



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

SUMMER – 2024 EXAMINATION
MODEL ANSWER

Subject: Operating System

Subject Code:

22516

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.
- 8) As per the policy decision of Maharashtra State Government, teaching in English/Marathi and Bilingual (English + Marathi) medium is introduced at first year of AICTE diploma Programme from academic year 2021-2022. Hence if the students in first year (first and second semesters) write answers in Marathi or bilingual language (English +Marathi), the Examiner shall consider the same and assess the answer based on matching of concepts with model answer.

Q. No	Sub Q.N.	Answer	Marking Scheme
1.	a) Ans.	Attempt any <u>FIVE</u> of the following: List different types of operating system • Batch Operating System • Multi-Programming Operating System • Time Shared Operating System • Multiprocessor System • Distributed System • Real Time Operating System • Mobile Operating System	10 2M <i>Any four types 1/2M each</i>
	b) Ans.	State any four services provided by an operating system • User Interface • Program Execution • I/O Operation	2M <i>Any four services 1/2M each</i>



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

**SUMMER – 2024 EXAMINATION
MODEL ANSWER**

Subject: Operating System

Subject Code:

22516

		• File System Manipulation • Communication • Error Detection • Resource Allocation • Accounting • Protection and Security	
	c) Ans.	Draw process state diagram <pre> graph LR new((new)) -- admitted --> ready((ready)) ready -- scheduler dispatch --> running((running)) running -- interrupt --> ready running -- exit --> terminated((terminated)) running -- "I/O or event wait" --> waiting((waiting)) waiting -- "I/O or event completion" --> ready </pre>	2M <i>Correct diagram 2M</i>
	d) Ans.	State two features of non-preemptive scheduling • Once CPU is allocated to the process, the process keeps the CPU until it releases the CPU either by terminating or by switching to the waiting state. • A process switches from running state to waiting state or running to terminated state. • Low scheduling overhead. • Simple to implement and manage. • Less responsive in terms of priority for urgent task.	2M <i>Any two relevant features 1M each</i>
	e) Ans.	Define following terms: i. Memory compaction ii. Fragmentation i. Memory compaction: The process of shuffling the memory contents so as to place all free memory together in one large block is known as memory compaction. ii. Fragmentation: When processes are loaded and removed from memory, the free memory space is broken into little pieces which is known as fragmentation.	2M <i>Correct definition 1M each</i>



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

SUMMER – 2024 EXAMINATION
MODEL ANSWER

Subject: Operating System

Subject Code:

22516

	f) Ans.	Write syntax of PWD command and explain its use with the help of suitable example. Syntax: \$ pwd PWD command is used to print (display) current working directory. Example: \$ pwd Above example displays absolute pathname. Output: /home/poly	2M <i>Syntax 1M</i> <i>Use & example 1M</i>
	g) Ans.	List any four file operations. • Creating a file • Writing a file • Reading a file • Repositioning within a file • Deleting a file • Appending new information to the end of the file • Renaming an existing file. • Truncating a file • Creating copy of a file, copy file to another I/O device such as printer or display	2M <i>Any four operations 1/2M each</i>
2.	a) Ans.	Attempt any <u>THREE</u> of the following: Explain Resource management of an operating system . • Resource Management in Operating System is the process to manage all the resources efficiently like CPU, memory, input/output devices, and other hardware resources among the various programs and processes running in the computer. • Computer's resources are limited and several users or programs may need to utilize the same resources—such as memory and CPU—at the same time. • Operating system has to manage and ensure that all processes get the resources they need to execute, without any problems like deadlocks. • Operating systems use various scheduling algorithms to ensure that each process gets its fair share of CPU time. • Operating systems use virtual memory techniques to manage memory efficiently. • Operating systems use file system management techniques to manage disk space efficiently. • File system management allows the operating system to create,	12 4M <i>Any relevant explanation 4M</i>



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

SUMMER – 2024 EXAMINATION
MODEL ANSWER

Subject: Operating System

Subject Code:

22516

		delete, and modify files and directories in Secondary Storage Device. Operating systems use various network management techniques to manage network bandwidth efficiently.	
b) Ans.	Explain different components of operating system. List of System Components: 1. Process Management 2. Main Memory Management 3. File Management 4. I/O System Management 5. Secondary Storage Management 1.Process Management: A program is a set of instructions. When CPU is allocated to a program, it can start its execution. A program in execution is a process. A word processing program run by a user on a PC is a process. A process needs various system resources including CPU time, memory, files and I/O devices to complete the job execution. These resources can be given to the process when it is created or allocated to it while it is running. The operating system responsible for the following activities in connection with process management: • Creation and deletion of user and system processes. • Suspension and resumption of processes. • A mechanism for process synchronization. • A mechanism for process communication. • A mechanism for deadlock handling. 2. Main-Memory Management Main memory is a large array of words or bytes, ranging in size from hundreds of thousands to billions. Each word or byte has its own address. Main memory is a repository of quickly accessible data shared by the CPU and I/O devices. The central processor reads instructions from main memory during the instruction fetch cycle and both reads and writes data from main memory during the data fetch cycle. The main memory is generally the only large storage device that the CPU is able to address and access directly. The operating system responsible for the following activities in connection with main memory s management:	4M <i>Any two components with relevant explanation 2M each</i>	



SUMMER – 2024 EXAMINATION
MODEL ANSWER

Subject: Operating System

Subject Code:

