**Lab Assignment 7**

|  |  |
| --- | --- |
| **Roll No.: A055** | **Name: Ibrahim Shaikh** |
| **Program: B. Tech**-**CSBS (2ND YEAR)** | **Date of Release: 5th September 2021** |
| **Batch:**  A/1 | **Date of Submission:** 18th September 2021 |

**PROBLEM STATEMENT:**

**Problem Statement 1:** Write a program to calculate the results of a student by combining internal and external marks out of 100. Write two separate classes as “Internal1” and “External1” with following data members common to both the classes. (Out of 50 marks for each class)

* + - * Marks of Math
      * Marks of OPP
      * Marks of SE
      * Marks of DS

Use constructor to input the marks for each class. Define binary operator overloading **using friend function** to calculate the final result by combining internal and external marks.

Perform following tasks:

1. Print the total marks subject wise.
2. Print the result depending on marks of individual subjects (Marks < 50 - fail)
3. Calculate total marks
4. Print percentage if student pass

**Problem Statement 2:** MCGM want to prepare the list of doctor area wise to assign the duty to take care of patients.In order to prepare the list, computer operator, enter the name of doctor and area. Finally, this list is sorted area wise, system uses operator overloading concept (function) to compare the area.

Implement comparison (<) operator function to compare the area alphabetically.

Use this function in the main to sort the list alphabetically on area.

Print the sorted list in the main function.

**Concept to be implemented:** Operator Overloading with friend function

**CODE:**

**Code for Problem Statement 1**

#include<iostream>

using namespace std;

class External1; //Declaration of a class

class Internal1

{

    float m, o, s, d;

    public: //access specifier

    void sdata()

    {

        cout<<"Enter the Internal Marks of Math (out of 50): ";

        cin>>m;

        cout<<"Enter the Internal Marks of OOP (out of 50): ";

        cin>>o;

        cout<<"Enter the Internal Marks of SE (out of 50): ";

        cin>>s;

        cout<<"Enter the Internal Marks of DSA (out of 50): ";

        cin>>d;

    }

    friend External1 operator+(Internal1, External1);      // friend function declaration

};

class External1

{

    float ma, oo, se, ds;

    public:

    void readdata()

    {

        cout<<"\nEnter the External Marks of Math (out of 50): ";

        cin>>ma;

        cout<<"Enter the External Marks of OOP (out of 50): ";

        cin>>oo;

        cout<<"Enter the External Marks of SE (out of 50): ";

        cin>>se;

        cout<<"Enter the External Marks of DSA (out of 50): ";

        cin>>ds;

    }

    friend External1 operator+(Internal1, External1);      // friend function declaration

};

External1 operator+(Internal1 i, External1 t)          //Operator overloading using friend function

{

    int p;

    cout<<"\nTotal Marks of Math (out of 100): "<<i.m+t.ma;

    if(i.m+t.ma<50)                                 //Checks whether total marks are below 50

    cout<<"\nResult: FAIL";

    else

    cout<<"\nResult: PASS";

    cout<<"\nTotal Marks of OOP (out of 100): "<<i.o+t.oo;

    if(i.o+t.oo<50)

    cout<<"\nResult: FAIL";

    else

    cout<<"\nResult: PASS";

    cout<<"\nTotal Marks of SE (out of 100): "<<i.s+t.se;

    if(i.s+t.se<50)

    cout<<"\nResult: FAIL";

    else

    cout<<"\nResult: PASS";

    cout<<"\nTotal Marks of DSA (out of 100): "<<i.d+t.ds;

    if(i.d+t.ds<50)

    cout<<"\nResult: FAIL";

    else

    cout<<"\nResult: PASS";

    p = (i.m+t.ma+i.o+t.oo+i.s+t.se+i.d+t.ds)/4;   //basically it is the

  // average of marks as they already are out of 100

    if(p>50)

    cout<<"\nPercentage = "<<p<<"%";

    return t;

}

int main()

{

    Internal1 a;

    External1 b;

    cout<<"Internal Marks:\n";

    a.sdata();

    cout<<"\nExternal Marks:";

    b.readdata();

    cout<<"\nTotal Marks and Result:";

    External1(a + b);             //operator+ used

    return 0;

