

OOP LABORATORY 10

Name: **ANIRBAN HAZRA**

Section: **B-12**

Roll : **2005643**

1. Create a class which stores two integers. Overload all the arithmetic operators for the class

- i) as member function
- ii) as friend function

PROGRAM CODE:-

// Using member function.

```
#include<iostream>
using namespace std;
class data
{
    int a;
    int b;
public:
    void input()
    {
        cin>>a>>b;
    }

    void output()
    {
        cout<<a<<"\t"<<b;
    }

    friend data operator+(data d1,data d2)
    {
        data d3;
        d3.a=d2.a+d1.a;
        d3.b=d2.b+d1.b;
        return d3;
    }

    friend data operator-(data d1,data d2)
    {
        data d3;
        d3.a=d1.a-d2.a;
        d3.b=d1.b-d2.b;
        return d3;
    }
}
```

```

friend data operator*(data d1,data d2)
{
    data d3;
    d3.a=d2.a*d1.a;
    d3.b=d2.b*d1.b;
    return d3;
}
friend data operator/(data d1,data d2)
{
    data d3;
    d3.a=d1.a/d2.a;
    d3.b=d1.b/d2.b;
    return d3;
}
};

int main()
{
    data d1,d2,sum,diff,pro,div;
    cout<<"Enter 2 integers for 1st object: ";
    d1.input();
    cout<<"Enter 2 integers for 2nd object: ";
    d2.input();
    sum=d1+d2;
    diff=d1-d2;
    pro=d1*d2;
    div=d1/d2;
    cout<<"\nSum of the 2 objects: ";
    sum.output();
    cout<<"\nDifference of the 2 objects: ";
    diff.output();
    cout<<"\nProduct of the 2 objects: ";
    pro.output();
    cout<<"\nDivision of the 2 objects: ";
    div.output();
    return 0;
}

```

OUTPUT:-

```

Enter 2 integers for 1st object: 6 7
Enter 2 integers for 2nd object: 8 9

Sum of the 2 objects: 14      16
Difference of the 2 objects: -2 -2
Product of the 2 objects: 48  63
Division of the 2 objects: 0   0

```

// Using Friend function.

PROGRAM CODE:-

```
#include<iostream>
using namespace std;
class data
{
    int a;
    int b;
public:
    void input()
    {
        cin>>a>>b;
    }

    void output()
    {
        cout<<a<<"\t"<<b;
    }
    friend data operator+(data d1,data d2)
    {
        data d3;
        d3.a=d2.a+d1.a;
        d3.b=d2.b+d1.b;
        return d3;
    }

    friend data operator-(data d1,data d2)
    {
        data d3;
        d3.a=d1.a-d2.a;
        d3.b=d1.b-d2.b;
        return d3;
    }

    friend data operator*(data d1,data d2)
    {
        data d3;
        d3.a=d2.a*d1.a;
        d3.b=d2.b*d1.b;
        return d3;
    }

    friend data operator/(data d1,data d2)
    {
        data d3;
        d3.a=d1.a/d2.a;
        d3.b=d1.b/d2.b;
        return d3;
    }
};
```

```

int main()
{
    data d1,d2,sum,diff,pro,div;
    cout<<"Enter 2 integers for 1st object: ";
    d1.input();
    cout<<"Enter 2 integers for 2nd object: ";
    d2.input();
    sum=d1+d2;
    diff=d1-d2;
    pro=d1*d2;
    div=d1/d2;
    cout<<"\nSum of the 2 objects: ";
    sum.output();
    cout<<"\nDifference of the 2 objects: ";
    diff.output();
    cout<<"\nProduct of the 2 objects: ";
    pro.output();
    cout<<"\nDivision of the 2 objects: ";
    div.output();
    return 0;
}