22516

	• Keeping track of which parts of memory are currently being used and by whom. • Deciding which processes (or parts thereof) and data to move into and out of memory. • Allocating and deallocating memory space as needed. 3. File Management A file is a collected of related information defined by its creator. Computer can store files on the disk (secondary storage), which provide long term storage. Some examples of storage media are magnetic tape, magnetic disk and optical disk. Each of these media has its own properties like speed, capacity, and data transfer rate and access methods. A file system normally organized into directories to ease their use. These directories may contain files and other directions. The operating system responsible for the following activities in connection with file management: • The creation and deletion of files. • The creation and deletion of directions. • The support of primitives for manipulating files and directions. • The mapping of files onto secondary storage. • The backup of files on stable storage media. 4. I/O device Management Input / Output device management provides an environment for the better interaction between system and the I/O devices (such as printers, scanners, tape drives etc.). To interact with I/O devices in an effective manner, the operating system uses some special programs known as device driver. The device drivers take the data that operating system has defined as a file and then translate them into streams of bits or a series of laser pulses (in regard with laser printer). The I/O subsystem consists of several components: • A memory management component that includes buffering, caching, spooling • A general device driver interface • Drivers for specific hardware devices	
--	---	--

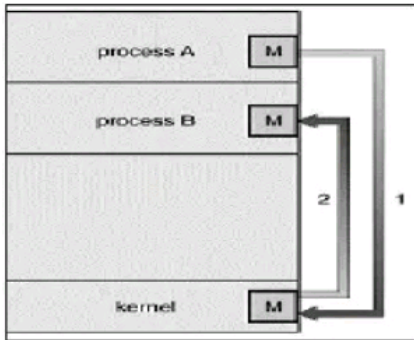


SUMMER – 2024 EXAMINATION
MODEL ANSWER

Subject: Operating System

Subject Code:

22516

	5. Secondary-Storage Management The computer system provides secondary storage to back up main memory. Secondary storage is required because main memory is too small to accommodate all data and programs, and the data that it holds is lost when power is lost. Most of the programs including compilers, assemblers, word processors, editors, and formatters are stored on a disk until loaded into memory. Secondary storage consists of tapes drives, disk drives, and other media. The operating system is responsible for the following activities in connection with disk management: • Free space management • Storage allocation • Disk scheduling.	
c) Describe message passing system of interprocess communication (IPC) Ans. 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 exists. 	4M <i>Relevant explanation 3M</i> <i>Correct diagram 1M</i>	



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

SUMMER – 2024 EXAMINATION
MODEL ANSWER

Subject: Operating System

Subject Code:

22516

	d) Ans.	What is CPU Scheduler? Explain the preemptive and non-preemptive type of scheduling CPU Scheduler: When the CPU becomes idle, the CPU scheduler selects one of the processes in the ready queue for execution. The CPU scheduler selects a process from the processes in memory that are ready to execute and allocates the CPU to that process. Pre-emptive scheduling: - Once the CPU is allocated to a process, the CPU is forcefully removed from that process due to occurrence of interrupt or completion of I/O request by the requesting process. In this scheduling, a process changes its status from running state to ready state or waiting state to ready state. Non-Preemptive scheduling: - Once the CPU is allocated to a process, the process keeps the CPU until it releases the CPU by terminating or by switching to the waiting state. Control of CPU is with the process to which it is allocated. In this scheduling, a process changes its status from running to waiting or running to terminate.	4M CPU scheduler 2M Relevant explanation of Preemptive 1M & non-preemptive 1M																		
3.	a) Ans.	Attempt any <u>THREE</u> of the following: Define Process. Draw a Process Control Block and explain the information in PCB Process: A process is defined as, a program under execution, which competes for the CPU time and other resources. A process is a program in execution. Process is also called as job, task and unit of work. <table><tr><td>Pointer</td><td>Process State</td></tr><tr><td colspan="2">Process Number</td></tr><tr><td colspan="2">Program Counter</td></tr><tr><td colspan="2">CPU Registers</td></tr><tr><td colspan="2">Memory Allocations</td></tr><tr><td colspan="2">Event Information</td></tr><tr><td colspan="2">List of Open Files</td></tr><tr><td colspan="2">:</td></tr><tr><td colspan="2">.</td></tr></table> Fig: Process Control Block	Pointer	Process State	Process Number		Program Counter		CPU Registers		Memory Allocations		Event Information		List of Open Files		:		.		12 4M Define process 1M Diagram of PCB with correct naming / block 1M Any 4 blocks of information in PCB 2M
Pointer	Process State																				
Process Number																					
Program Counter																					
CPU Registers																					
Memory Allocations																					
Event Information																					
List of Open Files																					
:																					
.																					



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

SUMMER – 2024 EXAMINATION
MODEL ANSWER

Subject: Operating System

Subject Code:

22516

	Following are the blocks in PCB: 1. Process Number: Each process is identified by its process number, called Process Identification Number (PID). Every process has a unique process-id through which it is identified. The process-id is provided by the OS. The process id of two processes could not be same because process-id is always unique. 2. Priority: Each process is assigned a certain level of priority that corresponds to the relative importance of the event that it services process priority is the preference of the one process over other process for execution. Priority may be given by the user/system manager or it may be given internally by OS. This field stores the priority of a particular process. 3. Process State: This information is about the current state of the process. The state may be new, ready, running, and waiting, halted, and so on. 4. Program Counter: The counter indicates the address of the next instruction to be executed for this process. 5. CPU Registers: The registers vary in number and type, depending on the computer architecture. They include accumulators, index registers, stack pointers, and general-purpose registers, plus any condition-code information. Along with the program counter, this state information must be saved when an interrupt occurs, to allow the process to be continued correctly afterward. 6. CPU Scheduling Information: This information includes a process priority, pointers to scheduling queues, and any other scheduling parameters. 7. Memory Management Information: This information may include such information as the value of the base and limit registers, the page tables, or the segment tables, depending on the memory system used by the operating system.	
--	---	--



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

SUMMER – 2024 EXAMINATION
MODEL ANSWER

Subject: Operating System

Subject Code:

