File Handling In C

A **file** is a place on the disk where group of related data are stored.The data file always allows us to store information permanently and to access and alter the information whenever necessary.Or **file** can also be said as a system of bytes or characters stored contiguously in memory.

Types of file:

a)Text File

b)Binary File

Difference between text and binary file

Text File	Binary File
1.It consists of text in contrast to ASCII form.	1.It consists of information in binary form i.e. 0's and 1's.
2.It is readable and hence no need of conversion.	2.It is non-readable hence conversion is required.
3.The end of the text file is indicated by EOF(End Of File) whose ASCII code is 26.	3.The end of the binary file has no character indicated.
4.File opening mode in text file are :r,w,a,r+,w+,a+.	4.File opening modes in binary file are:rb,wb,ab,rb+,ab+,wb+.

5.The extension of text file is .txt.	5.The extension of binary file is .dat.

File operations or File Handling in C

1.Opening or creating a file.

2.Writing data to file.

3.Reading from file.

4.Closing a file.

Opening or creating a file: Before writing any content to file or reading from file we must create or open a file.

- Opening a file establishes a link between program and the operating system about which file we are going to access and how.
- This provides operating system name of file and the mode in which file is to be opened.
- The link between our program and the operating system is a structure called FILE which has been defined under a header file stdio.h.
- When we request operating system to open a file,then we get back a pointer to structure FILE.

- Each file has its own FILE structure that contains information about the file being used as its current size,its location in memory,e.t.c.
- More important it contains a character pointer which points to the character that we are going to read.

Syntax for opening or creating a file:

FILE *file_pointer_variable;

file_pointer_variable=fopen(filename,file_opening_mode);

e.g:

FILE *fp;

Fp=fopen(“myfile.txt”,”r”);

Here,both the parameters of fopen() should be included inside “ “ as these parameters are treated as string.

- After opening a file ,then only we can process as per requirement.
- fopen() returns NULL depending upon the various file opening modes.
- While opening any file following important tasks are performed:

  a)Firstly,it searches in the disk for the file that is to be opened.

  b)If file is available in the disk then it loads the file from the disk into a place in memory called **Buffer.**But if the file is not available in the disk then it returns the NULL message.

  c)Finally, it sets up a character pointer that points to first character of buffer.

Closing a file: All file that is opened must be closed as soon as the operations in file to be performed are completed.The closing of the file ensures that all information associated with file have been flushed from buffer and all the link to file is broken.

Syntax for closing a file:

fclose(file_pointer_variable);

e.g:

fclose(fptr);

While closing any file using fclose() two operations are performed:

- The character in buffer would be written to file on the disk.
- The buffer is freed from the memory so that all the link to the file is broken.

File opening modes in C

Different file opening modes in C are listed below:

1. **“r”(read only mode):**
    - This mode opens file in read mode only.
    - If the file exists, it loads into the memory and sets up a pointer which points to the first character in it but if the file doesn’t exists, then it returns NULL.

      e.g:

      FILE *fp;

      fp=fopen(“abc.txt”,”r”);

      NOTE: If binary file is to be opend in read only mode the “rb” is used.

e.g:

```
FILE *fp;
fp=fopen("abc.dat","rb");
```

2. **"w"(write only mode):**
   - This mode opens a text file for writing only.
   - It searches specified file.If the file already exists,the contents are overwritten(i.e. content are deleted first and then written).If the file does not exists,a new file is created.It returns NULL, if it is unable to open the file in write mode.

     e.g:

     ```
     FILE *fp;
     fp=fopen("abc.txt","w");
     ```

     NOTE: If binary file is to be opend in write only mode the "rw" is used.

     e.g:

     ```
     FILE *fp;
     fp=fopen("abc.dat","wb");
     ```

3. **"a"(append only mode):**
   - It opens an existing file for appending(i.e. adding new information at the end of the file).
   - It searches specified file.If the specified file exists,it loads into memory and set up a pointer that points to the last character in it.If the file doesnot exists, a new file is created.It returns NULL if file cannot be opend.

