NAME	**SURAJ VASANTRAO WARBHE**
ROLL NO.	**231066**
GR NO.	**21910631**
SUBJECT	**OOP**
ASSIGNMENT NO.	**02(A)**
DATE	**04 September 2020**

****

ASSIGNMENT NO. : 2(A)

Design a class 'Complex 'with data members for real and imaginary part. Provide default and parameterized constructors. Write a program to perform Arithmetic operations of two complex numbers using operator overloading. Addition and subtraction using friend functions.

AIM : To design a class 'Complex 'with data members for real and imaginary part. Provide default and parameterized constructors and write a program to perform Arithmetic operations of two complex numbers using operator overloading like Addition and subtraction using friend functions.

OBJECTIVE : To understand and implement the concept of

1. Class and Objects
2. Operator overloading
3. Friend Function in class

THEORY :

1. OPERATOR OVERLOADING :

   An overloaded declaration is a declaration that is declared with the same name as a previously declared declaration in the same scope, except that both declarations have different arguments and different definition (implementation). In C++, we can change the way operators work for user-defined types like objects and structures. This is known as operator overloading.

   SYNTAX :

To overload an operator, we use a special operator function.

```
class className
{
 ... .. ...
      Public :
      returnType operator symbol (arguments)
         {
            ... .. ...
         }
 ... .. ...
};
```

Here,

a. return Type is the return type of the function.
b. operator is a keyword.
c. symbol is the operator we want to overload. Like: +, <, -, ++, etc.
d. arguments is the arguments passed to the function.

2. FRIEND FUNCTION :

If a function is defined as a friend function in C++, then the protected and private data of a class can be accessed using the function. Friend function can be represent by using keyword 'friend'. For accessing the data, the declaration of a friend function should be done inside the body of a class starting with the keyword friend.

Declaration of friend function in C++ :

```
class class_name
{
 friend data_type function_name(argument/s);    // syntax of friend function.
};
```

The function can be defined anywhere in the program like a normal C++ function. The function definition does not use either the keyword friend or scope resolution operator.

PROGRAM CODE :

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

class complex
{
  public:
        float real;
        float imag;
  public:
        complex()
        {
                real = 0;
                imag = 0;
        }

        void ACCEPT()
        {
                cout << "\nReal Part : ";
                cin >> real;
                cout << "Imaginary Part : ";
                cin >> imag;
        }

        friend complex operator+(complex m, complex n);
```

```
        friend complex operator-(complex m, complex n);

        void DISPLAY()
        {
                cout << "Entered Number is : ";
                cout << real << "+" << imag << "i"
                        << "\n";
        }
};

complex operator+(complex m, complex n)
{
        complex c;
        c.real = m.real + n.real;
        c.imag = m.imag + n.imag;
        return c;
}

complex operator-(complex m, complex n)
{
        complex c;
        c.real = m.real - n.real;
        c.imag = m.imag - n.imag;
        return c;
}
```

```
int main()
{
	complex c1, c2, c3;
	int choice = 0;

	cout << "\nEnter First Complex Number";
	c1.ACCEPT();
	c1.DISPLAY();
	cout << "\nEnter Second Complex Number";
	c2.ACCEPT();
	c2.DISPLAY();

	do
	{
		cout << "\n*****MENU*****\n\n(1) ADDITION\n(2) SUBTRACTION\n(3) EXIT \n\n*****************\n\n";
		cout << "Enter your choice : ";
		cin >> choice;

		switch (choice)
		{
		case 1:
			c3 = c1 + c2;
			c3.DISPLAY();
			break;
		case 2:
			c3 = c1 - c2;
```

```
                    c3.DISPLAY();
                    break;
                }
        } while (choice != 3);

        return 0;
}
```

OUTPUT :

```
Enter First Complex Number
Real Part : 12
Imaginary Part : 8
Entered Number is : 12+8i

Enter Second Complex Number
Real Part : 5
Imaginary Part : 6
Entered Number is : 5+6i

*****MENU*****

(1) ADDITION
(2) SUBTRACTION
```

(3) EXIT

****

Enter your choice : 1

Entered Number is : 17+14i

*****MENU*****

(1) ADDITION

(2) SUBTRACTION

(3) EXIT

****

Enter your choice : 2

Entered Number is : 7+2i

*****MENU*****

(1) ADDITION

(2) SUBTRACTION

****

Enter your choice : 3

(Program finished)

CONCLUSION : This Assignment helps us to learn basic concepts of class, objects, friend function and operator overloading in the 'C++' Programming language. We got an insight about how to declare friend function of class in program, accept and display it and how to perform operator overloading in the class.