Date: 09/04/2022

Experiment no. 8

Q. To study and demonstrate structures and pointers.

PROGRAM STATEMENTS:

a) Write a C program (using Structures) to create a file named STUD\_FI which will store information of 20 students. (A class of B.E. first year has 20 student. Each student records comprised of roll number, Name, Fee Paid)

b) Write a C program to read the STUD\_FI file and print contents of it.

THEORY: File Handling: File Handling in C enables us to create, update, read, and delete the files stored on the local life system through our C program. The following operations can be performed on a file:

* Creation of the new file
* Opening an existing file
* Reading from the file
* Writing to the file
* Deleting the file
* **Functions for file handling**
* There are many functions in the C library to open, read, write, search and close the file. A list of file functions are given below:

|  |  |  |
| --- | --- | --- |
| **No.** | **Function** | **Description** |
| 1 | fopen() | opens new or existing file |
| 2 | fprintf() | write data into the file |
| 3 | fscanf() | reads data from the file |
| 4 | fputc() | writes a character into the file |
| 5 | fgetc() | reads a character from file |
| 6 | fclose() | closes the file |
| 7 | fseek() | sets the file pointer to given position |
| 8 | fputw() | writes an integer to file |
| 9 | fgetw() | reads an integer from file |
| 10 | ftell() | returns current position |
| 11 | rewind() | sets the file pointer to the beginning of the file |

## Opening File: fopen()

We must open a file before it can be read, write, or update. The fopen() function is used to open a file. The syntax of the fopen() is given below.

The fopen() function accepts two parameters:

* The file name (string). If the file is stored at some specific location, then we must mention the path at which the file is stored. For example, a file name can be like **"c://some\_folder/some\_file.ext"**.
* The mode in which the file is to be opened. It is a string.

We can use one of the following modes in the fopen() function.

|  |  |
| --- | --- |
| **Mode** | **Description** |
| r | opens a text file in read mode |
| w | opens a text file in write mode |
| a | opens a text file in append mode |
| r+ | opens a text file in read and write mode |
| w+ | opens a text file in read and write mode |
| a+ | opens a text file in read and write mode |
| rb | opens a binary file in read mode |
| wb | opens a binary file in write mode |
| ab | opens a binary file in append mode |
| rb+ | opens a binary file in read and write mode |
| wb+ | opens a binary file in read and write mode |
| ab+ | opens a binary file in read and write mode |

The fopen function works in the following way.

* Firstly, It searches the file to be opened.
* Then, it loads the file from the disk and place it into the buffer. The buffer is used to provide efficiency for the read operations.
* It sets up a character pointer which points to the first character of the file.

**Program a)**

**#include<stdio.h>**

**#include<conio.h>**

**struct student**

**{**

**char name[10];**

**int roll;**

**int fees;**

**};**

**int main()**

**{**

**struct student e[20];**

**FILE \*F1;**

**F1 = fopen("STUD\_FI","w");**

**int i;**

**for(i=0;i<20;i++)**

**{**

**printf("\nEnter the information of %d student:",i+1);**

**printf("\nEnter Name: ");**

**scanf("%s",e[i].name);**

**printf("Enter Roll No.: ");**

**scanf("%d",&e[i].roll);**

**printf("Enter fees: ");**

**scanf("%d",&e[i].fees);**

**}**

**for(i=0;i<20;i++)**

**{**

**fprintf(F1,"\n Name = %s",e[i].name);**

**fprintf(F1,"\n Roll No. = %d",e[i].roll);**

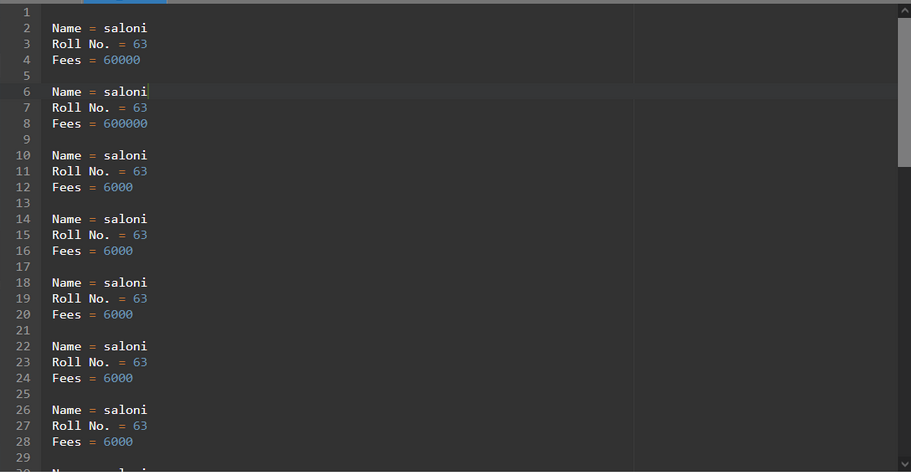
**fprintf(F1,"\n Fees = %d \n",e[i].fees);**

**}**

**return 0;**

**}**

OUTPUT:



**Program b)**

**#include<stdio.h>**

**#include<conio.h>**

**struct student**

**{**

**char name[10];**

**int roll;**

**int fees;**

**};**

**int main()**

**{**

**struct student e[20];**

**FILE \*F1;**

**F1 = fopen("STUD\_FI","r");**

**int i;**

**for(i=0;i<20;i++)**

**{**

**printf("\nEnter the information of %d student:\n",i+1);**

**fscanf(F1,"%s",e[i].name);**

**printf("\nEnter Name: %s",e[i].name);**

**fscanf(F1,"%d",&e[i].roll);**

**printf("\nEnter Roll No.: %d",e[i].roll);**

**fscanf(F1,"%d",&e[i].fees);**

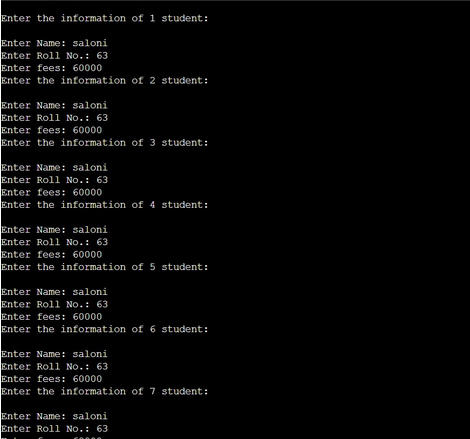
**printf("\nEnter fees: %d",e[i].fees);**

**}**

**return 0;**

**}**

OUTPUT:



CONCLUSION: In above experiment we learnt about file handling. We also learn about how to input data from terminal to the file and vice versa.

Moreover about certain functions like 'fprintf' and 'fscanf' used to read and write data in the file.