**A.**

1. What is a character special file?

It is a type of file in UNIX operating system where system transmits data one character at a time.

2. What is a block special file?

It is a type of file where system transmits data in the form of blocks. It is used in disks read and write.

3. What is metadata?

Metadata is data about the data; it provides information about the data like size and structure, sometime it provides information about version number.

4. What is a FAT? What is kept there?

FAT is a File Allocation Table that the operating system uses to locate files on a disk. Due to fragmentation, a file may be divided into many sections that are scattered around the disk. The FAT keeps track of all these pieces.

5. What is an i-node? What is kept there?

An i-node is a data structure on a that stores basic information about a regular file, directory. File attributes and address of the file’s are kept in the i-node.

**B.**

1.

/etc/passwd

/./etc/passwd

/././etc/passwd

/etc/../etc/passwd

/etc/../etc/../etc/passwd

2. One way file extension corresponds to a file type and to some program that handles that type. Another way is to remember which program created the file and run that program.

3. UNIX Systems load the program directly in memory and began executing at word 0 (magic number). To avoid trying to execute the header as code, the magic number was a BRANCH instruction with a target address just above the header.

4. Operating system cares about record length when files can be structured as records with keys at a specific position within each record and it is possible to ask for a record with a given key.

5. No.

6. The mapped portion of the file must start at a page boundary and be an integral number of pages in length. Each mapped page uses the file itself as backing store. Unmapped memory uses a scratch file or partition as backing store.

7. Using file names.

8. One way is to add an extra parameter to the read system call that tells what address to read from. So, every read has a potential for doing a seek within the file.

13. It finds the address of the first block in the directory entry, then it follows the chain of block pointers in the FAT until it has located the block it needs, and then it remembers this block number for the next read system call.

17. The bitmap requires B bits. The free list requires DF bits. The free list requires fewer bits if DF < B. Alternatively, the free list is shorter if F/B < 1/D, where F/B is the fraction of blocks free. For 16-bit disk addresses, the free list is shorter if 6 percent or less of the disk is free.

18. The recovery solution is to make a list of all the blocks in all the files and take the complement as the new free list. In UNIX this can be done by scanning all the i-nodes. In FAT file system, the problem cannot occur because there is no free list.

21. In (a) and (b), 21 would not be marked. In (c), there would be no change. In (d), 21 would not be marked.

25. At 15,000 rpm, the disk takes 4 msec to go around once. The average access time (in msec) to read k bytes is then 8 + 2 + (k /262144) × 4. For blocks of 1 KB, 2 KB, and 4 KB, the access times are 10.015625 msec, 10.03125 msec,

and 10.0625 msec, respectively (hardly any different). These give rates of about 102,240 KB/sec, 204,162 KB/sec, and 407,056 KB/sec, respectively.

26. If all files were 1 KB, then each 2-KB block would contain one file and 1 KB of wasted space. Putting two files in a block is not allowed because the unit used to keep track of data is the block. This leads to 50 percent wasted space.

29. 16,843,018 blocks = 16.06 GB.