SRM INSTITUTE OF SCIENCE AND TECHNOLOGY, KATTANKULATHUR

PPS

MINI PROJECT IN C USING FILE MANAGEMENT

STUDENT DATA STORAGE MANAGEMENT SYSTEM

DESCRIPTION:

This is a basic program using file handling to store a student's data in a text file and manipulate it according to the choices provided to the user while running the program.

The data of the student contains basic information like first name, last name, roll number, address, course, etc.

The program allows the user to perform following tasks :

- Add record
- List out the Records
- Search for the records
- Delete a record
- Exit the portal

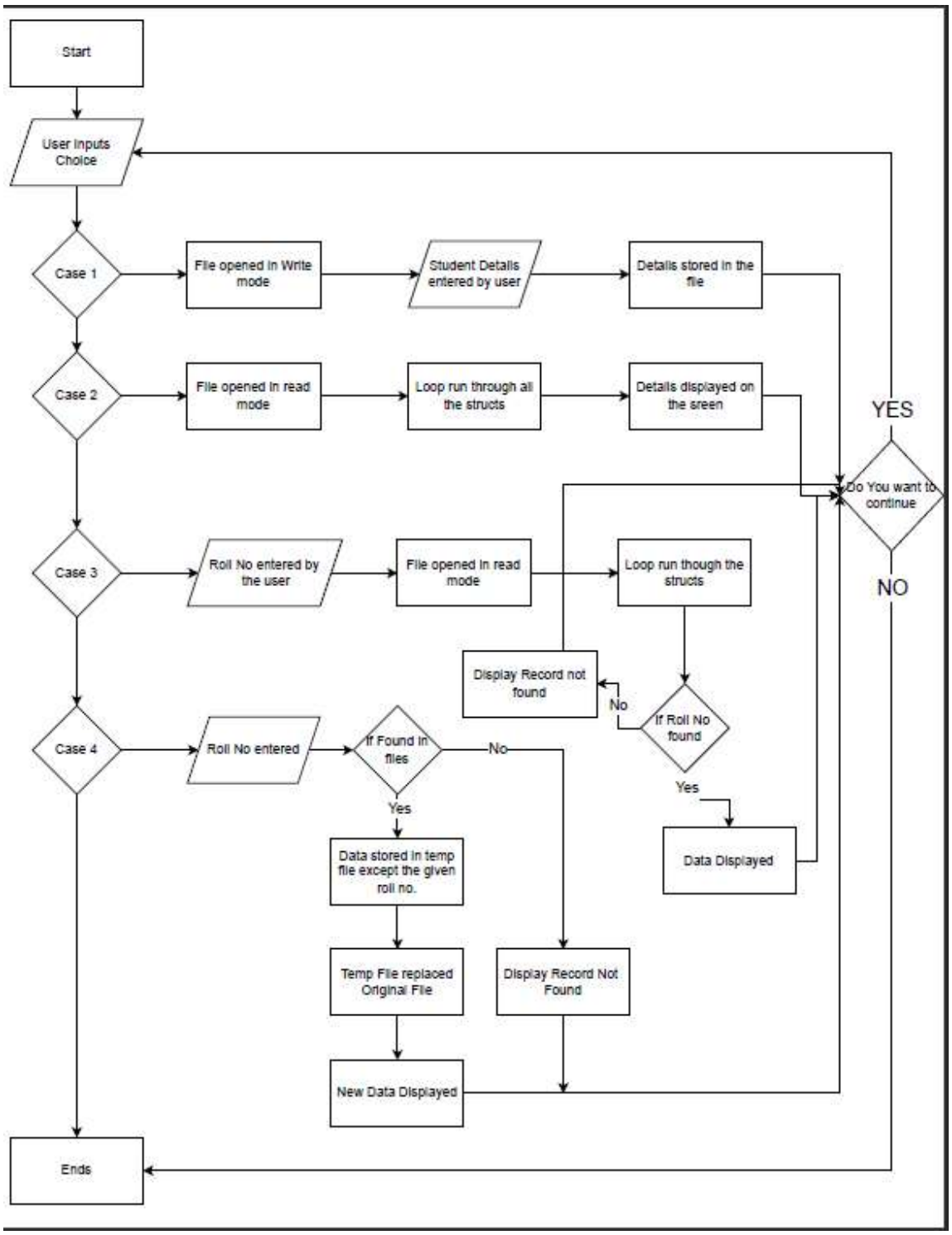
The following program creates a file to store the data and further manipulation is done with help of the information present in the file.