22516

	8. Accounting Information: This information includes the amount of CPU and real time used, time limits, account numbers, job or process numbers, and so on. 9. I/O Status Information: This information includes the list of I/O devices allocated to the process, a list of open files, and so on. 10. File Management: It includes information about all open files, access rights etc. 11. Pointer: Pointer points to another process control block. Pointer is used for maintaining the scheduling list.	
b) Ans.	Define deadlock. State the conditions necessary for deadlock. Deadlock: Deadlock is a situation when two or more processes get locked and cannot processed further because of inter-dependability. In real world, deadlocks can arise when two persons wait for phone calls from one another. OR Deadlock is defined as, "a situation where a set of processes are blocked because each process is holding a resource and waiting for another resource acquired by some other process". Necessary Conditions to Deadlock: Mutual Exclusion: At least one resource is held in a non-sharable mode, that is only one process at a time can use the resource. If another process requests that resource, the requesting process must be delayed until the resource has been released. Each resource is either currently assigned to exactly one process or is available. Hold and Wait: There must exist a process that is holding at least one resource and is waiting to acquire additional resources that are currently being held by another process. Process currently holding resources granted earlier can request new resources. No Pre-emption: Resources cannot be pre-empted; i.e. resource can only be released voluntarily by the process holding it, after the process has completed its task. Resources previously granted cannot be forcibly taken away from a process. They must be explicitly released by the process holding them.	4M <i>Define deadlock 1M</i> <i>State any 3 conditions necessary for deadlock 3M</i>

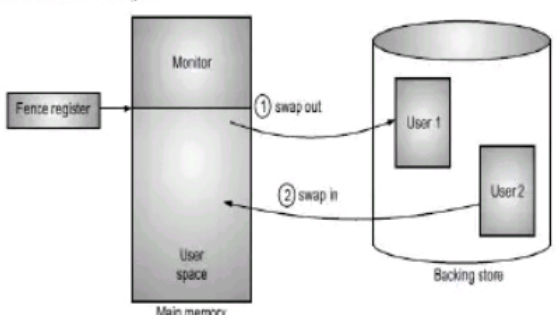


SUMMER – 2024 EXAMINATION
MODEL ANSWER

Subject: Operating System

Subject Code:

22516

		Circular Wait: There exist a set (P0, P1, ----- Pn) of waiting processes such that P0 is waiting for a resource which is held by P1, P1 is waiting for a resource which is held by P2. Pn-1 is waiting for resources which are held by Pn and Pn is waiting for a resource which is held by P0. Thus there must be a circular chain of two or more processes, each of which is waiting for a resource held by the next member of the chain.	
	c) Ans.	Explain the following terms with respect to memory management: i) Dynamic relocation ii) Swapping i) Dynamic Relocation When a program gets swapped out to a disk memory, then it is not always possible that when it is swapped back into main memory then it occupies the previous memory location, since the location may still be occupied by another process. We may need to relocate the process to a different area of memory. Thus there is a possibility that program may be moved in main memory due to swapping. ii) Swapping Swapping is mechanism in which a process can be swapped temporarily out of main memory (or move) to secondary storage (disk) and make that memory available to other processes. At some later time, the system swaps back the process from the secondary storage to main memory.  Fig: swapping in memory management	4M Each term explanation 2M



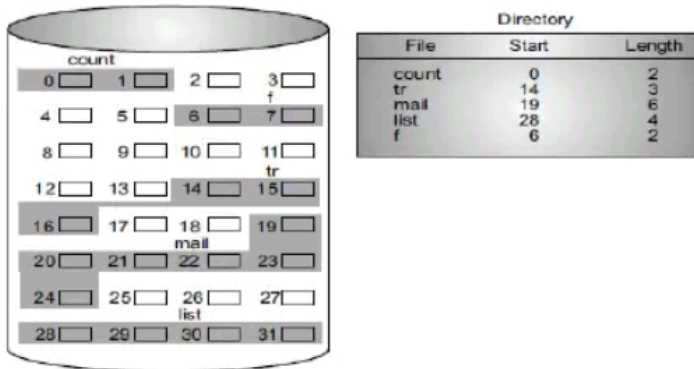
MAHARASHTRA STATE BOARD OF TECHNICAL EDUCATION
(Autonomous)
(ISO/IEC - 27001 - 2005 Certified)

SUMMER – 2024 EXAMINATION
MODEL ANSWER

Subject: Operating System

Subject Code:

22516

	d) Ans.	With suitable diagram, explain how contiguous file allocation is performed? • The contiguous allocation method requires each file to occupy a set of contiguous addresses on the disk. Disk addresses define a linear ordering on the disk. Contiguous allocation of a file is defined by the disk address of the first block and its length. If the file is 'n' blocks long and starts at location 'b', then it occupies blocks b, b+1, b+2, - - - - b+n-1. The directory entry for each file indicates the address of the starting block and the length of the area allocated for this file. • Contiguous allocation supports both sequential and direct access. • For direct access to block 'i' of a file, which starts at block 'b', we can immediately access block b+i. The difficulty with contiguous allocation is finding space for a new file. • For direct access to block 'i' of a file, which starts at block 'b', we can immediately access block b+i. • The difficulty with contiguous allocation is finding space for a new file. • If file to be created are 'n' blocks long, we must search free space list for 'n' free contiguous blocks. <table data-bbox="828 1247 1171 1382"><caption>Directory</caption><tr><th>File</th><th>Start</th><th>Length</th></tr><tr><td>count</td><td>0</td><td>2</td></tr><tr><td>tr</td><td>14</td><td>3</td></tr><tr><td>mail</td><td>19</td><td>6</td></tr><tr><td>list</td><td>28</td><td>4</td></tr><tr><td>f</td><td>6</td><td>2</td></tr></table>	File	Start	Length	count	0	2	tr	14	3	mail	19	6	list	28	4	f	6	2	4M Any relevant Diagram 1M Explanation 3M
File	Start	Length																			
count	0	2																			
tr	14	3																			
mail	19	6																			
list	28	4																			
f	6	2																			
4.	a)	Attempt any THREE of the following: Compare between Time sharing operating system and multiprogramming operative system.	12 4M																		

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

SUMMER – 2024 EXAMINATION
MODEL ANSWER

Subject: Operating System

Subject Code:

22516

Ans.

Time sharing operating system

In this time sharing Operating system many users/processes are allocated with computer resources in respective time slots.

