SRM VALLIAMMAI ENGINEERING COLLEGE
(An Autonomous Institution)

SRM Nagar, Kattankulathur – 603 203

DEPARTMENT OF INFORMATION TECHNOLOGY

QUESTION BANK

IV SEMESTER

1908402– OPERATING SYSTEMS CONCEPTS

Regulation – 2019

Academic Year 2020 – 2021(Even Semester)

Prepared by

Dr. S. Ravikumar, Assistant Professor (Sel.G)/IT

Mr. M. Kavinkumar, Assistant Professor (Sr.G)/IT

SRM VALLIAMMAI ENGINEERING COLLEGE

(An Autonomous Institution)

SRM Nagar, Kattankulathur – 603 203

DEPARTMENT OF INFORMATION TECHNOLOGY

QUESTION BANK

SUBJECT : 1908402– Operating Systems Concepts

SEM/YEAR : IV / II

UNIT – I: PROCESSES AND THREADS

Introduction to operating systems – OBJECTIVES: and functions, Evolution of Operating System - operating system-structures – system calls – system programs – System Generation and system boot Processes: Process concept – Process scheduling – Operations on processes –Inter process communication – Communication in client-server systems. Threads: Multi-threading models – Threading issues. Case study: IPC in Linux, Pthreads library

PART – A

Q.No.	Questions	BT Level	Competence
1	What are the 3 main purposes of an Operating System?	BTL1	Remembering
2	What is an Operating System?	BTL1	Remembering
3	List out the various operating system components.	BTL1	Remembering
4	List two programming examples of multithreading giving improved performance over a single-threaded solution.	BTL1	Remembering
5	List the five major activities of an operating system in regard to process management	BTL1	Remembering
6	What are threads?	BTL1	Remembering
7	Give the information that is kept in process control block	BTL2	Understanding
8	Infer the co-operating process	BTL2	Understanding
9	Outline the different differences between user level threads & Kernel supported threads	BTL2	Understanding
10	Compare tightly coupled systems with loosely coupled systems	BTL2	Understanding
11	Is OS are source Manager? If yes justify your answer.	BTL3	Applying
12	Illustrate how time sharing different from multiprogramming?	BTL3	Applying
13	Identify the use of fork and exec system calls.	BTL3	Applying
14	Analyze the dual mode operation and its need?	BTL4	Analyzing
15	Differentiate DMA and Cache memory	BTL4	Analyzing
16	Analyze some system calls which is required to control the communication system	BTL4	Analyzing
17	Judge How can a user program disturb the normal operation of the system	BTL5	Evaluating
18	Assess the use of inter process communication	BTL5	Evaluating
19	Can a multithreaded solution using multiple user-level threads achieve better performance on a multiprocessor system than on a single processor system?	BTL6	Creating
20	Some computer systems do not provide a privileged mode of	BTL6	Creating