     e.g:

```
FILE *fp;
fp=fopen("abc.txt","a");
```

NOTE: If binary file is to be opened in append only mode the "ab" is used.

e.g:

```
FILE *fp;
fp=fopen("abc.dat","ab");
```

4. **"r+"(read and write mode):**
   - It opens existing text file for reading and writing.
   - It searches the specified file.If the file exists,loads it into the memory and set up a pointer which points to the first character in it.It returns NULL if the file doesnot exists.

     e.g:

     ```
     FILE *fp;
     fp=fopen("abc.txt","r+");
     ```

     NOTE: If binary file is to be opened in read and write mode then "rb+" is used.

     e.g:

     ```
     FILE *fp;
     fp=fopen("abc.dat","rb+");
     ```

5. **"w+"(write and read mode):**
   - It opens text file for reading and writing.

- If the specified file exists,it's content are destroyed.If the file doesn't exists,then new file is created.It returns NULL if it is unable to open the specified file.

  e.g:

  FILE *fp;

  fp=fopen("abc.txt","w+");

  NOTE: If binary file is to be opened in write and read mode then "wb+" is used.

  For e.g:

  FILE *fp;

  fp=fopen("abc.dat","wb+");

6. "a+"(append and read mode):

- It opens an existing text file for both reading and appending.
- A new file will be created if the file doesnot exists.It returns NULL if it became unable to open the secified file.

  For e.g:

  FILE *fp;

  fp=fopen("abc.txt","a+");

  NOTE: If binary file is to be opend in append and read mode then "ab+" is used.

  For e.g:

  FILE *fp;

  fp=fopen("abc.dat","ab+");

****Character input/output functions in file****

a)fgetc():

It is used to read a character from a file.

Syntax:

char_variable=fgetc(file_ptr_variable);

b)fputc():

It is used to write a character to a file.

Syntax:

fputc(character or character variable,file_ptr_variable);

Qno1)WAP to read characters from keyboard and write to a file named "myfile.txt".

Ans:

```
#include<stdio.h>
#include<conio.h>
#include<stdlib.h>
int main(){
    char ch;
    FILE *fp;
    fp=fopen("myfile.txt","w");
    if(fp==NULL){
        printf("file couldn't be opened");
        getch();
        exit(0);
    }
    printf("enter some character");
    while((ch=getchar())!='\n'){
```

```
            fputc(ch,fp);
        }
        fclose(fp);
        getch();
        return 0;
    }
```

Qno2) WAP to read characters from file named “myfile.txt” and display it to monitor.

Ans:

```
#include<stdio.h>
#include<conio.h>
#include<stdlib.h>
int main(){
    char ch;
    FILE *fp;
    fp=fopen("myfile.txt","r");
    if(fp==NULL){
        printf("file couldn't be opened");
        getch();
        exit(0);
    }
    printf("The characters from file are:\n");
    while((ch=fgetc(fp))!=EOF){
```

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

Qno3)WAP to read characters from keyboard,store it in a file named “myfile.dat” and display it.

Ans:

```
#include<stdio.h>
#include<conio.h>
#include<stdlib.h>
int main(){
    char ch;
    FILE *fp;
    fp=fopen("myfile.dat","wb+");
    if(fp==NULL){
        printf("file couldn't be opened");
        getch();
        exit(0);
    }
    printf("Enter some characters");
```

```
        while((ch=getchar())!='\n'){
                fputc(ch,fp);
        }
        rewind(fp);
        printf("characters from the file are:\n");
        while((ch=fgetc(fp))!=EOF){
                putch(ch);
        }
        fclose(fp);
        getch();
        return 0;
}
```

****String input/output functions in file****

a)fgets():

It is used to read string from file.

Syntax:

fgets(string,n,file_ptr_variable);

where,n=string length.

b)fputs():

It is used to write a string to a file.