Processors time is shared with multiple users that's why it is called as time sharing operating system.

In this process, two or more users can use a processor in their terminal.

Time sharing OS has fixed time slice.

Time sharing system minimizes response time.

Example: Windows NT.

Multiprogramming operating system

Multiprogramming operating system allows to execute multiple processes by monitoring their process states and switching in between processes.

Processor and memory underutilization problem is resolved and multiple programs runs on CPU that's why it is called multiprogramming.

In this, the process can be executed by a single processor.

Multi-programming OS has no fixed time slice.

Multiprogramming system maximizes processor use.

Example: Mac OS.

*Any 4
correct
points 1M
each*

b) Explain any four types of system call.

Ans.

1.Process and Job Control

- A running program needs to be able to halt its execution either normally (end) or abnormally (abort). If the program discovers an error in its input and wants to terminate abnormally, it may also want to define an error level.
- A process or job executing one program may want to load and execute another program. This allows the control card interpreter to execute program as directly by the control cards of the user job. If we create a new job or process, we should able to control its execution.
- We may also want to terminate a job or process that we created (terminate process). If we find that it is incorrect or no longer needed we need waiting time to finish execution (wait time). Another set of system calls are helpful in debugging a program.

4M
*Explanation
of any 4
system calls
1M each*



SUMMER – 2024 EXAMINATION
MODEL ANSWER

Subject: Operating System

Subject Code:

22516

	Example of Process related system call: - End, Abort. - Load, Execute. - Create process, Terminate process. - Ready process, Dispatch process. - Suspend process, Resume process. - Get process attributes, Set process attributes. - Wait for Time. - Wait event, Signal event. - Change priority of process. 2.File Management - We can identify several common system calls dealing with files. We first need to be able to create and delete files such a system call requires the name of the file and perhaps some of its attributes. - Once, the file is created, we need to open it and use it. We may also read, write and reposition. Finally we need to close the file, indicating that we are no longer using it. Example of file related system call: - Create file, Delete file - Open a file, Close a file. - Create directory. - Read, Write, Reposition. - Get file attributes, Set file attributes. - Create a link. - Change working directory. 3.Device Management - Files can be thought of as abstract or virtual devices. Thus many of the system calls for files are also needed for devices. - If there are multiple users of the system however we must first request the device to ensure that we have exclusive use of it. After we are finished with the device, we must release it. These functions are similar to the open/close system calls for files. - Once, the device has been requested we can read, write and reposition the device just as with files.	
--	--	--



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

SUMMER – 2024 EXAMINATION
MODEL ANSWER

Subject: Operating System

Subject Code:

22516

	System call related to device management: - Request device, Release device. - Read, Write, Reposition. - Get device attributes, set device attributes. 4. Information Maintenance - Many system calls exist simply for the purpose of transferring information between the user program and the operating system. For example most systems have a system call to return the current time and date. - Other system calls may return information about the system such as the number of current users, the version number of the operating system, the amount of free memory or disk space and so on. - In addition the operating system keeps information about all of its jobs and processes and there are system calls to access this information. Generally, there are also calls to reset it. (get process attributes and set process attributes). Information Maintenance related system call: - 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. 5. Communication - There are two models of inter-process communication, the message-passing model and the shared memory model. - Message-passing uses a common mailbox to pass messages between processes - Shared memory use certain system calls to create and gain access to regions of memory owned by other processes Communication related system calls: - Establish a connection. - Send, receive messages. - Terminate a connection	
c) Ans.	Describe how context switch is executed by operating system When CPU switches to another process, the system must save the state of the old process and load the saved state for the new process. This task is known as a context switch.	4M Diagram 1M



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

SUMMER – 2024 EXAMINATION
MODEL ANSWER

Subject: Operating System

Subject Code:

22516

	• CPU switching from one process to another process is called a context switch. • 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. • This enables multiple processes to share a single CPU. The context switch is an essential feature of a multitasking operating system. • When the process is switched, the following information is stored: • Program Counter • Scheduling Information • Base and limit register value • Currently used register • Changed State • I/O State • Accounting	Explanation 3M						
	Fig: Context Switching							
d) Ans.	Compare Short Job First (SJF) and Shortest Remaining Time (SRTN) scheduling algorithm (any four points) <table><tr><th>Short Job First (SJF)</th><th>Shortest Remaining Time (SRTN)</th></tr><tr><td>It is a non-preemptive algorithm.</td><td>It is a preemptive algorithm.</td></tr><tr><td>It involves less overhead than SRJF.</td><td>It involves more overheads than SJF.</td></tr></table>	Short Job First (SJF)	Shortest Remaining Time (SRTN)	It is a non-preemptive algorithm.	It is a preemptive algorithm.	It involves less overhead than SRJF.	It involves more overheads than SJF.	4M Any 4 correct points 1M each
Short Job First (SJF)	Shortest Remaining Time (SRTN)							
It is a non-preemptive algorithm.	It is a preemptive algorithm.							
It involves less overhead than SRJF.	It involves more overheads than SJF.							



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

SUMMER – 2024 EXAMINATION
MODEL ANSWER

Subject: Operating System

Subject Code:

