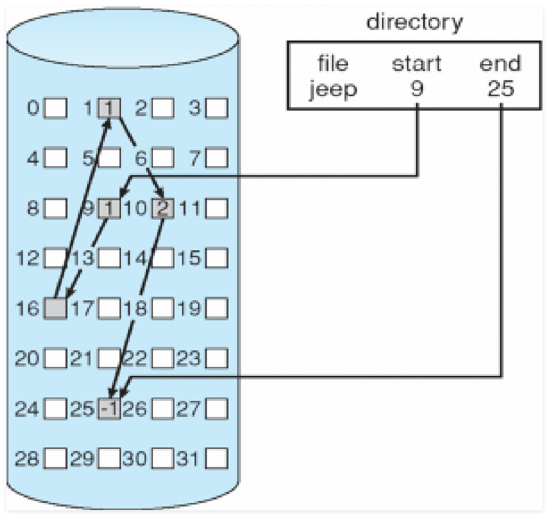
MODEL ANSWERS

ANS.1

Linked Allocation

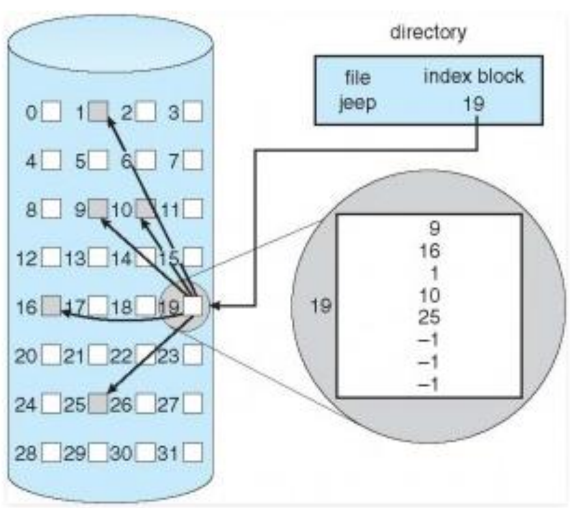
In this scheme, each file is a linked list of disk blocks which need not be contiguous. The disk blocks can be scattered anywhere on the disk.
The directory entry contains a pointer to the starting and the ending file block. Each block contains a pointer to the next block occupied by the file.

The file 'jeep' in following image shows how the blocks are randomly distributed. The last block (25) contains -1 indicating a null pointer and does not point to any other block.



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. The ith entry in the index block contains the disk address of the ith file block. The directory entry contains the address of the index block as shown in the image:

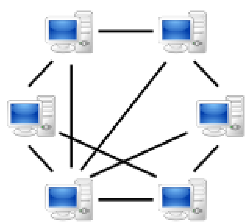


ANS.2

Peer-to-Peer

Peer-to-peer (P2P) computing or networking is a distributed application architecture that partitions tasks or workloads between peers. Peers are equally privileged, equipotent participants in the application. They are said to form a peer-to-peer network of nodes.

Peers make a portion of their resources, such as processing power, disk storage or network bandwidth, directly available to other network participants, without the need for central coordination by servers or stable hosts. Peers are both suppliers and consumers of resources, in contrast to the traditional client-server model in which the consumption and supply of resources is divided. Emerging collaborative P2P systems are going beyond the era of peers doing similar things while sharing resources, and are looking for diverse peers that can bring in unique resources and capabilities to a virtual community thereby empowering it to engage in greater tasks beyond those that can be accomplished by individual peers, yet that are beneficial to all the peers.

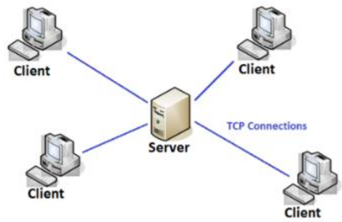


Client-Server

1. Client/server architecture is a producer-consumer computing architecture where the server acts as the producer and the client as a consumer.

2. Client/server architecture is a computing model in which the server hosts, delivers and manages most of the resources and services to be consumed by the client. This type of architecture has one or more client computers connected to a central server over a network or Internet connection. This system shares computing resources.

3. This computing model is especially effective when clients and the server each have distinct tasks that they routinely perform.For example : In hospital data processing, a client computer can be running an application program for entering patient information while the server computer is running another program that manages the database in which the information is permanently stored.