Syntax:

fputs(string,file_ptr_variable);

Qno1) Create a file named “test.txt” and write text “Welcome to my college” to this file.

Ans:

```
#include<stdio.h>
#include<conio.h>
#include<stdlib.h>
int main(){
	FILE *fp;
	fp=fopen("test.txt","w");
	if(fp==NULL){
		printf("file couldn't be opened");
		getch();
		exit(0);
	}
	fputs("Welcome to my college",fp);
	fclose(fp);
	getch();
	return 0;
}
```

Qno2)WAP to open an existing file “test.txt” ,read its content and display it to the screen.

Ans:

```
#include<stdio.h>
#include<conio.h>
#include<stdlib.h>
int main(){
	char s[100];
	FILE *fp;
	fp=fopen("test.txt","r");
	if(fp==NULL){
		printf("file couldn't be opened");
		getch();
		exit(0);
	}
	fgets(s,100,fp);
	printf("the text from file is %s",s);
	fclose(fp);
	getch();
	return 0;
}
```

Qno3)WAP to write text to a file named “test.txt” until the user hits enter key,read the content and display it to the screen.

Ans:

```
#include<stdio.h>
#include<conio.h>
#include<stdlib.h>
#include<string.h>
int main(){
	char s[200];
	FILE *fp;
	fp=fopen("test.txt","w+");
	if(fp==NULL){
		printf("file couldn't be opened");
		getch();
		exit(0);
	}
	while(strlen(gets(s))!=0){
		fputs(s,fp);
	}
	rewind(fp);
	printf("text from the file are:");
	while((fgets(s,100,fp))!=NULL){
		puts(s);
	}
	fclose(fp);
```

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

****Formatted input/output functions in file****

a)fprintf():

This function is formatted output function which is used to write some integer,float,char or string to a file.

Syntax:

fprintf(file_ptr_variable,"format specifiers",list_variables);

b)fscanf():

This function is formatted input function which is used to read some integer,float,char or string from a file.

Syntax:

fscanf(file_ptr_variable,"format specifier",&list_variables);

Qno1)WAP to create a file "student.txt" in D drive and write age,gender and marks of a student to the file and display all the information.

Ans:

```
#include<stdio.h>
#include<conio.h>
#include<stdlib.h>
#include<string.h>
int main(){
```

```
    int age;
    char gender[50];
    float marks;
    FILE *fp;
    fp=fopen("D:\\student.txt","w+");
    if(fp==NULL){
        printf("file couldn't be opened");
        getch();
        exit(0);
    }
    printf("enter age,gender and marks ");
    scanf("%d%s%f",&age,&gender,&marks);
    fprintf(fp,"%d,%c,%f",age,gender,marks);
    rewind(fp);
    fscanf(fp,"%d%c%f",&age,&gender,&marks);
    printf("age=%d,gender=%s and marks=%f",age,gender,marks);
    fclose(fp);
    getch();
    return 0;
}
```

****Record input/output functions in file****

a)fwrite():

Syntax:

fwrite(&ptr,size_of_array_or_structure,number_of_structure_or_array, fptr);

b)fread():

Syntax:

fread(&ptr,size_of_array_or_structure,number_of_structure_or_array,fptr);

Where,

- Ptr is the address of an array or structure to be written.
- size_of_array_or_structure is an integer value that shows the size of structure or array which is being read or written.
- number_of_structure_or_array is an integer value that indicates number of arrays or structures to be written to file or read from file.
- fptr is a file pointer of a file opened.