22516

		<table><tr><td>It leads to comparatively lower throughput.</td><td>It leads to increased throughput as execution time is less.</td></tr><tr><td>It minimizes the average waiting time for each process.</td><td>It may or may not minimize the average waiting time for each process.</td></tr><tr><td>It involves lesser number of context switching.</td><td>It involves higher number of context switching.</td></tr><tr><td>Short processes are executed first and then followed by longer processes.</td><td>Shorter processes run fast and longer processes show poor response time.</td></tr></table>	It leads to comparatively lower throughput.	It leads to increased throughput as execution time is less.	It minimizes the average waiting time for each process.	It may or may not minimize the average waiting time for each process.	It involves lesser number of context switching.	It involves higher number of context switching.	Short processes are executed first and then followed by longer processes.	Shorter processes run fast and longer processes show poor response time.													
It leads to comparatively lower throughput.	It leads to increased throughput as execution time is less.																						
It minimizes the average waiting time for each process.	It may or may not minimize the average waiting time for each process.																						
It involves lesser number of context switching.	It involves higher number of context switching.																						
Short processes are executed first and then followed by longer processes.	Shorter processes run fast and longer processes show poor response time.																						
e) Ans.	Describe variable partitioning with the help of suitable example • In variable memory partitioning the partitions can vary in number and size. In variable memory partitioning the amount of memory allocated is exactly the amount of memory a process requires. • The operating system keeps a table indicating which parts of memory are available and which are occupied. Initially all memory is available for user programs and is considered as one large block of available memory, a hole. • When a job arrives and needs memory, we search for a hole large enough for this job. If we find one, we allocate only as much as is needed, keeping the rest available to satisfy future requests. For example, assume 256K memory available and a resident monitor of 40K. This situation leaves 216K for user programs. 0 40K 256K Monitor 216K <table><tr><th colspan="3">Job Queue</th></tr><tr><th>Job</th><th>Memory</th><th>Time</th></tr><tr><td>1</td><td>60 K</td><td>10</td></tr><tr><td>2</td><td>100 K</td><td>5</td></tr><tr><td>3</td><td>30 K</td><td>20</td></tr><tr><td>4</td><td>70 K</td><td>8</td></tr><tr><td>5</td><td>50 K</td><td>15</td></tr></table>	Job Queue			Job	Memory	Time	1	60 K	10	2	100 K	5	3	30 K	20	4	70 K	8	5	50 K	15	4M <i>Explanation</i> 2M <i>Any relevant Example</i> 2M
Job Queue																							
Job	Memory	Time																					
1	60 K	10																					
2	100 K	5																					
3	30 K	20																					
4	70 K	8																					
5	50 K	15																					



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

SUMMER – 2024 EXAMINATION
MODEL ANSWER

Subject: Operating System

Subject Code:

22516

5.	a) Ans.	Attempt any <u>TWO</u> of the following: Explain the use of following OS tools: i) Device Manager ii) Task Scheduler i) Device Manager use: 1. Managing all the hardware or virtual devices of computer system. 2. Allow interaction with hardware devices through device driver. 3. Used to install device and component-level drivers as well as associated software. 4. Allocate devices to the process as per process requirement and priority. 5. Deallocate devices either temporarily or permanently depending on condition. 6. Keeping track of all device's data and location. 7. Monitoring device status like printers, storage drivers and other devices. 8. Used to enforce the predetermined policies and decides which process receives the device when and for how long. ii) Task scheduler use: 1. Used to automate the execution of programs, scripts, and various tasks at specific intervals or specific events. 2. Used to select the next jobs to be admitted into the system and the next process to run. 3. Schedules tasks on CPU. Which task to run first, which has high priority, which has low latency is decided by schedulers.	12 6M <i>Any 2 uses of Device Manager 1½ M each</i> <i>Any 2 uses of Task Scheduler 1½ M each</i>

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

SUMMER – 2024 EXAMINATION
MODEL ANSWER

Subject: Operating System

Subject Code:

22516

4. Schedulers aim to provide fair access to system resources among competing processes or users.
5. Task schedulers strive to maximize the overall throughput of the system by executing tasks in a manner that minimizes idle time and maximizes the number of tasks completed within a given timeframe.

b) Explain user level thread and Kernel level thread with its advantages and disadvantages **6M**

Ans. User Level Thread

- In a user thread, all of the work of thread management is done by the application and the kernel is not aware of the existence of threads.
- The thread library contains code for creating and destroying threads, for passing message and data between threads, for scheduling thread execution and for saving and restoring thread contexts.
- The application begins with a single thread and begins running in that thread.
- User level threads are generally fast to create and manage.

*For each
Thread
explanation
1M*

*1 advantage
of each 1M*

*1 Dis-
advantage
of each
1M*

Advantages of user level thread over Kernel level thread:

1. Thread switching does not require Kernel mode privileges.
2. User level thread can run on any operating system.
3. Scheduling can be application specific.
4. User level threads are fast to create and manage.

*(Diagram
optional)*

Disadvantages of user level thread:

1. Limited Parallelism: Since user-level threads are managed within the context of a single process, they are not well-suited for applications that need to take full advantage of multi-core processors or for applications that require parallel execution across different processes.
2. Blocking: If a user-level thread blocks due to I/O or other reasons, all threads in the same process can be blocked, leading to poor utilization of resources.
3. No True Parallelism: User-level threads cannot execute in true parallel on multi-core systems since they are managed by a single kernel-level thread



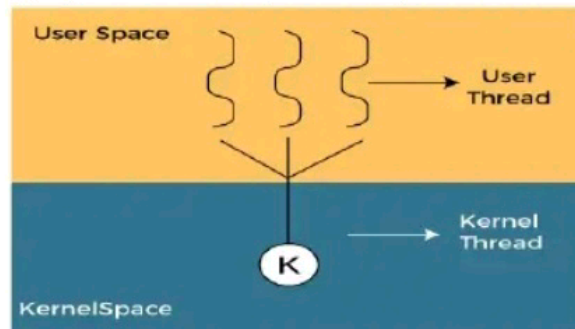
SUMMER – 2024 EXAMINATION
MODEL ANSWER

Subject: Operating System

Subject Code:

22516

4. It is not appropriate for a multiprocessor system.



Kernel Level Threads

- In Kernel level thread, thread management done by the Kernel.
- There is no thread management code in the application area.
- Kernel threads are supported directly by the operating system.
- Any application can be programmed to be multithreaded.
- All of the threads within an application are supported within a single process.
- The Kernel maintains context information for the process as a whole and for individual threads within the process.
- Scheduling by the Kernel is done on a thread basis.
- The Kernel performs thread creation, scheduling and management in Kernel space.
- Kernel threads are generally slower to create and manage than the user threads.