The source code is provided below.

Algorithm:

1. The user is presented with a list of options to choose from.
2. Based on the user input the respective switch-case block is executed.
3. If the user selects Add Record the user will be presented with a screen to input the details of the student.
4. After input the file is opened in write mode and the student details stored in struct are written into the file.
5. The program asks if the user wants to input any more records, if yes, process goes over again and if not, the screen falls back to the original one.
6. Here the user can choose to see all the records.
7. In this case, the file is opened in read only mode.
8. The mapping of the struct is done through the size of the stored structs in the file and a while loop is run until all details are printed.
9. File is then closed after printing.
10. The user goes back to the original screen.
11. Now the user can choose to search a student based on his/her roll no.
12. The File is again opened in read mode.
13. A while loop is run which iterates from 0 to the size of the total structs stored in the file.
14. For each iteration the program checks if the roll no of that struct is the same as the user input or not.
15. If it is the same the details of the student are printed and search successful is displayed.
16. If not found 'record not found' is displayed.
17. Now the user can choose to delete a students records based on their roll no.
18. This time a new file temp.txt is formed to stored duplicate data.
19. The original file is opened in read only mode and temp is opened in write mode.
20. The loop iterates through the structs stored in the original file.
21. If the roll no to be deleted doesn't match the data is copied to the temp.txt
22. If the roll is found a flag is returned as 1.
23. After the iteration is over, the program checks if the flag is 1.
24. If the flag is one it deletes the old file and renames the temp.txt same as the original file which does not contain the required record.
25. If the flag is 0 the program returns roll no. not found.
26. The user is returned to the home screen

FLOWCHART:


Start
User Inputs Choice
Case 1
File opened in Write mode
Student Details entered by user
Details stored in the file
Case 2
File opened in read mode
Loop run through all the structs
Details displayed on the sreen
YES
Do You want to continue
NO
Case 3
Roll No entered by the user
File opened in read mode
Loop run though the structs
Display Record not found
No
If Roll No found
Yes
Case 4
Roll No entered
If Found in files
No
Yes
Data Displayed
Data stored in temp file except the given roll no.
Temp File replaced Original File
Display Record Not Found
New Data Displayed
Ends

