Unit 4

PART – A: (Multiple Choice Questions) 2x10=20

Marks

1	RAID level _____ is also known as block interleaved parity organisation and uses block level striping and keeps a parity block on a seperate disk. A. 1 B. 2 C. 3 D. 4
2	In RAID level 4, one block read, accesses _____. A. only one disk B. all disks simultaneously C. all disks sequentially D. None of these
3	Access in which records are accessed from and inserted into file, is classified as 1. direct access 2. sequential access 3. random access 4. duplicate access
4	Preparation of disc for subsequent file storage is classified as 1. disc format 2. disc address 3. disc footer **4.** disc header
5	Secondary storage memory is basically 1. volatile memory 2. non volatile memory 3. backup memory 4. impact memory
6	In fixed head discs, sum of rotational delay and transfer time is equals to 1. access time 2. delay time

	3. processing time 4. storage time
7	Piece of time taken by disc to rotate and read data from right place is classified as 1. rotational delay 2. access delay 3. seek time delay **4.** reversal delay
8	Which of then following is example of direct access? 1. magnetic disc 2. floppy disc 3. program tape **4.** plain disc
9	File which is created to carry out processing of data is classified as 1. master file 2. transaction file 3. particular file **4.** reference file
10	Possible dangers and threats for files are that fie can be 1. destroyed 2. modified 3. accessed **4.** all of above

UNIT-4

PART – B: (Short Answer Questions)

1	**Elaborate about the file allocation techniques.** **Contiguous Allocation** In this scheme, each file occupies a contiguous set of blocks on the disk **Linked List Allocation** In this scheme, each file is a linked list of disk blocks which **need not be** contiguous. The disk blocks can be scattered anywhere on the disk. **Indexed Allocation**

	In this scheme, a special block known as the **Index block** contains the pointers to all the blocks occupied by a file. Each file has its own index block
2	**Write short note on SCAN scheduling**. In SCAN disk scheduling algorithm, head starts from one end of the disk and moves towards the other end, servicing requests in between one by one and reach the other end. Then the direction of the head is reversed and the process continues as head continuously scan back and forth to access the disk.
3	**How CLOOK scheduling works?** • Circular-LOOK Algorithm is an improved version of the LOOK Algorithm. • Head starts from the first request at one end of the disk and moves towards the last request at the other end servicing all the requests in between. • After reaching the last request at the other end, head reverses its direction. • It then returns to the first request at the starting end without servicing any request in between. • The same process repeats.
4	**Differentiate between sequential access and direct access**. Sequential access must begin at the beginning and access each element in order, one after the other. Direct access allows the access of any element directly by locating it by its index number or address. Arrays allow direct access. Magnetic tape has only sequential access, but CDs had direct access. If you are on a railroad train, to go from one car to another you must use sequential access. But when you board the train initially you have direct access. Direct access is faster than sequential access, but it requires some external mechanism (array index, file byte number, railroad platform)
5	**Explain the attributes of a file.** • **Name** . It is the only information which is in human-readable form. • **Identifier**. The file is identified by a unique tag(number) within file system. • **Type**. It is needed for systems that support different types of files. • **Location**. Pointer to file location on device. • **Size**. The current size of the file. • **Protection**. This controls and assigns the power of reading, writing, executing. • **Time, date, and user identification**. This is the data for protection, security, and usage monitoring.
6	**Differentiate between contiguous allocation and linked allocation.** Linked allocation is also called chained allocation In each chain the block contains the pointer to the next link Reallocation is possible. In contiguous allocation allocation is used using the first bit.

	There is a need for compaction When files grow there is a problem with the files. In indexed allocation contains separate level index for each file. There is only one entry for each.
7	**Explain briefly about DMA**.. **Direct Memory Access** (DMA) transfers the block of data between the *memory* and *peripheral devices* of the system, **without the participation** of the **processor**. The unit that controls the activity of accessing memory directly is called a **DMA controller**. The processor **relinquishes the system bus** for a few clock cycles. So, the DMA controller can accomplish the task of data transfer via the system bus
8	**What do you mean by seek time?** Seek time is the time taken for a hard disk controller to locate a specific piece of stored data. Other delays include transfer time (data rate) and rotational delay (latency)