	operation in hardware. Is it possible to construct a secure operating system for these computer systems?		
PART – B			
1	Discuss the evolution of operating system(13)	**BTL1**	**Remembering**
2	What are the advantages and disadvantages of using the same system call interface for both files and devices(13)	**BTL1**	**Remembering**
3	Discuss the essential properties of the following types of systems i) Time sharing systems(4) ii) Multi-processor systems(4) iii) Distributed systems(5)	**BTL1**	**Remembering**
4	What are the primary goals of conflict-resolution mechanisms used by the Linux kernel for loading kernel modules(13)	**BTL1**	**Remembering**
5	Describe the differences between symmetric and asymmetric multiprocessing. What are three advantages and one disadvantages of multiprocessor systems?(13)	**BTL2**	**Understanding**
6	(i)Summarize about the functions of Operating Systems in detail(7) (ii)Summarize the different multiprocessor organizations with block diagrams.(6)	**BTL2**	**Understanding**
7	Describe the cache memory and its mapping in detail(13)	**BTL2**	**Understanding**
8	Explain different operating system structures with neat sketch(13)	**BTL3**	**Applying**
9	Illustrate Multithreading models in detail(13)	**BTL3**	**Applying**
10	Demonstrate the three methods for passing parameters to the OS with examples. (13)	**BTL3**	**Applying**
11	i) Elaborate threads in detail? How do they differ from a process?(7) ii) Explain the difference in process level switching and thread level switching(6)	**BTL4**	**Analyzing**
12	How could a system be designed to allow a choice of operating systems from which to boot? What would the bootstrap program need to do? (13)	**BTL4**	**Analyzing**
13	(i)Evaluate the various types of system calls with an example for each.(6) (ii)Evaluate the functionality of system boot with respect to an Operating System.(7)	**BTL5**	**Evaluating**
14	State the operating system structure and its operations in detail. Justify the reason why the lack of a hardware supported dual mode can cause serious short coming in an operating system? (13)	**BTL6**	**Creating**
PART – C			
1	Give reasons why caches are useful. What problems do they solve and cause? If a catch can be made as large as the device for which it is catching why not make it that large and eliminate the device?(15)	**BTL6**	**Creating**
2	(i)With neat sketch discuss computer system overview.(8) (ii)Enumerate the different operating system structure and explain with neat sketch.(7)	**BTL6**	**Creating**
3	(i)Evaluate a thread creation and termination with example program and state how many threads does a process have? (10) (ii)How threads are created in Linux? Does Linux use threads?(5)	**BTL5**	**Evaluating**
4	(i)State the basic functions of OS and DMA.(5) (ii)Explain system calls system programs and OS generation.(10)	**BTL5**	**Evaluating**

UNIT - II: PROCESS SCHEDULING AND SYNCHRONIZATION

CPU Scheduling: Scheduling criteria – Scheduling algorithms – Multilevel Queue scheduling - Multilevel feedback Queue Scheduling-Process Synchronization: The critical section problem – Semaphores – Classic problems of synchronization –critical regions. Deadlock: System model – Deadlock characterization – Methods for handling deadlocks – Deadlock prevention – Deadlock avoidance –Deadlock detection – Recovery from deadlock. Case study: Process scheduling in Linux

PART – A

Q.No.	Questions	BT Level	Competence
1	List out different types of CPU Schedulers	BTL1	Remembering
2	What are classical problems of synchronization	BTL1	Remembering
3	What are semaphores	BTL1	Remembering
4	Define the terms critical section and mutual exclusion	BTL1	Remembering
5	What is a deadlock?	BTL1	Remembering
6	List the functions of Dispatcher Module.	BTL1	Remembering
7	What are the requirements that a solution to the critical section problem must satisfy?	BTL2	Understanding
8	What are the necessary conditions for deadlock to occur?	BTL2	Understanding
9	Outline the difference between the preemptive and non-preemptive scheduling	BTL2	Understanding
10	Give the queueing diagram representation of process scheduling	BTL2	Understanding
11	Distinguish between CPU bounded and I/O bounded processes.	BTL3	Applying
12	Under what circumstances would a user be better off using a time sharing system rather than a PC or single–user workstation?	BTL3	Applying
13	Differentiate deadlock and starvation	BTL3	Applying
14	Explain how resource allocation graph can be used to check for deadlock in a system	BTL4	Analyzing
15	Explain the deadlock avoidance algorithm	BTL4	Analyzing
16	Is the context switching an overhead? Justify your answer.	BTL4	Analyzing
17	Evaluate the concept behind strong semaphore and spinlock?	BTL5	Evaluating
18	Name two hardware instructions and their definitions which can be used for implementing mutual exclusion.	BTL5	Evaluating
19	“If there is a cycle in the resource allocation graph, it may or may not be indeed lock state“. Comment on this statement.	BTL6	Creating
20	“Priority in version is a condition that occurs in real time systems where a low priority process is starved because higher priority processes have gained hold of the CPU”–Comment on this statement.	BTL6	Creating

