

Course Name: A Level (2nd Sem)**Topic:** Operator Overloading using Friend Function**Subject:** Data Structure using C++**Date:** 08-04-2020**Overload Binary Operator using Friend Function**

If you define operator function as a **friend function** then it will accept **two arguments**. Because friend functions is not a member function so it is not invoked using object of the class. Thus we need to pass two objects as an argument explicitly.

Example1: To overload binary operator + using friend function.

```
#include <iostream.h>
class demo
{
    int x,y;
public:
    demo()
    {
        x=0;
        y=0;
    }
    demo(int a, int b)
    {
        x=a;
        y=b;
    }
    friend demo operator + (demo &d1, demo &d2)
    {
        demo d3;
        d3.x = d1.x + d2.x;
        d3.y = d1.y + d2.y;
        return d3;
    }
    void display()
    {
        cout<<"X="<<x<<endl;
        cout<<"Y="<<y<<endl;
    }
};

int main()
{
    demo d1(2,3);
    demo d2(4,5);
    demo d3;
    d3 = operator + (d1,d2);
    cout<<"Object C1\n";
    d1.display();
    cout<<"Object C2\n";
    d2.display();
    cout<<"Object C3\n";
    d3.display();
}
```

```
return 0;
}
```

Output:

```
Object C1
X=2
Y=3
Object C2
X=4
Y=5
Object C3
X=6
Y=8
```

Example2 : Write a C++ program to add two complex numbers using operator overloaded by a friend function.

```
#include<iostream>
using namespace std;
class Complex
{
    int num1, num2;
public:
    void accept()
    {
        cout<<"\n Enter Two Complex Numbers : ";
        cin>>num1>>num2;
    }

    //Overloading '+' operator using Friend function
    friend Complex operator+(Complex c1, Complex c2);

    void display()
    {
        cout<<num1<<"+"<<num2<<"i"<<"\n";
    }
};

Complex operator+(Complex c1, Complex c2)
{
    Complex c;
    c.num1=c1.num1+c2.num1;
    c.num2=c1.num2+c2.num2;
    return(c);
}

int main()
{
    Complex c1,c2, sum;    //Created Object of Class Complex i.e c1 and c2

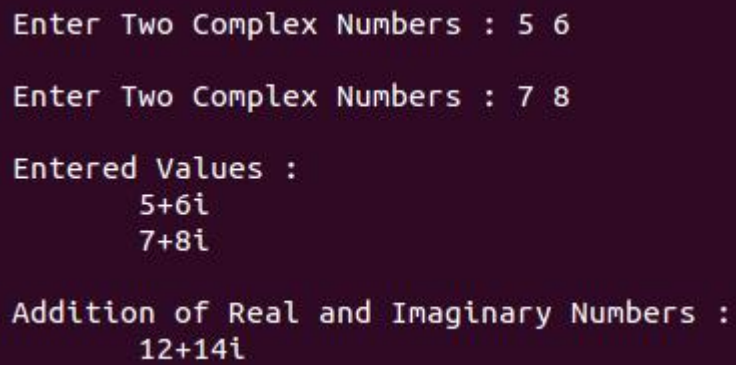
    c1.accept(); //Accepting the values
    c2.accept();

    sum = c1+c2; //Addition of object

    cout<<"\n Entered Values : \n";
    cout<<"\t";
    c1.display(); //Displaying user input values
```

```
        cout<<"\t";  
        c2.display();  
  
        cout<<"\n Addition of Real and Imaginary Numbers : \n";  
        cout<<"\t";  
        sum.display(); //Displaying the addition of real and imaginary numbers  
  
        return 0;  
    }
```

Output:

A screenshot of a terminal window with a dark purple background and light green text. The output shows the program's execution flow: it prompts for two complex numbers, displays the entered values in standard form, and then shows the result of their addition.

```
Enter Two Complex Numbers : 5 6  
Enter Two Complex Numbers : 7 8  
Entered Values :  
    5+6i  
    7+8i  
Addition of Real and Imaginary Numbers :  
    12+14i
```