

LECTURE NOTE 33

FILE

A File is a collection of data stored on a secondary storage device like hard disk. File operation is to combine all the input data into a file and then to operate through the C program. Various operations like insertion, deletion, opening closing etc can be done upon a file. When the program is terminated, the entire data is lost in C programming. If you want to keep large volume of data, it is time consuming to enter the entire data. But, if file is created these information can be accessed using few commands.

There are large numbers of functions to handle file I/O in C language. In this tutorial, you will learn to handle standard I/O(High level file I/O functions) in C. High level file I/O functions can be categorized as:

1. Text file
2. Binary file

A file can be open in several modes for these operations. The various modes are:

r	open a text file for reading
w	truncate to zero length or create a text file for writing
a	append; open or create text file for writing at end-of-file
rb	open binary file for reading
wb	truncate to zero length or create a binary file for writing
ab	append; open or create binary file for writing at end-of-file
r+	open text file for update (reading and writing)
w+	truncate to zero length or create a text file for update
a+	append; open or create text file for update
r+b or rb+	open binary file for update (reading and writing)
w+b or wb+	truncate to zero length or create a binary file for update
a+b or ab+	append; open or create binary file for update

fopen and freopen opens the file whose name is in the string pointed to by filename and associates a stream with it. Both return a pointer to the object controlling the stream, or if the open operation

fails a null pointer. The error and end-of-file(EOF) indicators are cleared, and if the open operation fails error is set. freopen differs from fopen in that the file pointed to by stream is closed first when already open and any close errors are ignored.

Q1. Write a program to open a file using fopen().

Ans:

```
#include<stdio.h> void
main()
{
fopen()
file *fp;

fp=fopen("student.DAT", "r");

if(fp==NULL)
{
printf("The file could not be open");
exit(0);
}
```

Q2. Write a C program to read name and marks of n number of students from user and store them in a file. If the file previously exists, add the information of n students.

Ans:

```
#include <stdio.h>

int main()
{
char name[50];
int marks, i,n;
```

```

printf("Enter number of students");

scanf("%d", &n);

FILE *fptr;
fptr=(fopen("C:\\student.txt","a"));

if (fptr==NULL){ printf("Error!");
exit(1);

}

for(i=0;i<n;++i)

{ printf("For student%d\nEnter name: ",i+1);
scanf("%s",name);

printf("Enter marks");

scanf("%d", &marks);

fprintf(fptr, "\nName: %s\nMarks=%d\n", name, marks);

} fclose(fptr);

Return 0;

}

```

The fclose function causes the stream pointed to be flushed and the associated file to be closed. Any unwritten buffered data for the stream are delivered to the host environment to be written to the file; any unread buffered data are discarded. The stream is disassociated from the file. If the associated buffer was automatically allocated, it is deallocated. The function returns zero if the stream was successfully closed or **EOF** if any errors were detected.

Q.3. Write a program to read data from file and close using fclose function.

Ans:

```

#include <stdio.h>

int main()

int n

```

```

FILE *fptr;

if ((fptr=fopen("C:\\program.txt","r"))==NULL){

printf("Error! opening file");

exit(1);    // Program exits if file pointer returns NULL.

}

fscanf(fptr,"%d",&n);

printf("Value of n=%d",n);

fclose(fptr);

return 0;

}

```

Q4. Write a C program to write all the members of an array of structures to a file using fwrite(). Read the array from the file and display on the screen.

Ans:

```

#include<stdio.h>

Struct s
{
Char name[50];
Int height;
};

Int main()
{
Struct s a[5], b[5];
FILE *fptr;

Int I;
Fptr=fopen("file.txt", "wb");

For(i=0; i<5; ++i)
{

```

```
fflush(stdin);

printf("Enter name: ") ;
gets(a[i].name);
printf("Enter height: ");
scanf("%d",&a[i].height);

}

fwrite(a,sizeof(a),1,fptr);
fclose(fptr);
fptr=fopen("file.txt","rb");
fread(b,sizeof(b),1,fptr);
for(i=0;i<5;++i)

{
printf("Name: %s\nHeight: %d",b[i].name,b[i].height);
}  fclose(fptr);
}
```