Remark
	Diagram of Process State  Process State Transition Diagram New Admitted Ready Dispatch Running Exit Terminated I/O or event completion I/O or wait event Blocked		
f)	What are the different responsibilities of memory management Explain? **Main-Memory Management** 1. Memory is a large array of words or bytes, each with its own address. It is a repository of quickly accessible data shared by the CPU and I/O devices. 2. Main memory is a volatile storage device. It loses its contents in the case of system failure. 3. The operating system is responsible for the following activities in connections with memory management: • Keep track of which parts of memory are currently being used and by whom. • Decide which processes to load when memory space becomes available. • Allocate and de allocate memory space as needed	Any four responsibilities 1 mark each	