1. What is mean by file system?

Ans:In Any Operating System the Concept of file is defined as ”Unformatted Uniform stream of Byte”.

1. Which file systems are used by Linux and Windows operating systems?

Ans: Windows mainly support FAT (File Allocation Table) and NTFS (New Technology File system). Windows NT 4.0, Windows 200, Windows XP, Windows .NET server and Windows workstation use NTFS as their preferred file system. Still, FAT can be used with floppy disks and older Windows versions (for multi-boot systems). FAT is the initial file system used in Windows. FAT was used with DOS, and its three versions are FAT12, FAT16 and FAT32. The number of bits used to identify a cluster is the number that is used as the suffix in the name. FAT12, FAT16 and FAT32 have 32MB, 4GB and 32GB as the maximum partition sizes.

A variety of files systems can be sued with Linux. Commonly used file systems are ext\* family (ext, ext2, ext3 and ext4) and XFS. Silicon Graphics developed XFS, which is a journaling system with high performance. The ext (extended file system) was developed in early 1990’s. It was the first file system used in Linux operating system. Remy Card developed it by getting inspiration from the UFS (UNIX File System).

1. What are the parts of the file system?

Ans:- A file system is a process of managing how and where data on a storage disk, which is also referred to as file management or FS

* NTFS.
* FAT.
* exFAT.
* HFS Plus.
* EXT.

1. Explain UAREA and its contents.

Ans: In addition to the text, data, and stack segments, the operating system also maintains for each process a region called the u area (user area). The u area contains information specific to the process (e.g., open files, current directory, signal actions, accounting information) and a system stack segment for process use. If the process makes a system call (e.g., the system call to write in the function main in Program 1.1), the stack frame information for the system call is stored in the system stack segment. Again, this information is kept by the operating system in an area that the process does not normally have access to. Thus, if this information is needed, the process must use special system calls to access it. Like the process itself, the contents of the u area for the process are paged in and out by the operating system.

1. Explain the use of the File Table and its contents.

Ans:In the traditional implementation of Unix, file descriptors index into a per-process file descriptor table maintained by the kernel, that in turn indexes into **a system-wide table of files opened by all processes**, called the file table.

1. Explain the use of InCore inode Table and its use.

Ans:- What is incore inode table?

In - core inode refers to **inode which is present in the main memory**. The kernel uses it whenever a process wants to manipulate a file in the secondary memory.It is a dynamic entity. On- disk inode is a static entity & is present only in the Secondary memory. HTH.

Use of Incore inode table:

An inode is the basis of a Linux file system. It **manages file and directory metadata** and is essential for the functioning of a file system. The inode is required to check the file system object locations, modification dates, size, and other relevant information

1. What is mean by inode?

Ans:- The **inode** (index node) is a data structure in a [Unix-style file system](https://en.wikipedia.org/wiki/Unix_filesystem) that describes a [file-system](https://en.wikipedia.org/wiki/File_system) object such as a [file](https://en.wikipedia.org/wiki/Computer_file) or a [directory](https://en.wikipedia.org/wiki/Directory_(computing)). Each inode stores the attributes and disk block locations of the object's data.[[1]](https://en.wikipedia.org/wiki/Inode#cite_note-1) File-system object attributes may include [metadata](https://en.wikipedia.org/wiki/Metadata) (times of last change,[[2]](https://en.wikipedia.org/wiki/Inode#cite_note-2) access, modification), as well as owner and [permission](https://en.wikipedia.org/wiki/File_system_permissions) data.[[3]](https://en.wikipedia.org/wiki/Inode#cite_note-3)

A directory is a list of inodes with their assigned names. The list includes an entry for itself, its parent, and each of its children.

1. What are the contents of Superblock?
2. Ans:- The superblock contains **information on the size of the file system, the number of inodes, the number of data blocks, the free and used inodes, and the block size for the file system**. The superblock is kept in memory and in multiple locations on disk for each file system.
3. What are the types of files?
4. Ans:There are 2 types of files 1. Regular File
5. 2. Special File .
6. What are the contents of the inode?

Ans: Contetnt of Inode :File Name, Inode Number, File Size, File Actual Size, File Type ,Buffer,Link Count, Refeernce Count,Permission, next Pointer

1. What is the use of a directory file?
2. A directory is a container that is used **to contain folders and files**. It organizes files and folders in a hierarchical manner.
3. How the operating system maintains security for files?

Security refers to providing a protection system to computer system resources such as CPU, memory, disk, software programs and most importantly data/information stored in the computer system. If a computer program is run by an unauthorized user, then he/she may cause severe damage to computer or data stored in it. So a computer system must be protected against unauthorized access, malicious access to system memory, viruses, worms etc. We're going to discuss following topics in this chapter.

1. Authentication
2. One Time passwords
3. Program Threats
4. System Threats
5. Computer Security Classifications
6. What happens when a user wants to open the file?
7. What happens when a user calls lseek system call?
8. Ans: **lseek()** system call repositions the read/write file offset i.e., it changes the positions of the read/write pointer within the file. In every file any read or write operations happen at the position pointed to by the pointer. lseek() system call helps us to manage the position of this pointer within a file.
9. What is the difference between library function and system call?

|  |  |
| --- | --- |
| **SYSTEM CALL** | **LIBRARY CALL** |
| 1. | A system call is a request made by the program to enter into kernel mode to access a process.. | A library call is a request made by the program to access a library function defined in a programming library. |
| 2. | In kernel mode the programs are directly accessible to the memory and hardware resources. | In user mode, the programs cannot directly accessible to the memory and hardware resources. |
| 3. | In system call, the mode is executed or switches from user mode to Kernel mode. | In library call, the mode is executed in user mode only. |

1. What is the use of this project?
2. Ans: Used to create Virtual file system on RAM .
3. What are the difficulties that you faced in this project?
4. Ans: This Project operating on Regular files only but special file system not supported by this project.
5. Is there any improvement needed in this project?