Swati Malt

Computer Science and Engineering

II Semester

File Handeling

Streams –

The C file system is designed to work with a wide variety of devices, including terminals, disk drives, and tape drives.

Even though each devices is very different the buffered file system transforms each into a logical device called a stream.

It is of two types —

* Text stream — A text stream is a sequence of characters.

* Binary stream — A binary stream is a sequence of bytes that has a one-to-one correspondence to the bytes in external device — that is, no character translation occur.

Files —

In C, a file may be anything from a disk file to a terminal or printer. You associate a stream with a specific file by performing an open operation. Once a file is open, information can be exchanged between it and your program. The 'C' I/O system provides a level of abstraction between the programmer and the device. This abstraction is called a **stream**, and the actual device is called a **file**.

File Pointer

The file pointer is a pointer to a structure of type FILE. It points to information that defines various things about the file, including its name, status, and the current position of the file.

To obtain a file pointer variable, use a statement like this:

```
FILE *fp;
```

Opening a File

The fopen() function opens a stream for use and links a file with that stream. Then it returns file pointer associated with that file.

The fopen() function has this prototype,

```
FILE *fopen(const char *filename, const char *mode);
```

Where filename is a pointer to a string of characters that make up a valid filename and may include a path specification.

The string pointed to by mode determines how the file will be opened. Below table shows the legal values for mode:

Modes	Meaning
r	Open a text file for reading.
w	Create a text file for writing.
a	Append to a text file.
rb	Open a binary file for reading.
wb	Create a binary file for writing.
ab	Append to a binary file.
r+	Open a text file for read/write.
w+	Create a text file for read/write.
a+	Append or create a text file for read/write.
r+b	Open a binary file for read/write.
w+b	Create a binary file for read/write.
a+b	Append or create a binary file for read/write.

Closing a File

The fclose( ) function closes a stream that was opened by a call to fopen( ).
It writes any data still remaining in the disk buffer to the file and does a formal operating-system-level close on the file.
Failure to close a stream invites all kinds of trouble, including lost data, destroyed files and possible intermittent errors in your program.
The fclose( ) function has this prototype,

```
int fclose(FILE *fp);
```

where fp is the file pointer returned by the call to fopen( ). A return value of zero signifies a successful close operation.