***Task 1: CGPA Calculator***

#include <iostream>

#include <iomanip> // for setprecision

using namespace std;

int main() {

int n;

cout << "Enter number of courses: ";

cin >> n;

double gradePoints[50]; // Store grade points (A=4, B=3, etc.)

double creditHours[50]; // Store credit hours

double totalCredits = 0, totalGradePoints = 0;

cout << "\nEnter grade points and credit hours for each course:\n";

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

cout << "Course " << i + 1 << ":\n";

cout << " Grade Points (A=4, B=3, C=2, D=1, F=0): ";

cin >> gradePoints[i];

cout << " Credit Hours: ";

cin >> creditHours[i];

totalCredits += creditHours[i];

totalGradePoints += gradePoints[i] \* creditHours[i];

}

double GPA = totalGradePoints / totalCredits; // semester GPA

double CGPA = GPA; // Simple version → GPA = CGPA

cout << fixed << setprecision(2); // show result with 2 decimal places

cout << "\n----- RESULT -----\n";

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

cout << "Course " << i + 1

<< " -> Grade Points: " << gradePoints[i]

<< ", Credit Hours: " << creditHours[i] << endl;

}

cout << "Semester GPA = " << GPA << endl;

cout << "Overall CGPA = " << CGPA << endl;

return 0;

}

***Task 2: Login and Registration System***

#include <iostream>

#include <fstream>

#include <string>

using namespace std;

struct User {

string username;

string password;

};

// Function to check if username already exists

bool isUserExists(const string &uname) {

ifstream fin("users.txt");

string u, p;

while (fin >> u >> p) {

if (u == uname) {

return true; // found existing user

}

}

return false;

}

// Registration function

void registerUser() {

User newUser;

cout << "\n===== Registration =====\n";

cout << "Enter username: ";

cin >> newUser.username;

// check if already exists

if (isUserExists(newUser.username)) {

cout << "Error: Username already exists! Try another.\n";

return;

}

cout << "Enter password: ";

cin >> newUser.password;

ofstream fout("users.txt", ios::app);

fout << newUser.username << " " << newUser.password << endl;

fout.close();

cout << "Registration successful!\n";

}

// Login function

void loginUser() {

string uname, pass;

cout << "\n===== Login =====\n";

cout << "Enter username: ";

cin >> uname;

cout << "Enter password: ";

cin >> pass;

ifstream fin("users.txt");

string u, p;

bool success = false;

while (fin >> u >> p) {

if (u == uname && p == pass) {

success = true;

break;

}

}

fin.close();

if (success) {

cout << "Login successful! Welcome, " << uname << "!\n";

} else {

cout << "Error: Invalid username or password.\n";

}

}

int main() {

int choice;

do {

cout << "\n==== User System Menu ====\n";

cout << "1. Register\n";

cout << "2. Login\n";

cout << "3. Exit\n";

cout << "Enter choice: ";

cin >> choice;

switch (choice) {

case 1: registerUser(); break;

case 2: loginUser(); break;

case 3: cout << "Exiting program...\n"; break;

default: cout << "Invalid choice! Try again.\n";

}

} while (choice != 3);

return 0;

}