PART – B

Q.No.	Questions	BT Level	Competence
1	Define CPU utilization, throughput, and turnaround time, waiting time and response time(13)	BTL1	Remembering
2	What is critical section problem? Write a solution to n process critical section problem(13)	BTL1	Remembering
3	Discuss how the following pairs of scheduling criteria conflict in certain settings. i. CPU utilization and response time.(4) ii. Average turn around time and maximum waiting time.(5) iii. I/O device utilization and CPU utilization.(4)	BTL1	Remembering
4	What is the criterion used to select the time quantum in case of round-robin scheduling algorithm? Explain it with a suitable example. (13)	BTL1	Remembering

<table>
<tr><td>5</td><td>Outline the Deadlock detection with suitable example.(13)</td><td><b>BTL2</b></td><td><b>Understanding</b></td></tr>
<tr><td>6</td><td>What is a semaphore and a counting semaphore? Explain how a semaphore can be used so that statement S1 of process P1 is always executed first and only then statement S2of process P2 is executed.(13)</td><td><b>BTL2</b></td><td><b>Understanding</b></td></tr>
<tr><td>7</td><td>Describe the differences among short- term, medium-term and long-term scheduling with suitable example(13)</td><td><b>BTL2</b></td><td><b>Understanding</b></td></tr>
<tr><td>8</td><td>Distinguish between symmetric and asymmetric communication between processes(13)</td><td><b>BTL3</b></td><td><b>Applying</b></td></tr>
<tr><td>9</td><td>Explain in detail about the Process scheduling in Linux(13)</td><td><b>BTL3</b></td><td><b>Applying</b></td></tr>
<tr><td>10</td><td>Explain the synchronizing protocol of a classical readers/writers problem. Write a symbolic program code to implement any one of the above protocol(13)</td><td><b>BTL3</b></td><td><b>Applying</b></td></tr>
<tr><td>11</td><td>Explain the differences in the degree to which the following scheduling algorithms discriminate in favor of short processes:
(i)RR(7)
(ii)Multilevel feedback queues.(6)</td><td><b>BTL4</b></td><td><b>Analyzing</b></td></tr>
<tr><td>12</td><td>(i)Explain why interrupts are not appropriate for implementing synchronous primitives in multiprocessor systems.(7)
(ii)Compute the average waiting time for the processes using non-preemptive SJF scheduling algorithm.(6)</td><td><b>BTL4</b></td><td><b>Analyzing</b></td></tr>
<tr><td>13</td><td>Consider the snapshot of a system
<table>
<tr><td></td><td colspan="4">Max</td><td colspan="4">Allocation</td><td colspan="4">Available</td></tr>
<tr><td></td><td>A</td><td>B</td><td>C</td><td>D</td><td>A</td><td>B</td><td>C</td><td>D</td><td>A</td><td>B</td><td>C</td><td>D</td></tr>
<tr><td>P0</td><td>2</td><td>0</td><td>0</td><td>0</td><td>4</td><td>2</td><td>1</td><td>2</td><td>3</td><td>3</td><td>2</td><td>1</td></tr>
<tr><td>P1</td><td>3</td><td>1</td><td>2</td><td>1</td><td>5</td><td>2</td><td>5</td><td>2</td><td></td><td></td><td></td><td></td></tr>
<tr><td>P2</td><td>2</td><td>1</td><td>0</td><td>3</td><td>2</td><td>3</td><td>1</td><td>6</td><td></td><td></td><td></td><td></td></tr>
<tr><td>P3</td><td>1</td><td>3</td><td>1</td><td>2</td><td>1</td><td>4</td><td>2</td><td>4</td><td></td><td></td><td></td><td></td></tr>
<tr><td>P4</td><td>1</td><td>4</td><td>3</td><td>2</td><td>3</td><td>6</td><td>6</td><td>5</td><td></td><td></td><td></td><td></td></tr>
</table>
Answer the following Using Banker's algorithm,
(i)Illustrate that the system is in safe state by demonstrating an order in which the processes may complete?(5)
(ii)If a request from process P1 arrives for(1,1,0,0)can the request be granted immediately?(4)
(iii)if the request from p4 arrives for(0,0,2,0)can the request be granted immediately?(4)</td><td><b>BTL5</b></td><td><b>Evaluating</b></td></tr>
<tr><td>14</td><td>Consider the set of 5 processes and calculate the turn around and waiting time for the execution of these processes using FCFS, a non preemptive priority and RR (quantum=1) (13)
<table>
<tr><td>Process</td><td>Burst</td><td>Priority</td><td>Arrival Time</td></tr>
<tr><td>P1</td><td>8</td><td>4</td><td>0</td></tr>
<tr><td>P2</td><td>6</td><td>1</td><td>2</td></tr>
<tr><td>P3</td><td>1</td><td>2</td><td>2</td></tr>
<tr><td>P4</td><td>9</td><td>2</td><td>1</td></tr>
<tr><td>P5</td><td>3</td><td>3</td><td>3</td></tr>
</table></td><td><b>BTL6</b></td><td><b>Creating</b></td></tr>
<tr><td colspan="4"><b>PART – C</b></td></tr>
<tr><td>1</td><td>Consider the set of 4 processes whose arrival time and burst time are given below-
<table>
<tr><td>Process</td><td>Arrival</td><td>Burst Time</td></tr>
</table></td><td><b>BTL6</b></td><td><b>Creating</b></td></tr>
</table>