Qno1)WAP to input name,address,faculty,program and GPA(in maximum 4.0) of 500 students and store in "RESULT.DAT" data file and display the records of those students whose faculty is "Engineering" and GPA>3.5**.[PU. 2016 Spring]**

Ans:

```
#include<stdio.h>

#include<conio.h>
```

```
#include<stdlib.h>
#include<string.h>
struct student{
	char name[100];
	char address[100];
	char faculty[100];
	char program[100];
	float gpa;
};
int main(){
	struct student s[500];
	int i;
	FILE *fp;
	fp=fopen("RESULT.DAT","wb+");
	if(fp==NULL){
		printf("file cannot be opened");
		getch();
		exit(0);
	}
	for(i=0;i<500;i++){
		printf("Enter name");
		fflush(stdin);
```

```
            gets(s[i].name);
            printf("Enter address");
            gets(s[i].address);
            printf("Enter faculty");
            gets(s[i].faculty);
            printf("Enter program");
            gets(s[i].program);
            printf("Enter gpa");
            scanf("%f",&s[i].gpa);
            fwrite(&s,sizeof(s),1,fp);
        }
        rewind(fp);
        for(i=0;i<500;i++){
            fread(&s,sizeof(s),1,fp);
            if((strcmp(s[i].faculty,"Engineering")==0) && (s[i].gpa>3.5))
            {
              printf("name=%s,address=%s,faculty=%s,program=%s and
gpa=%f",s[i].name,s[i].address,s[i].faculty,s[i].program,s[i].gpa);
        }
        }
        fclose(fp);
        getch();
```

```
    return 0;


}
```

Qno2)WAP to input name,address,registration number,faculty and academic year of admission in university of ‘n’ number of students of PU and append them in a data file called “STUDENT.DAT” .Then display records of those students by reading the records from “STUDENT.DAT” data file who got admission in 2016**.[PU. 2015 Spring]**

Ans:

```
#include<stdio.h>
#include<conio.h>
#include<stdlib.h>
#include<string.h>
struct student{
    char name[100];
    char address[100];
    long int reno;
    char faculty[100];
    float year;
};
int main(){
    struct student s[100];
    int i,n;
```

```
FILE *fp;
fp=fopen("STUDENT.DAT","ab+");
if(fp==NULL){
        printf("file cannot be opened");
        getch();
        exit(0);
}
printf("enter total number of students");
scanf("%d",&n);
for(i=0;i<n;i++){
        printf("Enter name");
        fflush(stdin);
        gets(s[i].name);
        printf("Enter address");
        gets(s[i].address);
        printf("Enter registration number");
        scanf("%ld",&s[i].reno);
        printf("Enter faculty");
        fflush(stdin);
        gets(s[i].faculty);
        printf("Enter academic year");
        scanf("%d",&s[i].year);
```

```
        fwrite(&s,sizeof(s),1,fp);
    }
    rewind(fp);
    for(i=0;i<n;i++){
        fread(&s,sizeof(s),1,fp);
        if(s[i].year==2016)
        {
            printf("name=%s,address=%s,faculty=%s,registration
number=%ld and academic
year=%d",s[i].name,s[i].address,s[i].faculty,s[i].reno,s[i].year);
    }
    }
    fclose(fp);
    getch();
    return 0;

}
```

Qno3)WAP to read the name,author and price of 500 books in a library from “library.dat”.Now print the book name and price of those books whose price is above Rs.300**.[PU. 2015 Fall]**

Ans:

```
#include<stdio.h>
#include<conio.h>
```

```
#include<stdlib.h>
#include<string.h>
struct student{
	char name[100];
	char author[100];
	float price;
};
int main(){
	struct student s[500];
	int i;
	FILE *fp;
	fp=fopen("library.dat","wb+");
	if(fp==NULL){
		printf("file cannot be opened");
		getch();
		exit(0);
	}
	for(i=0;i<500;i++){
		printf("Enter name");
		fflush(stdin);
		gets(s[i].name);
		printf("Enter author");
```

```
        gets(s[i].author);
        printf("Enter price of the book");
        scanf("%f",&s[i].price);
        fwrite(&s,sizeof(s),1,fp);
    }
    rewind(fp);
    for(i=0;i<500;i++){
        fread(&s,sizeof(s),1,fp);
        if(s[i].price>300)
        {

          printf(" Book name=%s and price=%f",s[i].name,s[i].price);
    }
    }
    fclose(fp);
    getch();
    return 0;

}
```

Qno4)WAP to open a new file,read name,address and telephone number of 10 employees from the user and write it to a file.

