**EXPERIMENT 6**

**Aim :**

Write a program to define a class Complex to represent set of all complex numbers. Overload ‘**+**’ operator to add two complex numbers using member function of the class. Overload ‘**\***’ operator to multiply two complex numbers using friend function of the class.

**Code :**

#include <iostream>

#include <cmath>

using namespace std;

class complex

{

private:

    float real;

    float img;

public:

    void getdata();

    void display();

    complex()

    { real = img = 0; }

    complex operator+(complex c1);

    friend complex operator\*(complex c1, complex c2);

};

void complex::display()

{

    cout << "(" << real << ")"

         << "+"

         << "(" << img << ")"

         << "i";

}

void complex::getdata()

{

    cout << "\nEnter the Real and Img part of Complex Number\n";

    cout << "Real : ";

    cin >> real;

    cout << "Img : ";

    cin >> img;

}

complex complex::operator+(complex c1)

{

    complex add;

    add.real = real + c1.real;

    add.img = img + c1.img;

    return (add);

}

complex operator\*(complex c1, complex c2)

{

    complex mul;

    mul.real = (c1.real \* c2.real) - (c1.img \* c2.img);

    mul.img = (c1.real \* c2.img) + (c1.img \* c2.real);

    return (mul);

}

int main()

{

    complex a, b, c;

    int opt;

    a.getdata();

    b.getdata();

    cout << "\n\t\t---Main Menu---\n\t1.Addition\n\t2.Multiplication";

    cout << "\n\t3.Exit\nEnter your choice---> ";

    cin >> opt;

    switch (opt)

    {

    case 1:

        c = a + b;

        cout << "\nResult : ";

        a.display();

        cout << " + ";

        b.display();

        cout << " = ";

        c.display();

        break;

    case 2:

        c = a \* b;

        cout << "\nResult : ";

        a.display();

        cout << " \* ";

        b.display();

        cout << " = ";

        c.display();

        break;

    case 3:

        return 0;

    default:

        cout << "\nInvalid choice\n";

        break;

    }

    return 0;

}

**Output Screenshot :**