SOURCE CODE:

```
#include "stdio.h"
#include "conio.h"

struct student
{
    char first_name[100];
    char last_name[100];
    int roll_no;
    char Class[100];
    char Address[100];
    float per;
};
// Main function
void main()
{

    int choice;
    while (choice != 5)
    {
        printf("\n\t\t\t       =-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-
=-=-=-=-=-");
        printf("\n\t\t\t       = WELCOME TO SRMIST STUDENT DATABASE
MANAGEMENT SYSTEM =");
        printf("\n\t\t\t       =-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-
=-=-=-=-=-");
        printf("\n\n\n\t\t\t\t     1. Add Student\n");
        printf("\t\t\t\t     2. Students Records\n");
        printf("\t\t\t\t     3. Search Student\n");
        printf("\t\t\t\t     4. Delete Student\n");
        printf("\t\t\t\t     5. Exit\n");
        printf("\t\t\t\t    -------------------------------------------\n");
        printf("\t\t\t\t         RA2111050010046 HARSH SRIVASTAVA\n");
        printf("\t\t\t\t    -------------------------------------------
\n\n\n\n");
        printf("\t\t\t\t     Enter the provided choice: \n\n\n\n");
        scanf("%d", &choice);

        switch (choice)
        {
        case 1:
            addstudent();
            break;
        case 2:
            studentrecord();
            printf("\t\t\t\tpress any key to exit..... \n");
            getch();
```

```
            break;
        case 3:
            searchstudent();
            printf("\n\t\t\t\tPress any key to exit.......\n");
            getch();
            break;
        case 4:
            delete ();
            printf("\n\t\t\t\tPress any key to exit.......\n");
            getch();
            break;
        case 5:
            printf("\n\t\t\t\tThank you, for used this software.\n\n");
            exit(0);
            break;
        default:
            printf("\n\t\t\t\tEnter a valid number\n\n");
            printf("\t\t\tPress any key to continue.......");
            getch();
            break;
        }
    }
    getch();
}
// Function to create a new student
void addstudent()
{
    char another;
    FILE *fp;
    int n, i;
    struct student info;
    do
    {
        printf("\t\t\t\t               =======Add Students
Info=======\n\n\n");
        fp = fopen("srmist_students_record.txt", "a");
        printf("\n\t\t\tEnter First Name     : ");
        scanf("%s", &info.first_name);
        printf("\n\t\t\tEnter Last Name     : ");
        scanf("%s", &info.last_name);
        printf("\n\t\t\tEnter Roll-No (Integer Only)      : ");
        scanf("%d", &info.roll_no);
        printf("\n\t\t\tEnter Class(course) : ");
        scanf("%s", &info.Class);
        printf("\n\t\t\tEnter Address       : ");
        scanf("%s", &info.Address);
        printf("\n\t\t\tEnter Percentage    : ");
        scanf("%f", &info.per);
```

```
        printf("\n\t\t\t\t_____\n");
        if (fp == NULL)
        {
            fprintf(stderr, "can't open file");
        }
        else
        {
            printf("\t\t\tRecord stored successfuly\n\n");
        }
        fwrite(&info, sizeof(struct student), 1, fp);
        fclose(fp);
        printf("\t\t\tYou want to add another record?(y/n) : ");
        scanf("%s", &another);
    } while (another == 'y' || another == 'Y');
}
// Function to list out the pre stored records
void studentrecord()
{

    FILE *fp;
    struct student info;
    fp = fopen("srmist_students_record.txt", "r");
    printf("\t\t\t\t=======SRMIST STUDENTS RECORD=======\n\n\n");
    if (fp == NULL)
    {
        fprintf(stderr, "can't open file\n");
        exit(0);
    }
    else
    {
        printf("\t\t\t\tRECORDS :\n");
        printf("\t\t\t\t__________\n\n");
    }
    while (fread(&info, sizeof(struct student), 1, fp))
    {
        printf("\n\t\t\t\t\t Student Name  : %s %s", info.first_name,
info.last_name);
        printf("\n\t\t\t\t\t Roll NO       : %d", info.roll_no);
        printf("\n\t\t\t\t\t Class         : %s", info.Class);
        printf("\n\t\t\t\t\t Address  : %s", info.Address);
        printf("\n\t\t\t\t\t Percentage    : %f%", info.per);
        printf("\n\t\t\t\t\t _____\n");
    }
    fclose(fp);
    getch();
}
// Function to search any student
void searchstudent()
```