<table>
<tr><td></td><td>
<table>
<tr><th>No.</th><th>Time</th><th>CPU Burst</th><th>I/O Burst</th><th>CPU Burst</th></tr>
<tr><td>P1</td><td>0</td><td>3</td><td>2</td><td>2</td></tr>
<tr><td>P2</td><td>0</td><td>2</td><td>4</td><td>1</td></tr>
<tr><td>P3</td><td>2</td><td>1</td><td>3</td><td>2</td></tr>
<tr><td>P4</td><td>5</td><td>2</td><td>2</td><td>1</td></tr>
</table>
If the CPU scheduling policy is Shortest Remaining Time First, calculate the average waiting time and average turn around time.(15)</td><td></td><td></td></tr>
<tr><td>2</td><td>Which of the following scheduling algorithms could result in starvation? Justify in detail.
(i)First-come, first-served(5)
(ii)Shortest job first(5)
(iii)Round robin(5)</td><td><b>BTL6</b></td><td><b>Creating</b></td></tr>
<tr><td>3</td><td>Consider a system consisting of ‘m’ resources of the same type, being shared by ‘n’ processes. Resources can be requested and released by processes only one at a time. Show that the system is dead lock free if the following two conditions hold
i) The maximum need of each process is between 1and m resources (8)
ii)The sum of all maximum needs is less than m + n.(7)</td><td><b>BTL5</b></td><td><b>Evaluating</b></td></tr>
<tr><td>4</td><td>Consider the following system snapshot using data structures in the Banker’s algorithm with resources A, B, C and D and process P0toP4:
<table>
<tr><th></th><th>Max</th><th>Allocation</th><th>Available</th><th>Need</th></tr>
<tr><td></td><td>ABCD</td><td>ABCD</td><td>ABCD</td><td>ABCD</td></tr>
<tr><td>P0</td><td>6012</td><td>4001</td><td>3211</td><td></td></tr>
<tr><td>P1</td><td>1750</td><td>1100</td><td></td><td></td></tr>
<tr><td>P2</td><td>2356</td><td>1254</td><td></td><td></td></tr>
<tr><td>P3</td><td>1653</td><td>0633</td><td></td><td></td></tr>
<tr><td>P4</td><td>1656</td><td>0212</td><td></td><td></td></tr>
</table>
Using Banker’s algorithm ,answer the following questions:
(i)How many resources of type A, B, C and D are there?(3)
(ii)What are the contents of the need matrix?(3)
(iii)Is the system in a safe state? Why?(3)
(iv)If a request from process P4 arrives for additional resources of (1,2,0,0)can the banker’s algorithm grant their quest immediately? Show the new system state and other criteria.(6)</td><td><b>BTL5</b></td><td><b>Evaluating</b></td></tr>
</table>