Advantages of Kernel level thread:

1. Kernel can simultaneously schedule multiple threads from the same process on multiple process.
2. If one thread in a process is blocked, the Kernel can schedule another thread of the same process.
3. Kernel routines themselves can multithreaded.

Disadvantages:

1. Kernel threads are generally slower to create and manage than the user threads.
2. Transfer of control from one thread to another within same process requires a mode switch to the Kernel.



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

SUMMER – 2024 EXAMINATION
MODEL ANSWER

Subject: Operating System

Subject Code:

22516

	c) Ans.	Consider the string: 0, 1, 2, 3, 0, 1, 2, 3, 0, 1, 2, 3, 4, 5, 6, 7 with frame size 3 and 4, calculate page fault in both the cases using FIFO algorithm <i>Note: Representation of frame can be in any order</i> FIFO : i)Frame Size=3 <table><tr><td>Ref.</td><td>0</td><td>1</td><td>2</td><td>3</td><td>0</td><td>1</td><td>2</td><td>3</td><td>0</td><td>1</td><td>2</td><td>3</td><td>4</td><td>5</td><td>6</td><td>7</td></tr><tr><td>F3</td><td></td><td></td><td>2</td><td>2</td><td>2</td><td>1</td><td>1</td><td>1</td><td>0</td><td>0</td><td>0</td><td>3</td><td>3</td><td>3</td><td>6</td><td>6</td></tr><tr><td>F2</td><td></td><td>1</td><td>1</td><td>1</td><td>0</td><td>0</td><td>0</td><td>3</td><td>3</td><td>3</td><td>2</td><td>2</td><td>2</td><td>5</td><td>5</td><td>5</td></tr><tr><td>F1</td><td>0</td><td>0</td><td>0</td><td>3</td><td>3</td><td>3</td><td>2</td><td>2</td><td>2</td><td>1</td><td>1</td><td>1</td><td>4</td><td>4</td><td>4</td><td>7</td></tr><tr><td>Fault</td><td>F</td><td>F</td><td>F</td><td>F</td><td>F</td><td>F</td><td>F</td><td>F</td><td>F</td><td>F</td><td>F</td><td>F</td><td>F</td><td>F</td><td>F</td><td>F</td></tr></table> OR <table><tr><td>Ref.</td><td>0</td><td>1</td><td>2</td><td>3</td><td>0</td><td>1</td><td>2</td><td>3</td><td>0</td><td>1</td><td>2</td><td>3</td><td>4</td><td>5</td><td>6</td><td>7</td></tr><tr><td>F1</td><td>0</td><td>0</td><td>0</td><td>3</td><td>3</td><td>3</td><td>2</td><td>2</td><td>2</td><td>1</td><td>1</td><td>1</td><td>4</td><td>4</td><td>4</td><td>7</td></tr><tr><td>F2</td><td></td><td>1</td><td>1</td><td>1</td><td>0</td><td>0</td><td>0</td><td>3</td><td>3</td><td>3</td><td>2</td><td>2</td><td>2</td><td>5</td><td>5</td><td>5</td></tr><tr><td>F3</td><td></td><td></td><td>2</td><td>2</td><td>2</td><td>1</td><td>1</td><td>1</td><td>0</td><td>0</td><td>0</td><td>3</td><td>3</td><td>3</td><td>6</td><td>6</td></tr><tr><td>Fault</td><td>F</td><td>F</td><td>F</td><td>F</td><td>F</td><td>F</td><td>F</td><td>F</td><td>F</td><td>F</td><td>F</td><td>F</td><td>F</td><td>F</td><td>F</td><td>F</td></tr></table> Total Page Faults :16 ii)Frame Size=4 <table><tr><td>Ref.</td><td>0</td><td>1</td><td>2</td><td>3</td><td>0</td><td>1</td><td>2</td><td>3</td><td>0</td><td>1</td><td>2</td><td>3</td><td>4</td><td>5</td><td>6</td><td>7</td></tr><tr><td>F4</td><td></td><td></td><td></td><td>3</td><td>3</td><td>3</td><td>3</td><td>3</td><td>3</td><td>3</td><td>3</td><td>3</td><td>3</td><td>3</td><td>3</td><td>7</td></tr><tr><td>F3</td><td></td><td></td><td>2</td><td>2</td><td>2</td><td>2</td><td>2</td><td>2</td><td>2</td><td>2</td><td>2</td><td>2</td><td>2</td><td>2</td><td>6</td><td>6</td></tr><tr><td>F2</td><td></td><td>1</td><td>1</td><td>1</td><td>1</td><td>1</td><td>1</td><td>1</td><td>1</td><td>1</td><td>1</td><td>1</td><td>1</td><td>5</td><td>5</td><td>5</td></tr><tr><td>F1</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>4</td><td>4</td><td>4</td><td>4</td></tr><tr><td>Fault</td><td>F</td><td>F</td><td>F</td><td>F</td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td>F</td><td>F</td><td>F</td><td>F</td></tr></table> OR <table><tr><td>Ref.</td><td>0</td><td>1</td><td>2</td><td>3</td><td>0</td><td>1</td><td>2</td><td>3</td><td>0</td><td>1</td><td>2</td><td>3</td><td>4</td><td>5</td><td>6</td><td>7</td></tr><tr><td>F1</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>4</td><td>4</td><td>4</td><td>4</td></tr><tr><td>F2</td><td></td><td>1</td><td>1</td><td>1</td><td>1</td><td>1</td><td>1</td><td>1</td><td>1</td><td>1</td><td>1</td><td>1</td><td>1</td><td>5</td><td>5</td><td>5</td></tr><tr><td>F3</td><td></td><td></td><td>2</td><td>2</td><td>2</td><td>2</td><td>2</td><td>2</td><td>2</td><td>2</td><td>2</td><td>2</td><td>2</td><td>2</td><td>6</td><td>6</td></tr><tr><td>F4</td><td></td><td></td><td></td><td>3</td><td>3</td><td>3</td><td>3</td><td>3</td><td>3</td><td>3</td><td>3</td><td>3</td><td>3</td><td>3</td><td>3</td><td>7</td></tr><tr><td>Fault</td><td>F</td><td>F</td><td>F</td><td>F</td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td>F</td><td>F</td><td>F</td><td>F</td></tr></table> Total Page Faults :08	Ref.	0	1	2	3	0	1	2	3	0	1	2	3	4	5	6	7	F3			2	2	2	1	1	1	0	0	0	3	3	3	6	6	F2		1	1	1	0	0	0	3	3	3	2	2	2	5	5	5	F1	0	0	0	3	3	3	2	2	2	1	1	1	4	4	4	7	Fault	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F	Ref.	0	1	2	3	0	1	2	3	0	1	2	3	4	5	6	7	F1	0	0	0	3	3	3	2	2	2	1	1	1	4	4	4	7	F2		1	1	1	0	0	0	3	3	3	2	2	2	5	5	5	F3			2	2	2	1	1	1	0	0	0	3	3	3	6	6	Fault	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F	Ref.	0	1	2	3	0	1	2	3	0	1	2	3	4	5	6	7	F4				3	3	3	3	3	3	3	3	3	3	3	3	7	F3			2	2	2	2	2	2	2	2	2	2	2	2	6	6	F2		1	1	1	1	1	1	1	1	1	1	1	1	5	5	5	F1	0	0	0	0	0	0	0	0	0	0	0	0	4	4	4	4	Fault	F	F	F	F									F	F	F	F	Ref.	0	1	2	3	0	1	2	3	0	1	2	3	4	5	6	7	F1	0	0	0	0	0	0	0	0	0	0	0	0	4	4	4	4	F2		1	1	1	1	1	1	1	1	1	1	1	1	5	5	5	F3			2	2	2	2	2	2	2	2	2	2	2	2	6	6	F4				3	3	3	3	3	3	3	3	3	3	3	3	7	Fault	F	F	F	F									F	F	F	F	6M FIFO implementat ion with frame size 3 3M FIFO implementat ion with frame size 4 3M
Ref.	0	1	2	3	0	1	2	3	0	1	2	3	4	5	6	7																																																																																																																																																																																																																																																																																																																																																																									
F3			2	2	2	1	1	1	0	0	0	3	3	3	6	6																																																																																																																																																																																																																																																																																																																																																																									
F2		1	1	1	0	0	0	3	3	3	2	2	2	5	5	5																																																																																																																																																																																																																																																																																																																																																																									
F1	0	0	0	3	3	3	2	2	2	1	1	1	4	4	4	7																																																																																																																																																																																																																																																																																																																																																																									
Fault	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F																																																																																																																																																																																																																																																																																																																																																																									