ANS.3a

```
Var         flag : array [0…1] of Boolean;
            Turn : 0..1;
Begin
            Flag[0] = false;
            Flag[1] = false;
Parbegin
    Repeat                                  Repeat
            Flag[0] = true                          Flag[1] = true
            Turn = 1                                Turn = 0
            While flag[1] && turn==1                While flag[0] && turn==0
              Do {nothing};                           Do {nothing};
            {Critical Section}                      {Critical Section}
            Flag[0] = false;                        Flag[1] = false;
            {Remainder}                             {Remainder}
    Forver;                                                 Forver;
Parend
end
```

ANS. 3b

Concurrency: It can be defined as the illusion of being in Parallel (using single processing unit)

Race Condition It can be defined as the issue b/w the various processes that may exist while writing a shared variable; in a non-synchronous environment.

Critical Section

The part of the program, where a variable changes its value. or where it is re-defined.

ANS.4

Conventional Page Table Size

$$\left(2^{64} / 1\,\text{kbyte} \right) \times 2\ \text{byte} = 2^{52}\ \text{bytes}$$

Inverted page Table Size

$$= 64 \times 2\,\text{byte}$$

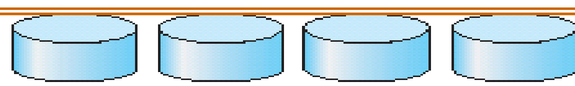
$$= 128\ \text{byte}$$

ANS. 5

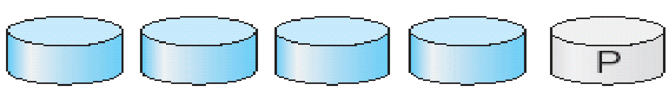
(a) Disk capacity = [#platters * (#surfaces/platter)*(#tracks/surface)* (#sectors/track)*sector capacity]*#disk
=[6*2*32*64*512bytes]*2
25165824 bytes

(b) # record = disk capacity /record size = 25165824/(2*152 bytes) = 24576

ANS. 6

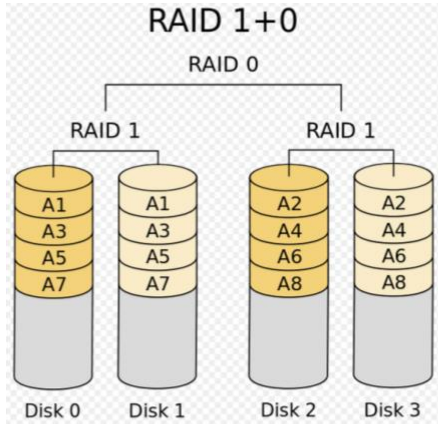


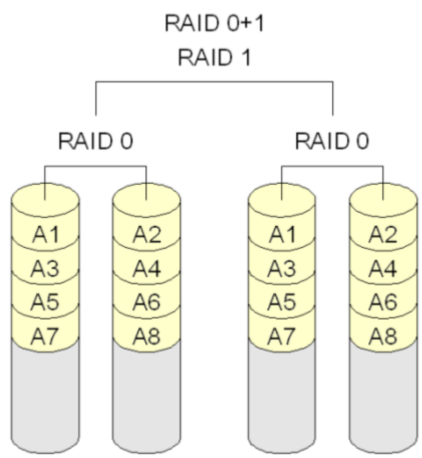
(a) RAID 0: non-redundant striping.



(d) RAID 3: bit-interleaved parity.

(f) RAID 5: block-interleaved distributed parity.