```

OUTPUT:-

```

Enter 2 integers for 1st object: 4 6
Enter 2 integers for 2nd object: 7 9

Sum of the 2 objects: 11      15
Difference of the 2 objects: -3 -3
Product of the 2 objects: 28   54
Division of the 2 objects: 0    0

```

2. Create a class complex which stores real and imaginary parts of a complex number.

Overload the following operators as member functions

+: to add two complex numbers

-: to subtract one complex number from another

*: to multiply two complex numbers

PROGRAM CODE:-

```

#include<iostream>
using namespace std;
class complex
{
    float real;

```

```

float img;
public:
    void getdata()
    {
        cin>>real>>img;
    }

    void putdata()
    {
        cout<<real<<"+"<<img<<"i";
    }

    complex operator+(complex c)
    {
        complex c1;
        c1.real=real+c.real;
        c1.img=img+c.img;
        return c1;
    }

    complex operator-(complex c)
    {
        complex c1;
        c1.real=real-c.real;
        c1.img=img-c.img;
        return c1;
    }

    complex operator*(complex c1)
    {
        complex c3;
        float a,b,c,d;
        a=real*c1.real;
        b=real*c1.img;
        c=img*c1.real;
        d=img*c1.img;
        c3.real=a-d;
        c3.img=b+c;
        return c3;
    }

};

int main()
{
    complex c1,c2,sum,diff,pro;
    cout<<"Enter the real and imaginary part of 1st complex number\n";
    c1.getdata();
    cout<<"Enter the real and imaginary part of 2nd complex number\n";
    c2.getdata();

```

```

sum=c1+c2;
diff=c1-c2;
pro=c1*c2;

cout<<"\n";
c1.putdata();
cout<<" + ";
c2.putdata();
cout<<" = ";
sum.putdata();

cout<<"\n";
c1.putdata();
cout<<" - ";
c2.putdata();
cout<<" = ";
diff.putdata();

cout<<"\n";
c1.putdata();
cout<<" * ";
c2.putdata();
cout<<" = ";
pro.putdata();

return 0;
}

```

OUTPUT:-

```

Enter the real and imaginary part of 1st complex number
5 7
Enter the real and imaginary part of 2nd complex number
8 9

5+7i + 8+9i = 13+16i
5+7i - 8+9i = -3+-2i
5+7i * 8+9i = -23+101i

```

3. WAP to overload following operators for class distance, which stores the distance in feet and inches. Overload binary plus and minus operators for the class, as friend function.

PROGRAM CODE:-

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

class Distance {
private:
    int feet;
    int inches;

public:

    Distance() {
        feet = 0;
        inches = 0;
    }
    Distance(int f, int i) {
        feet = f;
        inches = i;
    }
    friend ostream &operator<<( ostream &output, const Distance &D ) {
        output<< D.feet<< "f " << D.inches << "i ";
        return output;
    }

    friend istream &operator>>( istream &input, Distance &D ) {
        input >> D.feet >> D.inches;
        return input;
    }

    Distance operator -(Distance d){
        Distance temp;
        temp.feet = feet-d.feet;
        temp.inches = inches-d.inches;
        return temp;
    }

    Distance operator +(Distance d){
        Distance temp;
        temp.feet = feet+d.feet;
        temp.inches = inches+d.inches;
        return temp;
    }
};
```

```

int main()
{
    Distance D1,D2,D3,D4;

    cout << "Enter the value of object : " << endl;
    cin >> D1;
    cout << "Enter the value of object : " << endl;
    cin >> D2;
    cout << "First Distance : " << D1 << endl;
    cout << "Second Distance : " << D2 << endl;

    D3 = D1 + D2;
    D4 = D1 - D2;
    cout << "Add =" << D3 << endl;
    cout << "subtact=" << D4 << endl;
    return 0;
}

```

OUTPUT:-

```

Enter the value of object :
5 6
Enter the value of object :
6 5
First Distance : 5f 6i
Second Distance : 6f 5i
Add =11f 11i
subtact=-1f 1i

```