**LAB 10 – UNARY BINARY OPERATORS**

#include<iostream>

#include<string>

using namespace std;

class Student

{

private:

string name;

int rollNo;

float marks[3];

float average;

string grade;

static int studentCount;

public:

Student(string name = "Unknown", int rollNo = 0)

{

this->name = name;

this->rollNo = rollNo;

for (int i = 0; i < 3; i++)

{

marks[i] = 0.0;

}

this->average = 0.0;

this->grade = "F";

studentCount++;

}

Student(string name, int rollNo, float mark1, float mark2, float mark3)

{

this->name = name;

this->rollNo = rollNo;

this->marks[0] = mark1;

this->marks[1] = mark2;

this->marks[2] = mark3;

this->average = 0.0;

this->grade = "F";

studentCount++;

}

Student(const Student &student)

{

name = student.name;

rollNo = student.rollNo;

for (int i = 0; i < 3; i++)

{

marks[i] = student.marks[i];

}

average = student.average;

grade = student.grade;

studentCount++;

}

static int getStudentCount()

{

return studentCount;

}

void calculateAverage()

{

float total = 0;

for (int i = 0; i < 3; i++)

{

total += marks[i];

}

average = total / 3;

}

void calculateGrade()

{

if (average >= 90)

{

grade = "A+";

} else if (average >= 80)

{

grade = "A";

} else if (average >= 75)

{

grade = "B+";

} else if (average >= 70)

{

grade = "B";

} else if (average >= 60)

{

grade = "C";

} else {

grade = "F";

}

}

void displayDetails() const

{

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

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

}

void displayDetails(bool showMarks) const

{

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

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

if (showMarks)

{

cout << "Marks in 3 subjects:";

for (int i = 0; i < 3; i++)

{

cout << " " << marks[i];

}

cout << endl;

}

}

void displayDetails(bool showMarks, bool showGrade) const

{

displayDetails(showMarks);

if (showGrade)

{

cout << "Average Marks: " << average << endl;

cout << "Grade: " << grade << endl;

}

}

Student& operator++()

{

for (int i = 0; i < 3; i++)

{

marks[i]++;

}

calculateAverage();

calculateGrade();

return \*this;

}

Student& operator--()

{

for (int i = 0; i < 3; i++)

{

marks[i]--;

}

calculateAverage();

calculateGrade();

return \*this;

}

Student operator-()

{

Student temp = \*this;

for (int i = 0; i < 3; i++)

{

temp.marks[i] = -temp.marks[i];

}

temp.calculateAverage();

temp.calculateGrade();

return temp;

}

Student operator+(const Student &s)

{

Student temp = \*this;

for (int i = 0; i < 3; i++)

{

temp.marks[i] += s.marks[i];

}

temp.calculateAverage();

temp.calculateGrade();

return temp;

}

Student operator-(const Student &s)

{

Student temp = \*this;

for (int i = 0; i < 3; i++)

{

temp.marks[i] -= s.marks[i];

}

temp.calculateAverage();

temp.calculateGrade();

return temp;

}

bool operator==(const Student &s)

{

return (this->average == s.average);

}

friend void compareStudents(const Student &s1, const Student &s2);

};

int Student::studentCount = 0;

void compareStudents(const Student &s1, const Student &s2)

{

cout << "\nComparing students " << s1.name << " and " << s2.name << ":\n";

if (s1.average > s2.average)

{

cout << s1.name << " has higher marks with an average of " << s1.average << endl;

} else if (s2.average > s1.average)

{

cout << s2.name << " has higher marks with an average of " << s2.average << endl;

} else

{

cout << "Both have the same average marks.\n";

}

}

int main()

{

Student students[3] =

{

Student(),

Student("Bob", 102, 76.0, 88.5, 91.0),

Student("Alice", 103, 67.5, 72.0, 81.0)

};

for(int i = 0; i < 3; i++)

{

students[i].calculateAverage();

students[i].calculateGrade();

}

++students[1];

--students[2];

Student negatedStudent = -students[1];

Student totalMarks = students[1] + students[2];

Student diffMarks = students[1] - students[2];

bool areEqual = (students[1] == students[2]);

cout << (areEqual ? "Students have equal averages" : "Students do not have equal averages") << endl;

cout << "\nBasic Details of Students:";

for(int i = 0; i < 3; i++)

{

students[i].displayDetails();

}

cout << "\n\nDetailed Information of Students:";

for(int i = 0; i < 3; i++)

{

students[i].displayDetails(true);

}

cout << "\n\nFull Information of Students:";

for(int i = 0; i < 3; i++)

{

students[i].displayDetails(true, true);

}

compareStudents(students[1], students[2]);

cout << "\nTotal number of students: " << Student::getStudentCount() << endl;

return 0;

}