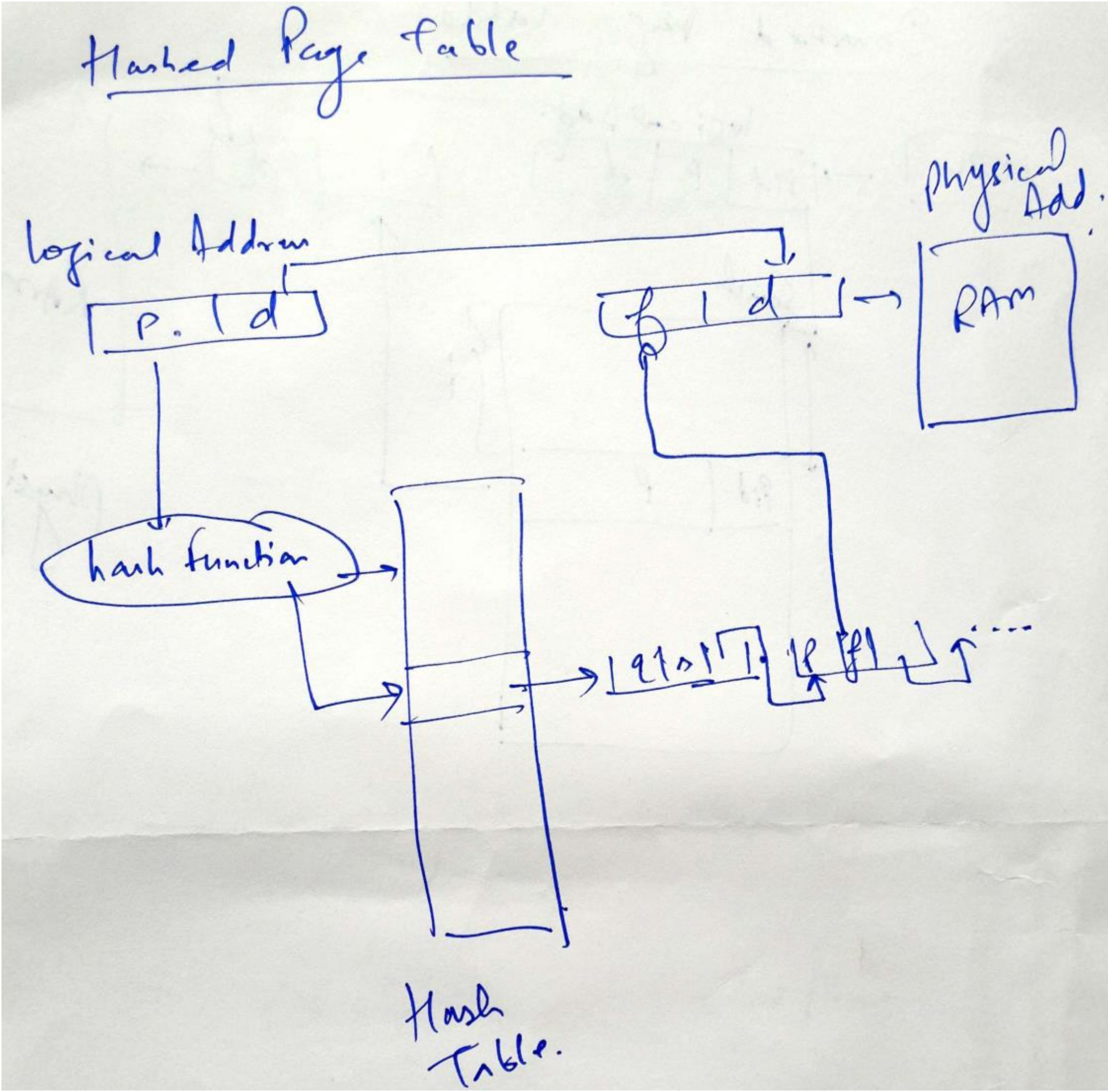


ANS. 7

The array has 200 x 200 = 40,000 elements, each requiring 4 bytes of storage. 160,000 bytes occupy 625 Pages.

The program accesses the locations in the array in the order they are stored. Thus, it will swap-in the instruction page, and each of the 625 data pages once, for a total of 626 page faults.

ANS. 8



Hashed Page table
physical Add.
logical Address
p. | d
f | d
RAM
hash function
Hash Table.