UNIT-4

PART – C: (Long Answer Questions)

1	**What is a file? Explain various file allocation techniques with their advantages and disadvantages.** A **file** is a named collection of related information that is recorded on secondary storage such as magnetic disks, magnetic tapes and optical disks. In general, a **file** is a sequence of bits, bytes, lines or records whose meaning is defined by the **files** creator and user. There are mainly three methods of file allocation in the disk. Each method has its advantages and disadvantages. Mainly a system uses one method for all files within the system. • Contiguous allocation • Linked allocation • Indexed allocation **Contiguous allocation** In this scheme, a file is made from the contiguous set of blocks on the disk. Linear ordering on the disk is defined by the disk addresses. • Each file in the disk occupies a contiguous address space on the disk. • In this scheme, the address is assigned in the linear fashion. • The is very easy to implement the contiguous allocation method. • In the contiguous allocation technique, external fragmentation is a major issue.

Advantages:

- In the contiguous allocation, sequential and direct access both are supported.
- For the direct access, the starting address of the kth block is given and further blocks are obtained by b+K,
- This is very fast and the number of seeks is minimal in the contiguous allocation method.

Disadvantages:

- Contiguous allocation method suffers internal as well as external fragmentation.
- In terms of memory utilization, this method is inefficient.
- It is difficult to increase the file size because it depends on the availability of contiguous memory.

Linked allocation

The problems of contiguous allocation are solved in the linked allocation method. In this scheme, disk blocks are arranged in the linked list form which is not contiguous.

Advantages:

1. In terms of the file size, this scheme is very flexible.
2. We can easily increase or decrease the file size and system does not worry about the contiguous chunks of memory.
3. This method free from external fragmentation this makes it better in terms of memory utilization.

Disadvantages:

1. In this scheme, there is large no of seeks because the file blocks are randomly distributed on disk.
2. Linked allocation is comparatively slower than contiguous allocation.
3. Random or direct access is not supported by this scheme we cannot access the blocks directly.
4. The pointer is extra overhead on the system due to the linked list.

Indexed Allocation

In this scheme, a special block known as the index block contains the pointer to all the blocks occupied by a file. each file contains its index which is in the form of an array of disk block addresses.

Advantages:

1. This scheme supports random access of the file.
2. This scheme provides fast access to the file blocks.
3. This scheme is free from the problem of external fragmentation.

Disadvantages:

1. The pointer head is relatively greater than the linked allocation of the file.
2. Indexed allocation suffers from the wasted space.

	3. For the large size file, it is very difficult for single index block to hold all the pointers. 4. For very small files say files that expend only 2-3 blocks the indexed allocation would keep on the entire block for the pointers which is insufficient in terms of memory utilization.
2	**Suppose that the head of a moving hard disk with 200 tracks, numbered 0 to 199, is currently serving a request at track 143 and has just finished a request at 125. The queue of request is kept in the FIFO order 86, 147, 91, 177, 94, 150, 102, 175 and 130.** **What is the total number of head movements needed to satisfy these requests for the following disk-scheduling algorithms?** **1. FCFS Scheduling** **2. SSTF Scheduling** **3. SCAN Scheduling** (a) FCFS: 565. $(143 \rightarrow 86 \rightarrow 147 \rightarrow 91 \rightarrow 177 \rightarrow 94 \rightarrow 150 \rightarrow 102 \rightarrow 175 \rightarrow 130)$ (b) SSTF: 162. $143 \rightarrow 147 \rightarrow 150 \rightarrow 130 irighrarrow 102 \rightarrow 94 \rightarrow 91 \rightarrow 86 \rightarrow 175 \rightarrow 177)$ (c) SCAN: 169. $143 \rightarrow 147 \rightarrow 150 \rightarrow 175 \rightarrow 177 \rightarrow 199 \rightarrow 130 \rightarrow 102 \rightarrow 94 \rightarrow 91 \rightarrow 86)$
3	**Discuss the linked allocation and index allocation schemes for a file allocation. Compare the index allocation scheme with the contiguous allocation scheme.** **Linked allocation** The problems of contiguous allocation are solved in the linked allocation method. In this scheme, disk blocks are arranged in the linked list form which is not contiguous. The disk block is scattered in the disk. In this scheme, the directory entry contains the pointer of the first block and pointer of the ending block. These pointers are not for the users. For example, a file of six blocks starts at block 10 and end at the block. Each pointer contains the address of the next block. When we create a new file we simply create a new entry with the linked allocation. Each directory contains the pointer to the first disk block of the file. when the pointer is nil then it defines the empty file. **Indexed Allocation** In this scheme, a special block known as the index block contains the pointer to all the blocks occupied by a file. each file contains its index which is in the form of an array of disk block addresses. The ith entry of index block point to the ith block of the file. The address of the index block is maintained by the directory. When we create a file all pointer is set to nil. A block is obtained from the free space manager when the first ith block is written. When the index block is very small it is difficult to hold all the pointers for the large file. to deal with this issue a mechanism is available. Mechanism includes the following: • Linked scheme • Multilevel scheme • Combined scheme