Ref.	0	1	2	3	0	1	2	3	0	1	2	3	4	5	6	7																																																																																																																																																																																																																																																																																																																																																																									
F1	0	0	0	3	3	3	2	2	2	1	1	1	4	4	4	7																																																																																																																																																																																																																																																																																																																																																																									
F2		1	1	1	0	0	0	3	3	3	2	2	2	5	5	5																																																																																																																																																																																																																																																																																																																																																																									
F3			2	2	2	1	1	1	0	0	0	3	3	3	6	6																																																																																																																																																																																																																																																																																																																																																																									
Fault	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F																																																																																																																																																																																																																																																																																																																																																																									
Ref.	0	1	2	3	0	1	2	3	0	1	2	3	4	5	6	7																																																																																																																																																																																																																																																																																																																																																																									
F4				3	3	3	3	3	3	3	3	3	3	3	3	7																																																																																																																																																																																																																																																																																																																																																																									
F3			2	2	2	2	2	2	2	2	2	2	2	2	6	6																																																																																																																																																																																																																																																																																																																																																																									
F2		1	1	1	1	1	1	1	1	1	1	1	1	5	5	5																																																																																																																																																																																																																																																																																																																																																																									
F1	0	0	0	0	0	0	0	0	0	0	0	0	4	4	4	4																																																																																																																																																																																																																																																																																																																																																																									
Fault	F	F	F	F									F	F	F	F																																																																																																																																																																																																																																																																																																																																																																									
Ref.	0	1	2	3	0	1	2	3	0	1	2	3	4	5	6	7																																																																																																																																																																																																																																																																																																																																																																									
F1	0	0	0	0	0	0	0	0	0	0	0	0	4	4	4	4																																																																																																																																																																																																																																																																																																																																																																									
F2		1	1	1	1	1	1	1	1	1	1	1	1	5	5	5																																																																																																																																																																																																																																																																																																																																																																									
F3			2	2	2	2	2	2	2	2	2	2	2	2	6	6																																																																																																																																																																																																																																																																																																																																																																									
F4				3	3	3	3	3	3	3	3	3	3	3	3	7																																																																																																																																																																																																																																																																																																																																																																									
Fault	F	F	F	F									F	F	F	F																																																																																																																																																																																																																																																																																																																																																																									
6.	a)	Attempt any <u>TWO</u> of the following: What is the average turnaround time for the following process using : i) FCFS scheduling algorithm ii) SJF non preemptive scheduling algorithm iii) Round Robin Scheduling algorithm	12 6M																																																																																																																																																																																																																																																																																																																																																																																						



