# STUDENT RECORD MANAGEMENT SYSTEM

# C Programming Project

## Introduction

The Student Record Management System is a C programming project that stores, updates, and manages student information. It demonstrates file handling, structures, and menu-driven programming in C language. The system allows adding, displaying, searching, modifying, and deleting student records. This system replaces traditional paper records and provides efficient digital data storage.

## Objectives

**1.** To implement file handling in C language

**2.** To understand create, read, update, and delete operations

**3.** To design a menu-driven interface

**4.** To organize and store records permanently

**5.** To improve logical thinking and coding skills

# Software and Hardware Requirements

## Software Requirements:

• Windows/Linux OS

• CodeBlocks / Dev C++ / VS Code / GCC Compiler

## Hardware Requirements:

• **2**GB RAM

• Dual Core processor

• Keyboard and Monitor

# Flowchart

START

↓

Display Menu

↓

Select Operation

↓

Perform Action

↓

Return to Menu

↓

Exit

↓

END

# Source Code

```
#include <stdio.h>
#include <stdlib.h>
#include <string.h>

struct student {
    int roll;
    char name[50];
    int age;
    float marks;
};

FILE *fp;
struct student s;

void addRecord();
void displayRecords();
void searchRecord();
void deleteRecord();
void modifyRecord();

int main() {
    int ch;

    while(1) {
```

```
        printf("\n===== STUDENT RECORD MANAGEMENT =====\n");
        printf("1. Add Record\n");
        printf("2. Display All Records\n");
        printf("3. Search Record\n");
        printf("4. Modify Record\n");
        printf("5. Delete Record\n");
        printf("6. Exit\n");
        printf("Enter your choice: ");
        scanf("%d", &ch);

        switch(ch) {
            case 1: addRecord(); break;
            case 2: displayRecords(); break;
            case 3: searchRecord(); break;
            case 4: modifyRecord(); break;
            case 5: deleteRecord(); break;
            case 6: exit(0);
            default: printf("Invalid Choice!\n");
        }
    }
    return 0;
}

void addRecord() {
    fp = fopen("student.dat", "ab");
    printf("Enter Roll No: ");
    scanf("%d",&s.roll);
    printf("Enter Name: ");
    scanf("%s",s.name);
    printf("Enter Age: ");
    scanf("%d",&s.age);
    printf("Enter Marks: ");
```

```
    scanf("%f",&s.marks);

    fwrite(&s,sizeof(s),1,fp);
    fclose(fp);
    printf("Record Added Successfully!\n");
}

void displayRecords() {
    fp = fopen("student.dat", "rb");
    if(fp == NULL){
        printf("No Record Found!\n");
        return;
    }

    printf("\n--- Student Records ---\n");
    while(fread(&s,sizeof(s),1,fp)) {
        printf("\nRoll: %d", s.roll);
        printf("\nName: %s", s.name);
        printf("\nAge: %d", s.age);
        printf("\nMarks: %.2f\n", s.marks);
    }
    fclose(fp);
}

void searchRecord() {
    int r, found = 0;
    fp = fopen("student.dat", "rb");
    printf("Enter Roll No to Search: ");
    scanf("%d", &r);

    while(fread(&s,sizeof(s),1,fp)) {
        if(s.roll == r) {
```

```
            printf("\nRecord Found!\n");
            printf("Name: %s\n", s.name);
            printf("Age: %d\n", s.age);
            printf("Marks: %.2f\n", s.marks);
            found = 1;
            break;
        }
    }
    fclose(fp);

    if(!found)
        printf("Record Not Found!\n");
}

void deleteRecord() {
    int r, found = 0;
    FILE *ft;
    fp = fopen("student.dat", "rb");
    ft = fopen("temp.dat", "wb");

    printf("Enter Roll No to Delete: ");
    scanf("%d", &r);

    while(fread(&s,sizeof(s),1,fp)) {
        if(s.roll != r)
            fwrite(&s,sizeof(s),1,ft);
        else
            found = 1;
    }

    fclose(fp);
    fclose(ft);
```

```
    remove("student.dat");
    rename("temp.dat","student.dat");

    if(found)
        printf("Record Deleted!\n");
    else
        printf("Record Not Found!\n");
}

void modifyRecord() {
    int r, found = 0;
    fp = fopen("student.dat", "rb+");

    printf("Enter Roll No to Modify: ");
    scanf("%d", &r);

    while(fread(&s,sizeof(s),1,fp)) {
        if(s.roll == r) {
            found = 1;
            printf("Enter New Name: ");
            scanf("%s",s.name);
            printf("Enter New Age: ");
            scanf("%d",&s.age);
            printf("Enter New Marks: ");
            scanf("%f",&s.marks);

            fseek(fp, -sizeof(s), SEEK_CUR);
            fwrite(&s,sizeof(s),1,fp);
            printf("Record Updated Successfully!\n");
            break;
        }
```

```
    }
    fclose(fp);

    if(!found)
        printf("Record Not Found!\n");
}
```

# Conclusion

This project helped in understanding the practical applications of C language in handling and managing data. It provided knowledge about file operations, structures, and modular programming. It can be enhanced by adding a graphical interface or more student details.


**Submitted By-**

**Name** : Aditya Uniyal

**Roll No.** : 590027963

**Course** : B.Tech CSE

**College** : UPES

**Semester** : 1st

**Subject** : C Programming