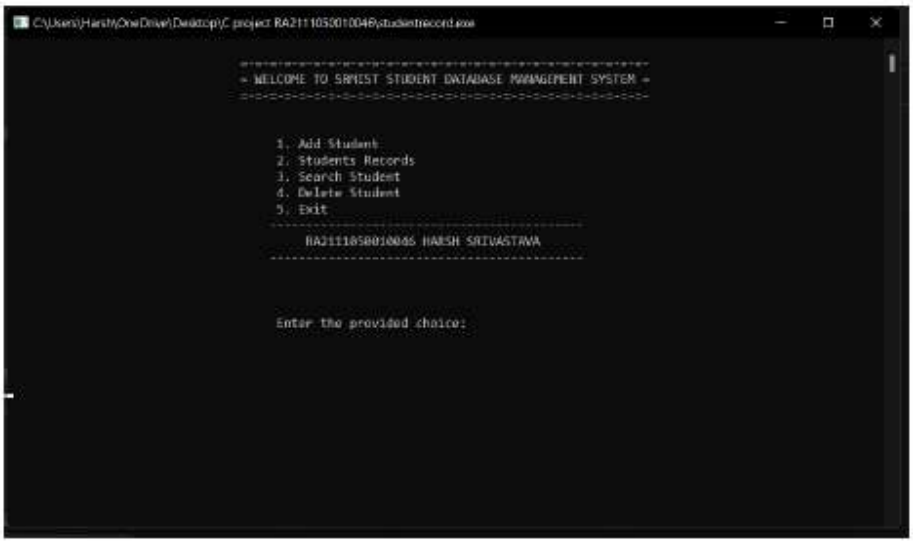
```
{
    struct student info;
    FILE *fp;
    int roll_no, found = 0;
    fp = fopen("srmist_students_record.txt", "r");
    printf("\t\t\t\t=======SEARCH SRMIST STUDENTS RECORD=======\n\n\n");
    printf("\t\t\tEnter the roll no : ");
    scanf("%d", &roll_no);
    while (fread(&info, sizeof(struct student), 1, fp) > 0)
    {
        if (info.roll_no == roll_no)
        {
            found = 1;
            printf("\n\n\t\t\tStudent Name : %s %s", info.first_name,
info.last_name);
            printf("\n\t\t\tRoll NO        : %d", info.roll_no);
            printf("\n\t\t\tClass          : %s", info.Class);
            printf("\n\t\t\tAddress        : %s", info.Address);
            printf("\n\t\t\tPercentage     : %f%", info.per);
            printf("\n\t\t\t_____________________________________\n");
        }
    }
    if (!found)
    {
        printf("\n\t\t\tRecord not found\n");
    }
    fclose(fp);
    getch();
}
// Function to delete a record/student
void delete ()
{
    struct student info;
    FILE *fp, *fp1;
    int roll_no, found = 0;
    printf("\t\t\t\t=======DELETE STUDENTS RECORD=======\n\n\n");
    fp = fopen("srmist_students_record.txt", "r");
    fp1 = fopen("temp.txt", "w");
    printf("\t\t\t\tEnter the roll no : ");
    scanf("%d", &roll_no);
    if (fp == NULL)
    {
        fprintf(stderr, "can't open file\n");
        exit(0);
    }

    while (fread(&info, sizeof(struct student), 1, fp))
    {
```

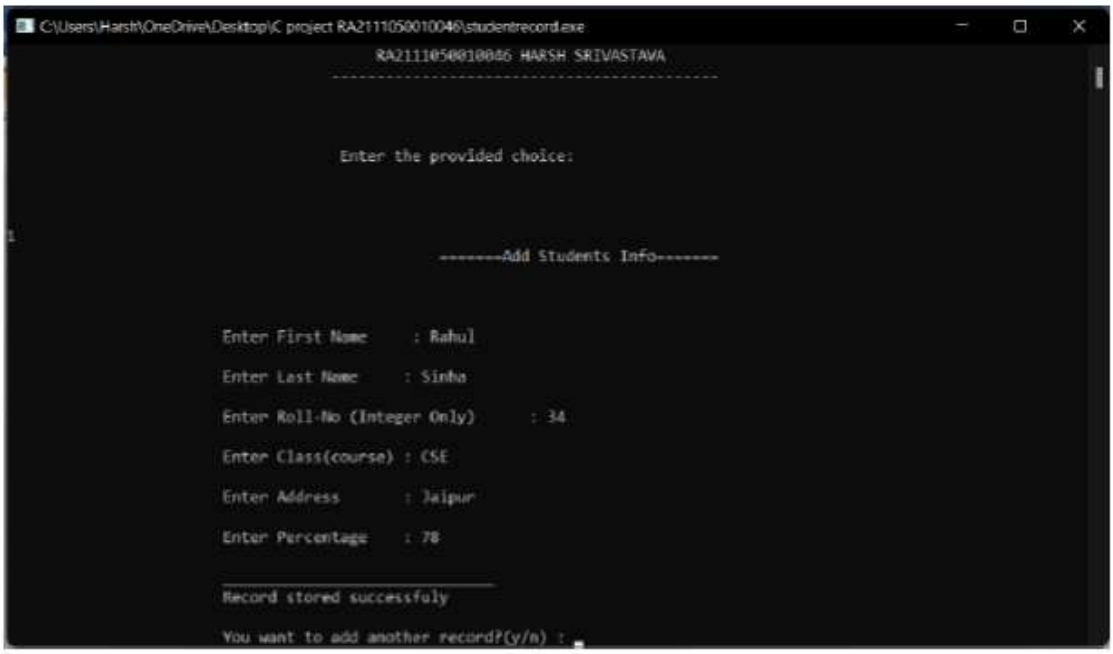
```
        if (info.roll_no == roll_no)
        {
            found = 1;
        }
        else
        {
            fwrite(&info, sizeof(struct student), 1, fp1);
        }
    }
    fclose(fp);
    fclose(fp1);
    if (!found)
    {
        printf("\n\t\t\t\tRecord not found\n");
    }
    if (found)
    {
        remove("srmist_students_record.txt");
        rename("temp.txt", "srmist_students_record.txt");
        printf("\n\t\t\t\tRecord deleted succesfully\n");
    }
    getch();
}
```