}

**Code for Problem Statement 2**

#include<iostream>

#include<string.h>          //for using 'string'

using namespace std;

class Doctor

{

    public:             //access specifier

    string name;

    string area;

    Doctor()            //Constructor

    {

        name ="NA";

        area ="NA";

    }

    Doctor(string n, string a)      //Parameterized Constructor

    {

        name = n;

        area = a;

    }

    void display(int x)             //display func for printing

    {

        int i = x;

        cout<<"\n\nName of the Doctor "<<i<<": "<<name;

        cout<<"\nArea of the Doctor "<<i<<": "<<area;

    }

    bool operator <(Doctor& a)

    {

    if(area < a.area)

    {

        return true;

    }

    else

        return false;

    }

};

int main()

{

    string n, a;

    int x;

    Doctor d[10];       //array of object

    Doctor temp;        //temp object for swapping

    cout<<"Enter the number of Doctors Avalaible: ";

    cin>>x;

    for(int i=1; i<=x; i++)             //for taking user input

    {

        cout<<"\nEnter the Name of Doctor "<<i<<": ";

        cin>>n;

        cout<<"Enter the Area of the Doctor "<<i<<": ";

        cin>>a;

        d[i]= Doctor(n,a);               //assigning the values

    }

    cout<<"\nThe List of Doctors: ";      //displaying without sorting

    for(int i=1; i<=x; i++)

    {

        d[i].display(i);

    }

    cout<<"\n\nThe List of Doctors area wise: ";    //displaying after sorting

    for(int i=1; i<=x; i++)

    {

        for(int j=1; j<=x-i; j++)

        {

        if(d[j+1].area<d[j].area)  //sorting area wise using operator <

        {

            temp = d[j+1];

            d[j+1] = d[j];

            d[j] = temp;

        }

        }

        d[i].display(i);

    }

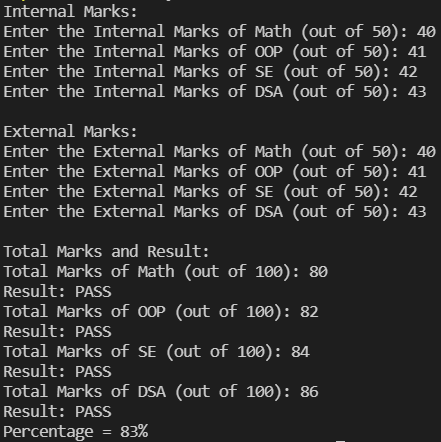
    return 0;

}

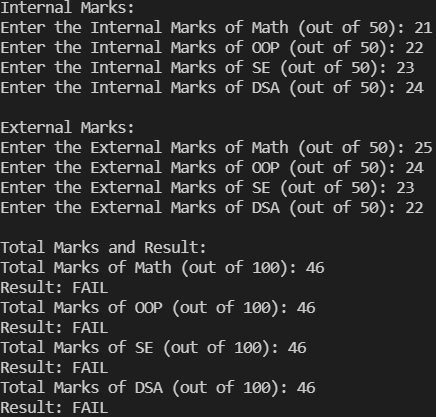
**OUTPUT:**

**Output for Problem Statement 1**

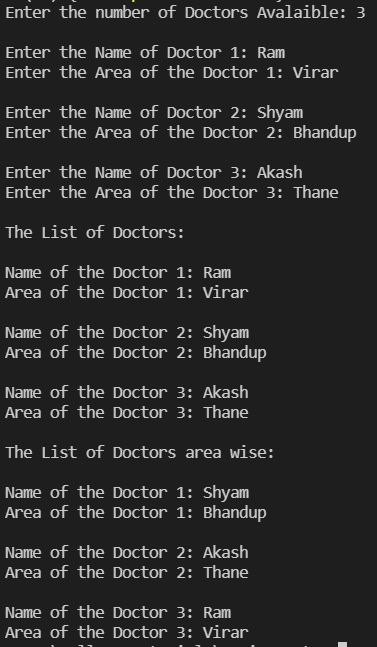
* If Pass



* If Fail



**Output for Problem Statement 2**

****

**LINK FOR THE CODE:**

Files for all Assignments: [GitHub - IamIbra8/OOP-Assignments: Assignment 6](https://github.com/IamIbra8/OOP-Assignments)

Code for Assignment 7: https://github.com/IamIbra8/Assignment-7