UNIT - III: STORAGE MANAGEMENT

Main Memory Management: Background – Swapping – Contiguous memory allocation – Paging – Segmentation – Segmentation with paging. Virtual Memory: Background – Demand paging – Process creation – Page replacement – Allocation of frames – Thrashing. Case Study: Memory management in Linux

PART – A

Q.No.	**Questions**	**BT Level**	**Competence**
1	Define: Belady’s anomaly	**BTL1**	**Remembering**
2	What is the purpose of paging the page table?	**BTL1**	**Remembering**
3	Define Overlays and swapping	**BTL1**	**Remembering**

4	Define demand paging in memory management	BTL1	Remembering
5	List the steps required to handle a page fault in demand paging?	BTL1	Remembering
6	Define lazy swapper and pure Demand Paging	BTL1	Remembering
7	How the problem of external fragmentation can be solved	BTL2	Understanding
8	Name two differences between logical and physical addresses.	BTL2	Understanding
9	What are the common strategies to select a free hole from a set of available holes	BTL2	Understanding
10	Outline about virtual memory	BTL2	Understanding
11	What is the basic approach for page replacement	BTL3	Applying
12	Illustrate the use of Valid-Invalid Bits in Paging?	BTL3	Applying
13	What you mean by compaction? In which situation is it applied.	BTL3	Applying
14	Why page sizes are always power of 2?	BTL4	Analyzing
15	Is the problem of external fragmentation can be solved? justify	BTL4	Analyzing
16	How does the system discover thrashing?	BTL4	Analyzing
17	How much virtual memory should I set for 4GB RAM	BTL5	Evaluating
18	Evaluating the maximum number of pages needed If a system supports 16 bit address line and 1K page size.	BTL5	Evaluating
19	Formulate how long a paged memory reference takes if memory reference takes 200 nanoseconds .Assume a paging system with page table stored in memory	BTL6	Creating
20	Program containing relocatable code was created, assuming it would be loaded at address 0. In its code, the program refers to the following addresses: 50,78,150,152,154. If the program is loaded into memory starting at location 250, how do those addresses have to be adjusted?	BTL6	Creating
PART – B			
1	Discuss the following page replacement algorithm with an example i) Optimal (7) ii) LRU (6)	BTL1	Remembering
2	When page faults will occur? Discuss the actions taken by operating system during page fault(13)	BTL1	Remembering
3	Discuss situation under which the most frequently used page replacement algorithm generates fewer page faults than the least frequently used page replacement algorithm. Also discuss under which circumstances the opposite holds(13)	BTL1	Remembering
4	What is thrashing and explain the methods to avoid thrash(13)	BTL1	Remembering
5	Describe the LRU page replacement algorithm, assuming there are 3 frames and the page reference string is7 0 1 2 0 3 0 4 2 3 0 3 2 1 2 0 1 7 0 1. Find the number of page faults.(13)	BTL2	Understanding
6	Compare paging with segmentation in terms of the amount of memory required by the address translation structures in order to convert virtual addresses to physical addresses.(13)	BTL2	Understanding
7	Outline copy-on write feature and under what circumstances it is beneficial? What hardware support is needed to implement this feature?(13)	BTL2	Understanding
8	Explain about the difference between internal fragmentation and external fragmentation(13)	BTL3	Applying
9	Differentiate local and global page replacement algorithm.(13)	BTL3	Applying
10	Illustrate in detail about the free space management on I/O buffering and blocking(13)	BTL3	Applying