OUTPUT:

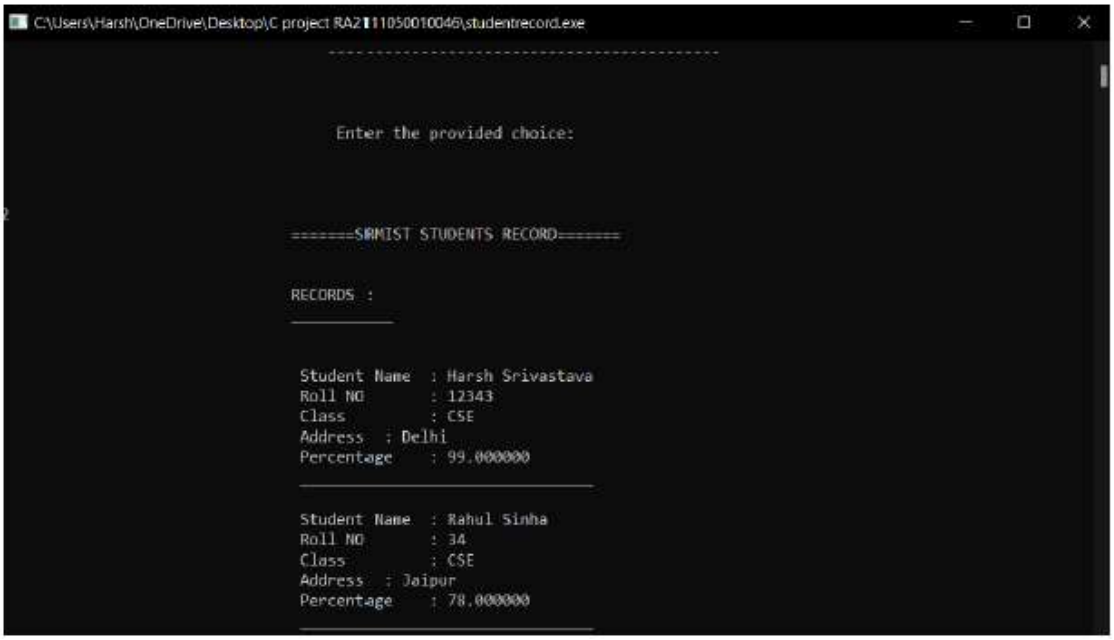
Interface of the Main menu of the program:



Interface of Adding a student record:



Interface of displaying the list of stored records:



Interface of deleting a record:

Interface of searching and displaying an existing record:

TEAM MEMBERS:-

- ARCHISMAN DAS
- MUHAMMAD RAIYAN
- HARSH SRIVASTAVA
- HARDIK GUPTA