4	**Answer the following** **1. Disk Structure** **2. RAID Structure.** 1. The actual physical details of a modern hard disk may be quite complicated. Simply, there are one or more surfaces, each of which contains several tracks, each of which is divided into sectors. There is one read/write head for every surface of the disk. Also, the same track on all surfaces is knows as a 'cylinder'. When talking about movement of the read/write head, the cylinder is a useful concept, because all the heads (one for each surface), move in and out of the disk together. 2. RAID, or "Redundant Arrays of Independent Disks" is a technique which makes use of a combination of multiple disks instead of using a single disk for increased performance, data redundancy or both. The term was coined by David Patterson, Garth A. Gibson, and Randy Katz at the University of California, Berkeley in 1987. **Key evaluation points for a RAID System** • **Reliability:** How many disk faults can the system tolerate? • **Availability:** What fraction of the total session time is a system in uptime mode, i.e. how available is the system for actual use? • **Performance:** How good is the response time? How high is the throughput (rate of processing work)? Note that performance contains a lot of parameters and not just the two. • **Capacity:** Given a set of N disks each with B blocks, how much useful capacity is available to the user?
5	**Describe Caching, Spooling and Buffering services of kernel I/O sub-system.** • **Buffering** − Kernel I/O Subsystem maintains a memory area known as **buffer** that stores data while they are transferred between two devices or between a device with an application operation. Buffering is done to cope with a speed mismatch between the producer and consumer of a data stream or to adapt between devices that have different data transfer sizes. • **Caching** − Kernel maintains cache memory which is region of fast memory that holds copies of data. Access to the cached copy is more efficient than access to the original. • **Spooling and Device Reservation** − A spool is a buffer that holds output for a device, such as a printer, that cannot accept interleaved data streams. The spooling system copies the queued spool files to the printer one at a time. In some operating systems, spooling is managed by a system daemon process. In other operating systems, it is handled by an in kernel thread.
B	**Explain contiguous, indexed and linked allocation of disk space.** **Contiguous Allocation: –** Contiguous allocation is one of the most used methods for allocation. Contiguous allocation means we allocate the block in such a manner, so that in the hard disk, all the blocks get the contiguous physical block. We can see in the below figure that in the directory, we have three files. In the table, we have mentioned the starting block and the length of all the files. We can see in the table that for each file, we allocate a contiguous block. **Linked allocation**

The problems of contiguous allocation are solved in the linked allocation method. In this scheme, disk blocks are arranged in the linked list form which is not contiguous. The disk block is scattered in the disk. In this scheme, the directory entry contains the pointer of the first block and pointer of the ending block. These pointers are not for the users. For example, a file of six blocks starts at block 10 and end at the block. Each pointer contains the address of the next block. When we create a new file we simply create a new entry with the linked allocation. Each directory contains the pointer to the first disk block of the file. when the pointer is nil then it defines the empty file.

Indexed Allocation

In this scheme, a special block known as the index block contains the pointer to all the blocks occupied by a file. each file contains its index which is in the form of an array of disk block addresses. The ith entry of index block point to the ith block of the file. The address of the index block is maintained by the directory. When we create a file all pointer is set to nil. A block is obtained from the free space manager when the first ith block is written. When the index block is very small it is difficult to hold all the pointers for the large file. to deal with this issue a mechanism is available. Mechanism includes the following:

- Linked scheme
- Multilevel scheme
- Combined scheme