Ans:

```
#include<stdio.h>
#include<conio.h>
#include<stdlib.h>
struct employee{
	char name[100];
	char address[100];
	long int telno;
};
int main(){
	struct employee e[10];
	int i;
	FILE *fp;
	fp=fopen("employee.dat","wb");
	if(fp==NULL){
		printf("File cannot be opened");
		getch();
		exit(0);
	}
	for(i=0;i<10;i++){
		printf("enter employee name");
		gets(e[i].name);
		printf("enter address");
```

```
            gets(e[i].address);
            printf("enter telephone number");
            scanf("%ld",&e[i].telno);
            fwrite(&e,sizeof(e),1,fp);
        }
        fclose(fp);
        getch();
}
```

Qno5)WAP to create structure for the following data for cricket game(country name,player name,playing type(e.g.bating,balling or both),number of matches played by player and salary).Save ,the information in a file and display the information of those players who had played more than 10 matches.**[PU. 2018 Fall].**

Ans:

```
#include<stdio.h>
#include<conio.h>
#include<stdlib.h>
#include<string.h>
struct cricket{
        char cname[100];
        char pname[100];
        char ptype[100];
```

```
        int matches;
        long int salary;
};
int main(){
        struct cricket c[100];
        int n,i;
        FILE *fp;
        fp=fopen("cricket.dat","wb+");
        if(fp==NULL){
                printf("File cannot be opened");
                getch();
                exit(0);
        }
        printf("enter total number of players");
        scanf("%d",&n);
        for(i=0;i<n;i++){
                printf("enter country name");
                fflush(stdin);
                gets(c[i].cname);
                printf("enter player name");
                gets(c[i].pname);
                printf("enter playing type");
```

```
            gets(c[i].ptype);
            printf("enter number of matches played");
            scanf("%d",&c[i].matches);
            printf("enter salary");
            scanf("%ld",&c[i].salary);
            fwrite(&c,sizeof(c),1,fp);
        }
        rewind(fp);
        for(i=0;i<n;i++){
            fread(&c,sizeof(c),1,fp);
            if(c[i].matches>10){
                printf("country name=%s,player name=%s,playing
type=%s,number of matches=%d and total salary
=%ld",c[i].cname,c[i].pname,c[i].ptype,c[i].matches,c[i].salary);
            }
        }
        fclose(fp);
        getch();
}
```

Qno6) Create a structure called goods that stores number,price,purchase date and quantity.WAP to store the information of 100 goods into the file named “goods.dat”.

Ans:

```
#include<stdio.h>
#include<conio.h>
#include<stdlib.h>
#include<string.h>
struct date{
        int dd;
        int mm;
        int yy;
};
struct goods{
        int num;
        int price;
        int quantity;
        struct date d;
};
int main(){
        struct goods g[100];
        int i;
        FILE *fp;
```

```
fp=fopen("goods.dat","wb");
if(fp==NULL){
        printf("File cannot be opened");
        getch();
        exit(0);
}
for(i=0;i<100;i++){
        printf("enter total number");
        scanf("%d",&g[i].num);
        printf("enter price");
        scanf("%d",&g[i].price);
        printf("enter quantity");
        scanf("%d",&g[i].quantity);
        printf("enter year ");
        scanf("%d",&g[i].d.yy);
        printf("enter month");
        scanf("%ld",&g[i].d.mm);
        printf("enter day");
        scanf("%ld",&g[i].d.dd);
        fwrite(&g,sizeof(g),1,fp);
}
fclose(fp);
```

```
    getch();
}
```

Qno7) Create a file called “university.dat” .Input ‘n’ records of college in a structure having college name,location and number of faculties of pokhara university.Now,display name of colleges whose location is “kathmandu”.

