

Program : Student Management System

Write a program to build a simple program for Student Information Management System which can perform the following operations:

Store the Name of the student.

Store the unique Roll number for every student.

Store the Marks of every student.

1.Add Student

2.Display All Students

3.Search student

4.Delete student

5.Exit

Source code:

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

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

```
#define MAX_STUDENTS 100
```

```
// Structure to represent a student
```

```
struct Student {
```

```
    char name[50];
```

```
    int rollNumber;
```

```
    float marks;
```

```
};
```

```
// Function prototypes
```

```
void addStudent(struct Student students[], int *numStudents);
```

```
void displayAllStudents(struct Student students[], int numStudents);
```

```
void searchStudent(struct Student students[], int numStudents);  
void deleteStudent(struct Student students[], int *numStudents);
```

```
int main() {  
    struct Student students[MAX_STUDENTS];  
    int numStudents = 0;  
    int choice;  
  
    do {  
        // Display menu  
        printf("\nStudent Management System\n");  
        printf("1. Add Student\n");  
        printf("2. Display All Students\n");  
        printf("3. Search Student\n");  
        printf("4. Delete Student\n");  
        printf("0. Exit\n");  
        printf("Enter your choice: ");  
        scanf("%d", &choice);  
  
        switch (choice) {  
            case 1:  
                addStudent(students, &numStudents);  
                break;  
            case 2:  
                displayAllStudents(students, numStudents);  
                break;  
            case 3:  
                searchStudent(students, numStudents);  
                break;
```

```

        case 4:
            deleteStudent(students, &numStudents);

            break;

        case 0:
            printf("Exiting program. Goodbye!\n");

            break;

        default:
            printf("Invalid choice. Please enter a valid option.\n");

    }

} while (choice != 0);

return 0;
}

```

// Function to add a new student

```

void addStudent(struct Student students[], int *numStudents) {
    if (*numStudents < MAX_STUDENTS) {
        struct Student newStudent;

        printf("Enter student name: ");
        scanf("%s", newStudent.name);

        printf("Enter roll number: ");
        scanf("%d", &newStudent.rollNumber);

        printf("Enter marks: ");
        scanf("%f", &newStudent.marks);

        students[*numStudents] = newStudent;

        (*numStudents)++;

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

```

```
    } else {  
        printf("Cannot add more students. Maximum limit reached.\n");  
    }  
}
```

// Function to display all students

```
void displayAllStudents(struct Student students[], int numStudents) {  
    if (numStudents > 0) {  
        printf("\nList of all students:\n");  
        for (int i = 0; i < numStudents; i++) {  
            printf("Name: %s, Roll Number: %d, Marks: %.2f\n",  
                students[i].name, students[i].rollNumber, students[i].marks);  
        }  
    } else {  
        printf("No students found.\n");  
    }  
}
```

// Function to search for a student by roll number

```
void searchStudent(struct Student students[], int numStudents) {  
    int rollNumber;  
    printf("Enter the roll number to search: ");  
    scanf("%d", &rollNumber);  
  
    for (int i = 0; i < numStudents; i++) {  
        if (students[i].rollNumber == rollNumber) {  
            printf("Student found!\n");  
            printf("Name: %s, Roll Number: %d, Marks: %.2f\n",  
                students[i].name, students[i].rollNumber, students[i].marks);  
        }  
    }  
}
```

```

        return;
    }
}

printf("Student not found with the given roll number.\n");
}

// Function to delete a student by roll number
void deleteStudent(struct Student students[], int *numStudents) {
    int rollNumber;

    printf("Enter the roll number to delete: ");
    scanf("%d", &rollNumber);

    for (int i = 0; i < *numStudents; i++) {
        if (students[i].rollNumber == rollNumber) {
            // Move all elements after the deleted student one position back
            for (int j = i; j < (*numStudents - 1); j++) {
                students[j] = students[j + 1];
            }
            (*numStudents)--;
            printf("Student deleted successfully!\n");
            return;
        }
    }

    printf("Student not found with the given roll number.\n");
}

```

Output:

Student Management System

1. Add Student
2. Display All Students
3. Search Student
4. Delete Student
0. Exit

Enter your choice: 1

Enter student name: sita

Enter roll number: 1

Enter marks: 87

Student added successfully!

