#include <stdio.h>

#include <stdlib.h>

#include <string.h>

#define MAX 100

struct Student {

int roll;

char name[50];

float marks;

};

struct Student students[MAX];

int count = 0;

void addStudent() {

if (count < MAX) {

printf("Enter Roll Number: ");

scanf("%d", &students[count].roll);

getchar(); // clear buffer

printf("Enter Name: ");

fgets(students[count].name, 50, stdin);

students[count].name[strcspn(students[count].name, "\n")] = 0; // remove newline

printf("Enter Marks: ");

scanf("%f", &students[count].marks);

count++;

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

} else {

printf("Maximum student limit reached.\n\n");

}

}

void displayStudents() {

if (count == 0) {

printf("No student records found.\n\n");

return;

}

printf("\n--- Student Records ---\n");

for (int i = 0; i < count; i++) {

printf("Roll: %d | Name: %s | Marks: %.2f\n", students[i].roll, students[i].name, students[i].marks);

}

printf("\n");

}

void searchStudent() {

int roll;

printf("Enter Roll Number to Search: ");

scanf("%d", &roll);

for (int i = 0; i < count; i++) {

if (students[i].roll == roll) {

printf("Student Found:\n");

printf("Roll: %d | Name: %s | Marks: %.2f\n\n", students[i].roll, students[i].name, students[i].marks);

return;

}

}

printf("Student not found.\n\n");

}

void deleteStudent() {

int roll, found = 0;

printf("Enter Roll Number to Delete: ");

scanf("%d", &roll);

for (int i = 0; i < count; i++) {

if (students[i].roll == roll) {

for (int j = i; j < count - 1; j++) {

students[j] = students[j + 1];

}

count--;

found = 1;

printf("Student record deleted.\n\n");

break;

}

}

if (!found) {

printf("Student not found.\n\n");

}

}

int main() {

int choice;

while (1) {

printf("=== Student Record System ===\n");

printf("1. Add Student\n");

printf("2. Display All Students\n");

printf("3. Search Student\n");

printf("4. Delete Student\n");

printf("5. Exit\n");

printf("Enter choice: ");

scanf("%d", &choice);

switch (choice) {

case 1:

addStudent();

break;

case 2:

displayStudents();

break;

case 3:

searchStudent();

break;

case 4:

deleteStudent();

break;

case 5:

exit(0);

default:

printf("Invalid choice. Try again.\n\n");

}

}

return 0;

}