ANS. 9

Optimal Replacement

Before	Page	fault	After
-,-,-	0	*	0,-,-
0,-,-	9	*	9,0,-
9,0,-	0		9,0,-
9,0,-	1	*	1,9,0
1,9,0	8	*	8,1,9
8,1,9	1		8,1,9
8,1,9	8		8,1,9
8,1,9	7	*	7,8,1
7,8,1	8		7,8,1
7,8,1	7		7,8,1
7,8,1	1		7,8,1
7,8,1	2	*	2,7,8
2,7,8	8		2,7,8
2,7,8	2		2,7,8
2,7,8	7		2,7,8
2,7,8	8		2,7,8
2,7,8	2		2,7,8
2,7,8	3		3,2,8
3,2,8	8	*	3,2,8
3,2,8	3		3,2,8

ANS. 10

SSTF The tracks traveled to will be:
345, 376, 475, 692, 874, 123 and 105

Total Distance:

529 + 769 = 1298

SCAN Tracks Traveled:

345, 123, 105, 0, 376, 475, 692, 874

Total Distance:

345 + 874 = 1219

LOOK Tracks traveled 345, 123, 105, 376, 475, 692, 874

Total Distance: 240 + 769 = 1009

C-SCAN

Tracks Traveled: 345, 123, 105, 0, 999, 874, 692, 475, 376

Total Distance: 345 + 999 + 623 = 1967.

C-LOOK

Tracks Traveled: 345, 123, 105, 874, 692, 475, 376

C-Look: = 240 + 769 + 498 = 1507

ANS. 11

No. of Surfaces = 8

Outer Track Dia. = 16 cm

Inner ————— = 6 cm

Track Total Width = (16-6)/2 = 5 cm

Track separation = 0.2 mm

No. of tracks per surface = (5/0.2) × 10 = 250

Bits needed for Addressing Surface = 3

————— Tracks = 8

————— Sector = 5

Total = 27

3+8+5 = 16

Remaining = 27-16 = 11

used to address Data of each sector.

Sector Capacity = 2^{11}

= 2KB

Rotation = 3600 rpm = 3600/60 rps = 60 rps.

Data Transfer Rate = Revolution per Second × Track Cap

= 60 × 32 × 2 KB per Second

= 3840 KB/sec.

ANS. 12

Total Resources = [8 5 7] + [2 1 0]

= [10 6 7]

Need = MAX − Allocation.

$$= \begin{matrix} 3 & 2 & 1 \\ 1 & 1 & 0 \\ 5 & 0 & 1 \\ 7 & 3 & 3 \\ 10 & 1 & 1 \end{matrix}$$

Safety Algo.

~~work = available = 2 1 0 , finish = 0 0 0 0 0~~

No sequence is possible, Hence the state is UNSAFE

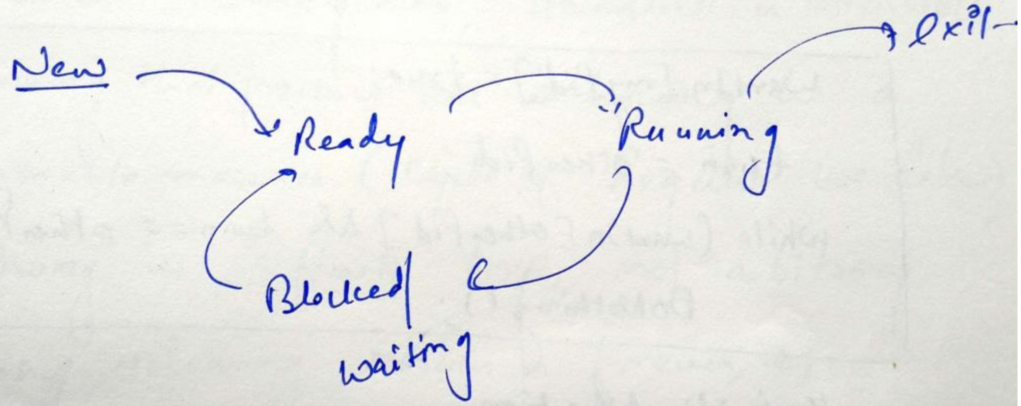
Deadlock Detection Algo.

No Deadlock and Sequence is

$\langle P_1 \; P_4 \; P_0 \; P_2 \; P_3 \rangle$

ANS. 13

There can be zero processes in any of the three states. It is possible for all the processes to be blocked while waiting on I/O operations, leaving no processes in the ready and running states. All processes can be either in the ready or run state, leaving no process blocked.



ANS. 14

Yes, there will be a deadlock.

There are 3 processes and 4 resources and max need of each process is 3

Scenario 1

2 processes P1 and P2 are allocated 2 resources each. Now the system will enter deadlock as all are waiting for their need to be satisfied.

Scenario 2

Each process gets 1 resource. But need is still 2 and we have only 1 instance of resource left.