File Handling in C

Many real life problems involve large volumes of data and in such situations the console oriented I/O operations pose two major problems,

1. It becomes tedious and time consuming to handle large volumes of data through terminals.
2. The entire data is lost when either the program is terminated or the computer is turned off.

It is thus necessary to have a more flexible approach where data can be stored on the disk and read whenever necessary. This method employs the concept of files to store the data.

A file is a place on the disk where a group of related data is stored.The basic file operations are

1. Naming a file
2. Opening a file
3. Reading data from a file
4. Writing data to file and closing a file

File handling functions

Function name	Description
fopen()	Creates new file for use or opens existing file for use
fclose()	Closes a file which has been opened for use
getc()	Reads a character from a file
putc()	Writes a character to file
fprintf()	Writes set of data values from a file
fscanf()	Reads a set of data values from a file
getw()	Reads an integer from a file
putw()	Writes an integer to a file
fseek()	Sets the position to a desired point in the file
ftell()	Gives the current position in the file(in terms of bytes from the start)
rewind()	Sets the position to the beginning of the file

File opening modes

1.”r” :-reading from a file.

2.”w”:-searches a file,if the file exists the contents are overwritten,if it does not exists a new file is created.

3.”a”:-appending new contents at end of file

4.”r+”:-reading existing contents,writing new contents and modifying existing contents of file.

5.”w+”:-writing new contents ,reading them back and modifying existing contents

6.”a+”:-reading existing contents,writing new contents however cant modify contents.

Following is the general format for declaring and opening a file:

```
FILE *fp;
fp=fopen(“filename”,”mode”);
```

The first statement declares the variable fp as a “pointer to data type FILE”.FILE is a structure that is defined in the I/O library.The second statement open the file named filename and assigns the identifier to the FILE type pointer fp.This pointer(fp) which contains all the information about the file is then used as a communication link between the system and the program.It also specifies the purpose of opening the file by specifying the mode of opening.

Eg:-

```
…..
FILE *p1,*p2;
p1=fopen(“data”,”r”);
p2=fopen(“showresult”,”w”);
…..
…..
fclose(p1);
fclose(p2);
……
```

This program opens two files and closes them after all operations on them are completed.

Examples:

1.Write a ‘C’ Program to read data from a text file and display the data on the screen. If file does not exist create a file.

```
#include<stdio.h>
int main()
{
  FILE *fp;
```

```
  char ch;
  clrscr();
  fp=fopen("sample1.txt","r");//use "w" mode if the file is to be created
  if(fp==NULL)
  {
    printf("File doesn't exist");
  }
  else
  {  while((ch=getc(fp))!=EOF)
     { printf("%c",ch);
     }

  }
  fclose(fp);
  getch();
  return 0;
}
```