Ans:

```
#include<stdio.h>
#include<conio.h>
#include<stdlib.h>
#include<string.h>
struct university{
    char cname[100];
    char location[100];
    int nof[100];
};
int main(){
    struct university u[100];
    int n,i;
    FILE *fp;
    fp=fopen("university.dat","wb+");
    if(fp==NULL){
        printf("File cannot be opened");
```

```
        getch();
        exit(0);
    }
    printf("enter total number of universities");
    scanf("%d",&n);
    for(i=0;i<n;i++){
        printf("enter college name");
        fflush(stdin);
        gets(u[i].cname);
        printf("enter location");
        gets(u[i].location);
        printf("enter number of faculties");
        scanf("%d",&u[i].nof);
        fwrite(&u,sizeof(u),1,fp);
    }
    rewind(fp);
    for(i=0;i<n;i++){
        fread(&u,sizeof(u),1,fp);
        if(strcmp(u[i].location,"kathmandu")==0){
            printf("college name=%s,location=%s and number of
faculties =%d",u[i].cname,u[i].location,u[i].nof);
        }
```

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

Qno8) Create the following structure:

Id	Name	Address	Salary	Date of birth		
				yy	dd	mm

Input 100 employee information and store it in file “employee.dat” and display records of those employee whose address is “pokhara”.

Ans:

```
#include<stdio.h>
#include<conio.h>
#include<stdlib.h>
struct date{
    int dd;
    int yy;
    int mm;
};
struct employee{
    int id;
```

```
        char name[100];
        char address[100];
        long int salary;
        struct date d;
};
int main(){
        struct employee e[100];
        int i;
        FILE *fp;
        fp=fopen("employee.dat","wb+");
        if(fp==NULL){
                printf("File cannot be opened");
                getch();
                exit(0);
        }
        for(i=0;i<100;i++){
                printf("enter employee id");
                scanf("%d",&e[i].id);
                printf("enter employee name");
                fflush(stdin);
                gets(e[i].name);
                printf("enter address");
```

```
            gets(e[i].address);
            printf("enter salary");
            scanf("%ld",&e[i].salary);
            printf("enter day");
            scanf("%d",&e[i].d.dd);
            printf("enter month");
            scanf("%d",&e[i].d.mm);
            printf("enter year");
            scanf("%d",&e[i].d.yy);
            fwrite(&e[i],sizeof(e),1,fp);
            }
            for(i=0;i<100;i++){
                  fread(&e,sizeof(e),1,fp);
                  if(strcmp(e[i].address,"pokhara")==0){
printf("id=%d,name=%s,address=%s,salary=%ld,day=%d,year=%d,mont
h=%d",e[i].id,e[i].name,e[i].salary,e[i].d.dd,e[i].d.yy,e[i].d.mm);
}
}
      fclose(fp);
      getch();
}
```

Qno9) Create a structure called student having name,age and rollno.Input ‘n’ records,store them in file called “student.txt” and display information.

Ans:

```
#include<stdio.h>
#include<conio.h>
#include<stdlib.h>
struct student{
	char name[100];
	int age;
	int roll;
};
int main(){
	struct student s[100];
	int i,n;
	FILE *fp;
	fp=fopen("student.txt","w+");
	if(fp==NULL){
		printf("File cannot be opened");
		getch();
		exit(0);
	}
	printf("enter total number of students");
```

```
	scanf("%d",&n);
	for(i=0;i<n;i++){
		printf("Enter name");
		fflush(stdin);
		gets(s[i].name);
		printf("enter age");
		scanf("%d",&s[i].age);
		printf("enter roll no.");
		scanf("%d",&s[i].roll);
		fwrite(&s,sizeof(s),n,fp);
	}
	rewind(fp);
	for(i=0;i<n;i++){
		fread(&s,sizeof(s),1,fp);
		printf("name=%s,age=%d and roll
no=%d",s[i].name,s[i].age,s[i].roll);
	}
	fclose(fp);
	getch();
}
```