**ASSIGNMENT 2**  
#include <iostream>

using namespace std;

const int MAX\_COURSES = 5;

const int MAX\_STUDENTS = 100;

class Student {

private:

string name;

int rollNumber;

float cgpa;

string courses[MAX\_COURSES];

int courseCount;

public:

Student() {

name = "Unknown";

rollNumber = 0;

cgpa = 0.0;

courseCount = 0;

}

Student(string n, int roll, float gpa) {

name = n;

rollNumber = roll;

setCGPA(gpa);

courseCount = 0;

}

Student(const Student &s) {

name = s.name;

rollNumber = s.rollNumber;

cgpa = s.cgpa;

courseCount = s.courseCount;

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

courses[i] = s.courses[i];

}

}

~Student() {

cout << "Deleting student: " << name << endl;

}

void setCGPA(float gpa) {

if (gpa >= 0.0 && gpa <= 4.0)

cgpa = gpa;

else {

cout << "Invalid CGPA. Setting to 0.0\n";

cgpa = 0.0;

}

}

void addCourse(string courseName) {

if (courseCount < MAX\_COURSES) {

courses[courseCount] = courseName;

courseCount++;

} else {

cout << "Cannot add more than " << MAX\_COURSES << " courses.\n";

}

}

void displayInfo() {

cout << "Name: " << name << endl;

cout << "Roll Number: " << rollNumber << endl;

cout << "CGPA: " << cgpa << endl;

cout << "Courses Enrolled:\n";

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

cout << " - " << courses[i] << endl;

}

}

int getRollNumber() {

return rollNumber;

}

};

class StudentManagementSystem {

private:

Student students[MAX\_STUDENTS];

int studentCount;

public:

StudentManagementSystem() {

studentCount = 0;

}

void addStudent(const Student &s) {

if (studentCount < MAX\_STUDENTS) {

students[studentCount] = s;

studentCount++;

} else {

cout << "Cannot add more students.\n";

}

}

void searchStudent(int roll) {

bool found = false;

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

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

students[i].displayInfo();

found = true;

break;

}

}

if (!found) {

cout << "Student with roll number " << roll << " not found.\n";

}

}

void displayAllStudents() {

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

cout << "\n--- Student " << i + 1 << " ---\n";

students[i].displayInfo();

}

}

};

int main() {

StudentManagementSystem sms;

Student s1("Ali", 101, 3.5);

s1.addCourse("Math");

s1.addCourse("Physics");

Student s2("Sara", 102, 3.8);

s2.addCourse("Chemistry");

sms.addStudent(s1);

sms.addStudent(s2);

// Search for a student

cout << "\nSearching for roll number 101:\n";

sms.searchStudent(101);

// Display all student records

cout << "\nAll Students:\n";

sms.displayAllStudents();

return 0;

}