11	Explain why sharing a reentrant module is easier when segmentation is used than when pure paging is used with example.(13)	**BTL4**	**Analyzing**
12	Why are segmentation and paging sometimes combined into one scheme?(13)	**BTL4**	**Analyzing**
13	Explain about given memory management techniques. (i) Partitioned allocation (7) (ii) Paging and translation look-aside buffer(6)	**BTL5**	**Evaluating**
14	Consider the following page reference string: 1, 2, 3, 4, 2, 1, 5, 6, 2, 1, 2, 3, 7, 6, 3, 2, 1, 2, 3, 6. Identify the number of page faults would occur for the following replacement algorithms, assuming one, two, three, four, five, six, or seven frames? Remember all frames are initially empty, so your first unique pages will all cost one fault each. i). LRU replacement (4) ii). FIFO replacement (5) iii).Optimal replacement (4)	**BTL6**	**Creating**

PART – C

1	(i) Consider the following page reference string: 1,2, 3, 2, 5, 6, 3, 4, 6, 3, 7, 3, 1, 5, 3, 6, 3, 4, 2, 4, 3, 4, 5, 1 Indicate page faults and calculate total number of page faults and successful ratio for FIFO, optimal and LRU algorithms. Assume there are four frames and initially all the frames are empty. (12) ii)Explain the effect of thrashing. (3)	**BTL6**	**Creating**
2	(i) Explain in detail about paging in 32-bit and 64-bit architectures (5) (ii) Consider a system that allocated pages of different sizes to its processes. What are the advantages of such a paging scheme? What are modifications to the virtual memory system provide this functionality? (10)	**BTL6**	**Creating**
3	Explain paging scheme of memory management. What hardware support is needed for its implementation?(15)	**BTL5**	**Evaluating**
4	(i)Explain the difference between internal and external fragmentation. (7) (ii)Discuss situations in which the most frequently used (MFU) page replacement algorithm generates fewer page faults than the least recently used (LRU) page-replacement algorithm. Also discuss under what circumstances the opposite holds. (8)	**BTL4**	**Analyzing**

UNIT- IV : FILE SYSTEMS

File-System Interface: File concept – Access methods – Directory structure – File system mounting – Protection. File-System Implementation: Directory implementation – Allocation methods – Free-space management – efficiency and performance – recovery – log-structured file systems.

PART – A

Q.No.	**Questions**	**BT Level**	**Competence**
1	List out the major attributes and operations of a file system.	**BTL1**	**Remembering**
2	What is the advantage of bit vector approach in free space management?	**BTL1**	**Remembering**
3	What is boot control block?	**BTL1**	**Remembering**
4	Write Short notes on file system mounting.	**BTL1**	**Remembering**
5	List out the drawbacks in indexed allocation	**BTL1**	**Remembering**
6	Define UFD and MFD.	**BTL1**	**Remembering**
7	Give the disadvantages of Contiguous allocation.	**BTL2**	**Understanding**

