# Assignment No 6: File System

int fopen (char *filename, int flags);

Find the inode corresponding to the filename and since the file is open it is, loaded into memory into the inode table. A new open file table entry is created to refer to this file. The calling process allocates a file table entry to point to the open file table entry. The index of the file table entry i.e the file descriptor is returned to the process.

---

int fclose(int fd);

fclose: It uses the file descriptor from the system call and finds the open file table entry for this file. When we close the file, the inode is removed from the table and written back to disk.

---

int fread(int fd, void *buf, int nbytes);

It uses the open file table to find the inode referred to by use of given file descriptor. It then reads n bytes starting from the seek pointer into the process buffer memory, and updates the seek pointer. If the seek pointer is at the end of the file, EOF is returned to the process. File must be in open mode to read it

---

int fwrite(int fd, void *buf, int nbytes);

1) Its write number of bytes specified from a process's buffer memory into a file.

2)if write proceeds past the end of the file, additional blocks are found in a list of free blocks and added to the inode address entries.

3) The write call should also use the seek pointer. New data written into a file should start at the seek pointer, and when write is complete the seek pointer should be updated.

4) The blocks used for writing are marked to avoid rewriting.Done with the help of bitmask.

---

int fseek(int fd, int offset);

1)Allows to move the seek pointer in a file

2) Used in write function call.

3) used in read function call to move file pointer to desired location from where we want to start reading the file.

---

int fcreate(char *filename, int mode);

1)It allocates a free inode to the file and a new open file table entry is created to refer to this file.The file descriptor which is index of the file table entry is returned to the process.