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

SUMMER – 2024 EXAMINATION
MODEL ANSWER

Subject: Operating System

Subject Code:

22516

Process	Arrival time	Burst time
P ₁	0	8
P ₂	1	4
P ₃	2	1

Ans. *Note: Any time slice can be considered by student for Round Robin e.g. 2, 3, 4 etc*

i) Gantt Chart FCFS

P ₁	P ₂	P ₃
----------------	----------------	----------------

0

For each algorithm:
Gantt Chart
1M

Correct turnaround time calculation
1M each



SUMMER – 2024 EXAMINATION
MODEL ANSWER

Subject: Operating System

Subject Code:

22516

	iii) <u>Gantt Chart Round Robin</u> <u>Consider Time Quantum= 2 Units</u> <table><tr><td>P₁</td><td>P₂</td><td>P₃</td><td>P₁</td><td>P₂</td><td>P₁</td><td>P₁</td></tr><tr><td>0</td><td>2</td><td>4</td><td>5</td><td>7</td><td>9</td><td>11</td><td>13</td></tr></table> Turnaround Time =Completion Time - Arrival Time P₁=13-0=13 P₂=9-1=8 P₃=5-2=3 Average turnaround time=Turnaround time of all processes / Number of processes 	P ₁	P ₂	P ₃	P ₁	P ₂	P ₁	P ₁	0	2	4	5	7	9	11	13
P ₁	P ₂	P ₃	P ₁	P ₂	P ₁	P ₁										
0	2	4	5	7	9	11	13									



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

SUMMER – 2024 EXAMINATION
MODEL ANSWER

Subject: Operating System

Subject Code:

22516

For example, consider a disk where blocks 2, 3, 4, 5, 8, 9, 10, 11, 12, 13 are free and the rest of the blocks are allocated.

The free-space bit map would be : 001111001111100

0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
0	0	1	1	1	1	0	0	1	1	1	1	1	1	0	0

1=Free block 0= Allocated block

The main advantage of this approach is its relative simplicity and its efficiency in finding the first free block or n consecutive free blocks on the disk.

Advantages of Bit Map vector method

1. Simple and easy to understand.
2. Consumes less memory.
3. It is efficient to find free space.

Disadvantages of the Bit Map vector method

1. The operating system goes through all the blocks until it finds a free block.
2. It is not efficient when the disk size is large.

2) Linked List

In this approach, the free disk blocks are linked together i.e. a free block contains a pointer to the next free block. The block number of the very first disk block is stored at a separate location on disk and is also cache in memory. In this approach, link all the disk blocks together, keeping a pointer to the first free block. This block contains a pointer to the next free disk block, and so on.

Any 1
Advantage
1M,

Any 1
Disadvantage
1M

Linked List
explanation
1M

Any 1
Advantage
1M

Any 1
Disadvantage
1M

(diagram is
optional)



**SUMMER – 2024 EXAMINATION
MODEL ANSWER**

Subject: Operating System

Subject Code:

22516

		Advantages of linked list: 1. Files can be extended, no fragmentation problems. 2. Sequential Access is easy: just chase links. 3. No extra space for the free list. 4. Starting and ending of the file can be easily found out by the linked list. Disadvantages of linked list: 1. Cannot be effectively used for direct access method. 2. Random access is virtually impossible. Lots of seeking, even in sequential access. 3. Pointers which are required need additional memory space. 4. It is not reliable as the blocks are scattered on the disk anywhere.	
	c) Ans.	Explain with diagram single level directory structure and two level directory structure with its advantages and disadvantages. Single Level Directory Structure: This is the simplest directory structure. All the files are stored in the same directory which is easy to support and understand. But a unique name must be assigned to each file. If the number of files in the directory increases, it may become difficult to remember names of all files. Advantage: • Less time required to search the file, because all files are stored in same directory. • Single level directory structure is easy to implement and maintain. • Single level directory structure, the operations like creation, searching, deletion, updating are very easy and faster	6M <i>Single level directory structure explanation 1M,</i> <i>Any 1 Advantage 1M</i>



SUMMER – 2024 EXAMINATION
MODEL ANSWER

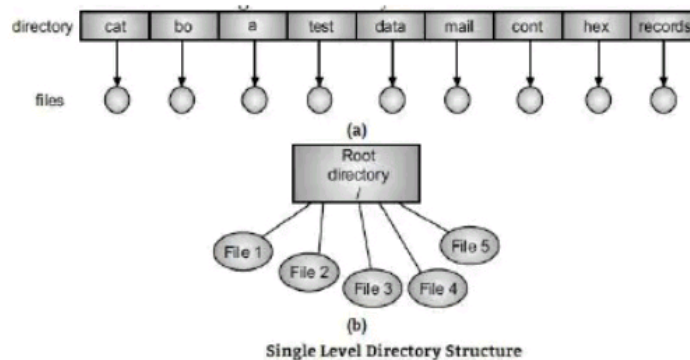
Subject: Operating System

Subject Code:

22516

Disadvantages

- Not suitable for a large number of files and more than one user.
- Because of single directory files, files require unique file names.
- Difficult to remember names of all the files as the number of files increases. MS-DOS OS allows only 11 character file name where as UNIX allows 255 character.



Two-level directory structure: -In the two-level structures, each user has its own user file directory (UFD). The UFD lists only files of a single user. System contains a master file directory (MFD) which is indexed by user name or account number. Each entry in MFD points to the UFD for that user.

When a user refers to a particular file, only his own UFD is searched. Different users can have files with the same name, as long as all the file names within each UFD are unique.

When we create a file for a user, operating system searches only that user's UFD to find whether same name file already present in the directory. For deleting a file again operating system checks the file name in the user's UFD only.

Any 1
disadvantage 1M

Two level
directory
structure
explanation
1M

Any 1
Advantage
1M

Any 1
disadvantage 1M

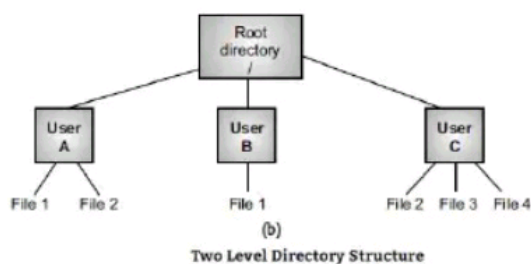


SUMMER – 2024 EXAMINATION
MODEL ANSWER

Subject: Operating System

Subject Code:

22516



Advantages:

- It solves isolation. i.e. data of one user is separated from other user.
- Searching is also efficient in this type of structure.
- Every user accesses his own directory and files.
- There is no problem of files with same names in different directories.

Disadvantages:

- Two users can not share data or communicate with each other.
- It is difficult to search the files with same names in different directories.