C LANGUAGE PROJECT

NAME

NAMEER ALAM SIDDIQUI

ROLL NO

22F-BSAI-60

SUBJECT

PROGRAMMING FUNDAMENTAL

INSTRUCTOR

MISS NUR-UL-HUDA

Q: Write a c program to make a Student Management System.

CODE:

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

#define MAX_STUDENTS 20
#define MAX_NAME_LENGTH 50

struct Student
{
    char student_class[4];
    int id;
    char name[MAX_NAME_LENGTH];
    int age;
    float marks;
};

void addStudent(struct Student students[], int *count);
void displayStudents(const struct Student students[], int count);
void searchStudent(const struct Student students[], int count, int studentID);
float calculateAverageMarks(const struct Student students[], int count);

int main()
{
    struct Student students[MAX_STUDENTS];
    int student_count = 0;
    int studentID;

    printf("\tSTUDENT MANAGEMENT SYSTEM\n");

    while (1)
          {
        printf("\n Please select your Choice:\n");
        printf("1. Add Student\n");
        printf("2. Display Students\n");
        printf("3. Search Student\n");
        printf("4. Calculate Average Marks\n");
        printf("5. Exit\n");
        printf("Enter your choice: ");

        int choice;
        scanf("%d", &choice);
```

```
        switch (choice)

{
        case 1:
           addStudent(students, &student_count);
           break;
        case 2:
           displayStudents(students, student_count);
           break;
        case 3:
           if (student_count > 0)

            {
                printf("Enter student ID to search: ");
                scanf("%d", &studentID);
                searchStudent(students, student_count, studentID);
             }
             else
                  {
                printf("No students added yet.\n");
                   }
           break;
        case 4:
                 if(student_count > 0)
                 {
                          float average = calculateAverageMarks(students, student_count);
                          printf("Average Marks: %.2f\n",average);
                                  }
                                  else
                                  {
                                          printf("No students added yet.\n");
                                  }
                                  break;
        case 5:
           printf("EXITING PROGRAM...\n");
           exit(0);
        default:
           printf("Invalid choice. Please try again.\n");
      }
   }

   return(0);
}
```

```
void addStudent(struct Student students[], int *count)
{
    if (*count >= MAX_STUDENTS)
          {
        printf("Maximum number of students reached.\n");
        return;
    }

    struct Student newStudent;
    printf("Enter student Class: ");
    scanf("%s" ,newStudent.student_class);
    printf("Enter student ID: ");
    scanf("%d", &newStudent.id);
    printf("Enter student name: ");
    scanf(" %[^\n]s", newStudent.name);
    printf("Enter Age of student: ");
    scanf("%d" ,&newStudent.age);
    printf("Enter student marks: ");
    scanf("%f", &newStudent.marks);

    students[*count] = newStudent;
    (*count)++;

    printf("Student added successfully.\n");
}

void displayStudents(const struct Student students[], int count)
{
    if (count == 0)
          {
        printf("No students added yet.\n");
        return;
    }

    printf("\nStudent List:\n");
    for (int i = 0; i < count; i++)
          {
        printf("Student %d:\n", i + 1);
        printf("Class: %s\n",students[i].student_class);
        printf("ID: %d\n", students[i].id);
        printf("Name: %s\n", students[i].name);
        printf("Age: %d\n",students[i].age);
        printf("Marks: %.2f\n", students[i].marks);
```

```
        printf("\n");
    }
}

void searchStudent(const struct Student students[], int count, int studentID)
{
    int found = 0;

    for (int i = 0; i < count; i++)
          {
        if (students[i].id == studentID)
                  {
            found = 1;
            printf("Student found:\n");
            printf("Class: %s\n",students[i].student_class);
            printf("ID: %d\n", students[i].id);
            printf("Name: %s\n", students[i].name);
            printf("Marks: %.2f\n", students[i].marks);
            printf("\n");
            break;
        }
    }

    if (!found)
          {
        printf("Student with ID %d not found.\n", studentID);
    }
}
float calculateAverageMarks(const struct Student students[], int count)
{
          float sum = 0;
    for (int i = 0; i < count; i++)
          {
       sum += students[i].marks;
    }
    return sum / count;
}
```

OUTPUT

```
        STUDENT MANAGEMENT SYSTEM

 Please select your Choice:
1. Add Student
2. Display Students
3. Search Student
4. Calculate Average Marks
5. Exit
Enter your choice: 1
Enter student Class: 12
Enter student ID: 1001
Enter student name: Nameer
Enter Age of student: 18
Enter student marks: 500
Student added successfully.
```

```
 Please select your Choice:
1. Add Student
2. Display Students
3. Search Student
4. Calculate Average Marks
5. Exit
Enter your choice: 2

Student List:
Student 1:
Class: 12
ID: 1001
Name: Nameer
Age: 18
Marks: 500.00
```

```

 Please select your Choice:
1. Add Student
2. Display Students
3. Search Student
4. Calculate Average Marks
5. Exit
Enter your choice: 3
Enter student ID to search: 1001
Student found:
Class: 12
ID: 1001
Name: Nameer
Marks: 500.00
```

```
 Please select your Choice:
1. Add Student
2. Display Students
3. Search Student
4. Calculate Average Marks
5. Exit
Enter your choice: 4
Average Marks: 500.00

 Please select your Choice:
1. Add Student
2. Display Students
3. Search Student
4. Calculate Average Marks
5. Exit
Enter your choice: 5
EXITING PROGRAM...
```