8	Outline the difference between file and directory.	**BTL2**	**Understanding**
9	What is consistency checking?	**BTL2**	**Understanding**
10	Outline the contiguous allocation with linked allocation method.	**BTL2**	**Understanding**
11	How the information in the file can be accessed?	**BTL3**	**Applying**
12	What is relative block number?	**BTL3**	**Applying**
13	Enlist different types of directory structure.	**BTL3**	**Applying**
14	Do FAT file system advantageous? Justify your answer?	**BTL4**	**Analyzing**
15	Mention the common file types	**BTL4**	**Analyzing**
16	Analyze the backup and restore of a file system.	**BTL4**	**Analyzing**
17	Evaluate the various file access methods.	**BTL5**	**Evaluating**
18	How does DMA increase system concurrency?	**BTL5**	**Evaluating**
19	Identify the advantages of bit vector free space management	**BTL6**	**Creating**
20	Identify the two important function of virtual File System (VFS) layer in the concept of file system implementation.	**BTL6**	**Creating**
PART – B			
1	(i)Describe with a neat sketch about the various directory structure. (7) (ii)Describe in detail about free space management with neat examples.(6)	**BTL1**	**Remembering**
2	(i)Brief in detail the various allocation methods with their pros and cons (8) (ii)Brief the various procedures need to be followed in disk management(5)	**BTL1**	**Remembering**
3	i) Discuss about the various file access methods.(7) ii) With neat sketch explain about the: (6) a) Directory structure b) File sharing	**BTL1**	**Remembering**
4	Describe in detail about file sharing and protection.(13)	**BTL1**	**Remembering**
5	Outline in detail about the protection of file system.(13)	**BTL2**	**Understanding**
6	Discuss in detail about file attributes and file operation.(13)	**BTL2**	**Understanding**
7	(i)Why is it important to balance file system I/O among the disks and controllers on a system in a multitasking environment? (6) (ii) Discuss the advantages and disadvantages of supporting links to files that cross mount points. (7)	**BTL2**	**Understanding**
8	Illustrate an application that could benefit from operating system support for random access to indexed files. (13)	**BTL3**	**Applying**
9	(i)Explain why logging metadata updates ensures recovery of a file system after a file-system crash. (7) (ii)Explain the issues in designing a file system. (6)	**BTL3**	**Applying**
10	Explain in detail about tree structured and acyclic graph directories.(13)	**BTL3**	**Applying**
11	(i)In a variable partition scheme, the operating system has to keep track of allocated and free space. Suggest a means of achieving this. Describe the effects of new allocations and process terminations in your suggested scheme. (5) (ii) Explain in brief about different allocation methods with neat sketch. (8)	**BTL4**	**Analyzing**
12	Analyze the various file system mounting methods in detail	**BTL4**	**Analyzing**
13	Examine in detail about Directory and disk structure. (13)	**BTL5**	**Evaluating**
14	Consider a file system where a file can be deleted and its disk space Reclaimed while links to that file still exist. What problems may occur if a new file is created in the same storage area or with the same	**BTL6**	**Creating**

	absolute path name? How can these problems be avoided? (13)		
	PART – C		
1	Consider an example of an application in which data in a file should be accessed in the following order (i) Sequential (8) (ii) Random (7)	**BTL-6**	**Creating**
2	Evaluate how performance optimizations for file systems might result in difficulties in maintaining the consistency of the systems in the event of computer crashes. (15)	**BTL-5**	**Evaluating**
3	(i) Analyze in detail about the functions of files and file implementation. (8) (ii)Explain free space management with neat example. (7)	**BTL-4**	**Analyzing**
4	Evaluate some advantages and disadvantages of using SSDs as a caching tier and as a disk-drive replacement compared with using only magnetic disks. (15)	**BTL-5**	**Evaluating**
	UNIT - V: I/O SYSTEMS		
	I/O Systems – I/O Hardware – Application I/O interface – kernel I/O subsystem - streams – performance. Mass-Storage Structure: Disk scheduling – Disk management – Swap space management – disk attachment. Case study: I/O in Linux		
	PART – A		
Q.No.	**Questions**	**BT Level**	**Competence**
1	List out the disk scheduling algorithms?	**BTL1**	**Remembering**
2	Define Streams?	**BTL1**	**Remembering**
3	What are the advantages of caching?	**BTL1**	**Remembering**
4	Define rotational latency	**BTL1**	**Remembering**
5	Describe the typical pc bus structure	**BTL1**	**Remembering**
6	What is meant by interrupt driven I/O Cycle?	**BTL1**	**Remembering**
7	Give the advantages of polling.	**BTL2**	**Understanding**
8	Mention the various bus structures.	**BTL2**	**Understanding**
9	Summarize the advantages of swap space management?	**BTL2**	**Understanding**
10	Outline the system calls in Streams	**BTL2**	**Understanding**
11	Compare the synchronous and asynchronous streams	**BTL3**	**Applying**
12	Lists the advantages of blocking and non blocking I/O	**BTL3**	**Applying**
13	Illustrate the various RAID levels	**BTL3**	**Applying**
14	Why rotational latency is usually not considered in disk scheduling?	**BTL4**	**Analyzing**
15	Analyze why it is important to scale up system bus and device speeds as CPU speed increases?	**BTL4**	**Analyzing**
16	Explain device reservation?	**BTL4**	**Analyzing**
17	How SSTF is more optimal than other disk scheduling algorithms?	**BTL5**	**Evaluating**
18	Why Disk Scheduling necessary	**BTL5**	**Evaluating**
19	State the typical bad-sector transactions	**BTL6**	**Creating**
20	Tell the function of Conflict Resolution mechanism?	**BTL6**	**Creating**
	PART – B		
1	(i) What are the advantages of polling (3) (ii) Explain in detail about application I/O Interface (10)	**BTL1**	**Remembering**
2	Discuss in detail about the streams with a neat sketch (13)	**BTL1**	**Remembering**

