Object Oriented Programming Lab Evaluation 3

1. Friend Function

Create a class `Box` with the following private attributes: `length` (**float**), `width` (**float**), `height` (**float**)

Objective:

1. Implement a friend function that compares two `Box` objects and determines which box has a larger volume.
2. Take user input to create two `Box` objects and call the friend function to find and print which box is larger.

Additional Requirements: How would you implement it if the box objects were needed to be taken as pointers?

Ans.: Here is how I would create a class named `Box` that meets the objectives mentioned above:

```
#include <iostream>
using namespace std;

class Box
{
    float length, width, height;

public:
    Box(float l, float w, float h)
    {
        length = l;
        width = w;
        height = h;
    }

    friend void compareBoxes(const Box &b1, const Box &b2);
};

void compareBoxes(const Box &b1, const Box &b2)
{
    float v1 = b1.length * b1.width * b1.height;
    float v2 = b2.length * b2.width * b2.height;
    if (v1 > v2)
        cout << "Box 1 has a larger volume." << endl;
    else
        cout << "Box 2 has a larger volume." << endl;
}

int main()
{
    float length, width, height;

    cout << "Enter the dimensions for Box 1:" << endl;
    cin >> length >> width >> height;
    Box b1(length, width, height);

    cout << endl;
    cout << "Enter the dimensions for Box 2:" << endl;
    cin >> length >> width >> height;
    Box b2(length, width, height);

    compareBoxes(b1, b2);
    return 0;
}
```

Now, we can update the C++ program like below to accept the box objects as pointers:

```cpp
#include <iostream>
using namespace std;

class Box
{
    float length, width, height;

public:
    Box(float l, float w, float h)
    {
        length = l;
        width = w;
        height = h;
    }

    friend void compareBoxes(const Box *b1, const Box *b2);
};

void compareBoxes(const Box *b1, const Box *b2)
{
    float v1 = b1->length * b1->width * b1->height;
    float v2 = b2->length * b2->width * b2->height;
    if (v1 > v2)
        cout << "Box 1 has a larger volume." << endl;
    else
        cout << "Box 2 has a larger volume." << endl;
}

int main()
{
    float length, width, height;

    cout << "Enter the dimensions for Box 1:" << endl;
    cin >> length >> width >> height;
    Box b1(length, width, height);

    cout << endl;
    cout << "Enter the dimensions for Box 2:" << endl;
    cin >> length >> width >> height;
    Box b2(length, width, height);

    compareBoxes(&b1, &b2);
    return 0;
}
```

Input:

```
+ Enter the dimensions for Box 1:
  3.14 1.67 3.33

+ Enter the dimensions for Box 2:
  9.81 9.86 1.21
```

Output:

```
Box 2 has a larger volume.
```

2. Student Grading System

Implement a C++ program that defines a class `Student` with the following private attributes: `name` (**string**), `studentID` (**int**), marks (float array of side 3 to hold marks of three subjects)

Instructions:

1. Create an array of 5 `Student` objects.
2. Write a function to input details for each student (`name` , `studentID` , `marks`)
3. Implement a function to display average marks of all students.
4. Implement a function to calculate and display the total marks for each student.

Ans.: Here is the implementation of a C++ program that satisfies the conditions above:

```
#include <iostream>
#define subjects 3
#define students 5
using namespace std;

class Student
{
    string name;
    int studentID;
    float marks[subjects];

public:
    void setData(int count)
    {
        cout << "Enter details for Student " << (count + 1)
```

```
<< endl;
        cout << "Name : ";
        cin >> name;
        cout << "ID   : ";
        cin >> studentID;
        cout << "Marks: ";
        for (int i = 0; i < subjects; ++i)
            cin >> marks[i];
        cout << endl;
    }

    float calculateTotalMarks()
    {
        float total = 0;
        for (int i = 0; i < subjects; ++i)
            total += marks[i];
        return total;
    }

    void displayTotalMarks()
    {
        cout << endl;
        cout << "Student Name: " << name << endl;
        cout << "Student ID  : " << studentID << endl;
        cout << "Total Marks : " << calculateTotalMarks()
<< endl;
    }
};

int main()
{
    int i = 0;
    float sum = 0;
    Student array[students];

    for (i = 0; i < students; i++)
    {
        array[i].setData(i);
        sum += array[i].calculateTotalMarks();
    }

    cout << endl;
    cout << "Individual marks of each student: ";
    for (i = 0; i < students; i++)
        array[i].displayTotalMarks();

    cout << endl;
    cout << "Average marks of all students: " << (sum /
students) << endl;

    return 0;
}
```

Input:

```
+ Enter details for Student 1
  Name : Shahriar
  ID   : 408
  Marks: 67 77 87

+ Enter details for Student 2
  Name : Fikrat
  ID   : 328
  Marks: 78 89 81

+ Enter details for Student 3
  Name : Manzirul
  ID   : 358
  Marks: 81 82 84

+ Enter details for Student 4
  Name : John
  ID   : 331
  Marks: 79 81 88
```

```
+ Enter details for Student 5
  Name : Shawon
  ID   : 333
  Marks: 81 73 85
```

Output:

```
Individual marks of each student:
Student Name: Shahriar
Student ID  : 408
Total Marks : 231

Student Name: Fikrat
Student ID  : 328
Total Marks : 248

Student Name: Manzirul
Student ID  : 358
Total Marks : 247

Student Name: John
Student ID  : 331
Total Marks : 248

Student Name: Shawon
Student ID  : 333
Total Marks : 239

Average marks of all students: 242.6
```

Code

You can find all the code snippets **here**.