1. The objective of this experiment is to create a userdefined class Complex to represent a complex number. This includes: • Constructors for initialization • Operator overloading for addition and multiplication • Overloading input/output operators to facilitate easy handling of complex numbers.

***CODE :-***

#include<iostream>

using namespace std ;

class complex

{

int x ;

int y ;

public:

complex()

{

x = 0 ;

y = 0 ;

}

complex operator +(const complex &c)

{

complex temp ;

temp.x = x+c.x ;

temp.y = y+c.y;

return temp;

}

complex operator \*(const complex &c)

{

complex temp ;

temp.x = x\*c.x ;

temp.y = y\*c.y;

return temp;

}

friend istream & operator>>(istream & input , complex &t )

{

cout <<"Enter the imagnary number" <<endl ;

input>>t.x ;

cout<<"Enter the real number" <<endl;

input>> t.y ;

return input ;

}

friend ostream & operator<<(ostream & output ,const complex &t)

{

output<<t.x << "i" << t.y;

return output ;

}

};

int main()

{

complex c1 , c2 ,add , mult ;

cout<<"Default constructor" <<endl;

cout<< c1 ;

cout<<"Enter the First number "<<endl;

cin>>c1;

cout<<"Enter Second Number"<<endl;

cin>>c2;

add = c1+c2 ;

mult = c1\*c2 ;

cout<<"Addtion "<<add << endl;

cout<<"Multiplication"<< mult <<endl;

}

***OUTPUT :-***