3	Discuss in detail about the various disk attachment methods. (13)	**BTL1**	**Remembering**
4	Demonstrate in detail about kernel I/O Subsystems (13)	**BTL1**	**Remembering**
5	Describe in detail about interrupts .(13)	**BTL2**	**Understanding**
6	Summarize in detail about swap space management(13)	**BTL2**	**Understanding**
7	Summarize briefly about the RAID structure in disk management with various RAID levels of organization in detail (13)	**BTL2**	**Understanding**
8	Illustrate the I/O hardware with a typical pc bus structure (13)	**BTL3**	**Applying**
9	Explain in detail about DMA Structure (13)	**BTL3**	**Applying**
10	Illustrate in detail about Disk management (13)	**BTL3**	**Applying**
11	State and explain the FCFS, SSTF and SCAN disk scheduling with examples.(13)	**BTL4**	**Analyzing**
12	Explain in detail about mass storage structures. (13)	**BTL4**	**Analyzing**
13	Suppose that the disk drive has 5000 cylinders number 0 to 4999. The drive is serving a request at cylinder 143. The queue of pending request in FIFO order is: 86,1470,913,1774,948,1509.1022,1750,130 starting from the head position, what is the total distance (cylinders) that the disk arm moves to satisfy all the pending requests for each of the disk scheduling algorithms? FCFS,SSTF,SCAN ,LOOK,C-SCAN,C-LOOK. Explain the pros and cons of all disks scheduling algorithms (13)	**BTL5**	**Evaluating**
14	(i) Explain about kernel I/O subsystems and transforming I/O to hardware operations. (7) (ii)On a disk with 1000 cylinders, numbers 0 to 999, compute the number of tracks, the disk arm must move to satisfy the entire requests in the disk queue. Assume the last request service was at track 345 and the head is moving toward track 0. The queue in FIFO order contains requests for the following tracks: 123, 874, 692, 475, 105, and 376. Find the seek length for the following scheduling algorithm. (6) a)SSTF b) LOOK c) CSCAN	**BTL6**	**Creating**
	PART – C		
1	On a disk with 200 cylinders, numbered 0 to199. Compute the number of tracks the disk arm must move to satisfy the entire request in the disc queue. Assume the last request received at track 100. The queue in FIFO order contains requests for the following tracks 55, 58, 39, 18, 90, 160, 150, 38, 184. Perform the computation to find the seek time for the following disk scheduling algorithms (i)FCFS (3) (ii) SSTF (3) (iii)SCAN (3) (iv)C-SCAN (3) (v)LOOK (3)	**BTL6**	**Creating**
2	How does a DMA increases system concurrency? How does it complicate the hardware design? (15)	**BTL5**	**Evaluating**
3	Distinguish between a STREAMS driver and a STREAMS module. (15)	**BTL5**	**Evaluating**
4	Why rotational latency usually not considered in disk scheduling. How would you modify SSTF,SCAN and C-SCAN to include latency optimization? (15)	**BTL6**	**Creating**