#include <stdio.h>

#include <stdlib.h>

#include <string.h>

#define MAX\_STUDENTS 100

#define MAX\_NAME\_LENGTH 50

typedef struct {

char name[MAX\_NAME\_LENGTH];

float grade;

} Student;

void addStudent(Student students[], int \*count);

void removeStudent(Student students[], int \*count);

void displayStudents(const Student students[], int count);

float calculateAverageGrade(const Student students[], int count);

int getValidGrade();

int main() {

Student students[MAX\_STUDENTS];

int count = 0;

int choice;

while (1) {

printf("\nStudent Grades Tracker\n");

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

printf("2. Remove Student\n");

printf("3. Display Students\n");

printf("4. Calculate Average Grade\n");

printf("5. Exit\n");

printf("Enter your choice: ");

if (scanf("%d", &choice) != 1) {

while(getchar() != '\n');

printf("Invalid input. Please enter a valid choice.\n");

continue;

}

switch (choice) {

case 1:

addStudent(students, &count);

break;

case 2:

removeStudent(students, &count);

break;

case 3:

displayStudents(students, count);

break;

case 4:

if (count > 0) {

float average = calculateAverageGrade(students, count);

printf("The average grade is: %.2f\n", average);

} else {

printf("No students to calculate average grade.\n");

}

break;

case 5:

printf("Exiting program.\n");

return 0;

default:

printf("Invalid choice. Please try again.\n");

}

}

}

void addStudent(Student students[], int \*count) {

if (\*count >= MAX\_STUDENTS) {

printf("Cannot add more students. The list is full.\n");

return;

}

printf("Enter the student's name: ");

getchar();

fgets(students[\*count].name, MAX\_NAME\_LENGTH, stdin);

students[\*count].name[strcspn(students[\*count].name, "\n")] = '\0';

printf("Enter the grade for %s: ", students[\*count].name);

students[\*count].grade = getValidGrade();

(\*count)++;

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

}

void removeStudent(Student students[], int \*count) {

if (\*count == 0) {

printf("No students to remove.\n");

return;

}

char name[MAX\_NAME\_LENGTH];

printf("Enter the name of the student to remove: ");

getchar();

fgets(name, MAX\_NAME\_LENGTH, stdin);

name[strcspn(name, "\n")] = '\0';

int index = -1;

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

if (strcmp(students[i].name, name) == 0) {

index = i;

break;

}

}

if (index != -1) {

for (int i = index; i < \*count - 1; i++) {

students[i] = students[i + 1];

}

(\*count)--;

printf("Student '%s' removed successfully.\n", name);

} else {

printf("Student '%s' not found.\n", name);

}

}

void displayStudents(const Student students[], int count) {

if (count == 0) {

printf("No students to display.\n");

return;

}

printf("\nStudent List:\n");

printf("%-30s %-10s\n", "Student Name", "Grade");

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

printf("%-30s %-10.2f\n", students[i].name, students[i].grade);

}

}

float calculateAverageGrade(const Student students[], int count) {

float total = 0.0;

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

total += students[i].grade;

}

return total / count;

}

int getValidGrade() {

float grade;

while (1) {

if (scanf("%f", &grade) != 1) {

while(getchar() != '\n');

printf("Invalid input. Please enter a valid grade: ");

} else if (grade < 0 || grade > 100) {

printf("Grade must be between 0 and 100. Please enter a valid grade: ");

} else {

break;

}

}

return grade;

}

