Rajvaibhav Rahane

17u283 223045

SE-C Comp,Viit,Pune

**Aim:- To Study Operator Overloading in C++**

**THEORY :-**

**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 obviously different definition (implementation).

***CODE:***

#include<iostream>

using namespace std;

class ComplexNumber{

private:

float imaginary,real;

public:

ComplexNumber(){

cout<<"Default Constructor called\n";

imaginary=real=0;

}

ComplexNumber(float real,float imaginary){

cout<<"Parameterised constructor called\n";

this->imaginary=imaginary;

this->real=real;

}

friend istream & operator>>(istream&,ComplexNumber&);

friend ostream & operator<<(ostream &,const ComplexNumber);

ComplexNumber operator+(const ComplexNumber);

ComplexNumber operator\*(const ComplexNumber);

};

istream & operator>>(istream& in,ComplexNumber& complexNumber){

cout<<"Enter Real Part:\t";

in>>complexNumber.real;

cout<<"Enter Imaginary Part:\t";

in>>complexNumber.imaginary;

return in;

}

ostream & operator<<(ostream & out,const ComplexNumber complexNumber){

out<<complexNumber.real;

if(complexNumber.imaginary>=0)

out<<"+";

out<<complexNumber.imaginary<<"i\n";

return out;

}

ComplexNumber ComplexNumber::operator+(const ComplexNumber secondNumber){

cout<<"In operator+ fx\n";

return ComplexNumber(real+secondNumber.real,imaginary+secondNumber.imaginary);

}

ComplexNumber ComplexNumber::operator\*(const ComplexNumber secondNumber){

cout<<"In operator\* fx\n";

return ComplexNumber(real\*secondNumber.real-imaginary\*secondNumber.imaginary,real\*secondNumber.imaginary+imaginary\*secondNumber.real);

}

int main(){

ComplexNumber c1;

cin>>c1;

cout<<c1<<endl;

ComplexNumber c2(4.5,5.5);

cout<<c2<<endl;

ComplexNumber c3=c1+c2;

cout<<"Sum : "<<c3<<endl;

ComplexNumber c4=c1\*c2;

cout<<"Product : "<<c4;

return 0;

}

***Output:***



Conclusion-Operator Overloading of Extraction, Insertion,+ and \* operators were understood