Student Management System

1. Add Student
2. Display All Students
3. Search Student
4. Delete Student
0. Exit

Enter your choice: 1

Enter student name: geetha

Enter roll number: 2

Enter marks: 76

Student added successfully!

Student Management System

1. Add Student
2. Display All Students
3. Search Student
4. Delete Student
0. Exit

Enter your choice: 1

Enter student name: ram

Enter roll number: 3

Enter marks: 87

Student added successfully!

Student Management System

1. Add Student

2. Display All Students

3. Search Student

4. Delete Student

0. Exit

Enter your choice: 1

Enter student name: krishna

Enter roll number: 4

Enter marks: 89

Student added successfully!

Student Management System

1. Add Student

2. Display All Students

3. Search Student

4. Delete Student

0. Exit

Enter your choice: 1

Enter student name: radha

Enter roll number: 5

Enter marks: 85

Student added successfully!

Student Management System

1. Add Student
2. Display All Students
3. Search Student
4. Delete Student
0. Exit

Enter your choice: 1

Enter student name: rani

Enter roll number: 6

Enter marks: 93

Student added successfully!

Student Management System

1. Add Student
2. Display All Students
3. Search Student
4. Delete Student
0. Exit

Enter your choice: 1

Enter student name: rekha

Enter roll number: 7

Enter marks: 81

Student added successfully!

Student Management System

1. Add Student
2. Display All Students
3. Search Student

4. Delete Student

0. Exit

Enter your choice: 1

Enter student name: manasa

Enter roll number: 8

Enter marks: 90

Student added successfully!

Student Management System

1. Add Student

2. Display All Students

3. Search Student

4. Delete Student

0. Exit

Enter your choice: 1

Enter student name: akhila

Enter roll number: 9

Enter marks: 76

Student added successfully!

Student Management System

1. Add Student

2. Display All Students

3. Search Student

4. Delete Student

0. Exit

Enter your choice: 1

Enter student name: aravind

Enter roll number: 10

Enter marks: 91

Student added successfully!

Student Management System

1. Add Student
2. Display All Students
3. Search Student
4. Delete Student
0. Exit

Enter your choice: 1

Enter student name: saiharsha

Enter roll number: 11

Enter marks: 82

Student added successfully!

Student Management System

1. Add Student
2. Display All Students
3. Search Student
4. Delete Student
0. Exit

Enter your choice: 2

List of all students:

Name: sita, Roll Number: 1, Marks: 87.00

Name: geetha, Roll Number: 2, Marks: 76.00

Name: ram, Roll Number: 3, Marks: 87.00

Name: krishna, Roll Number: 4, Marks: 89.00

Name: radha, Roll Number: 5, Marks: 85.00

Name: rani, Roll Number: 6, Marks: 93.00

Name: rekha, Roll Number: 7, Marks: 81.00

Name: manasa, Roll Number: 8, Marks: 90.00

Name: akhila, Roll Number: 9, Marks: 76.00

Name: aravind, Roll Number: 10, Marks: 91.00

Name: saiharsha, Roll Number: 11, Marks: 82.00

Student Management System

1. Add Student

2. Display All Students

3. Search Student

4. Delete Student

0. Exit

Enter your choice: 3

Enter the roll number to search: 10

Student found!

Name: aravind, Roll Number: 10, Marks: 91.00

Student Management System

1. Add Student

2. Display All Students

3. Search Student

4. Delete Student

0. Exit

Enter your choice: 4

Enter the roll number to delete: 11

Student deleted successfully!

Student Management System

1. Add Student

2. Display All Students

3. Search Student

4. Delete Student

0. Exit

Enter your choice: 0

Exiting program. Goodbye!