Function define outside the class

Pointer variable of the class

#include <iostream>

using namespace std;

class Complex1

{

int real; //data members

int imag;

public:

void input();

void display();

};

void Complex1::input() //member function

{

cout<<"enter real part"<<endl;

cin>>real;

cout<<"enter imag part"<<endl;

cin>>imag;

}

void Complex1::display()

{

cout<<real<<" + i"<<imag<<endl;

}

int main()

{

Complex1 c1;

Complex1 \*p;

cout<<"address of c1"<<&c1<<endl;

p=&c1;

cout<<"value of of p"<<p<<endl;

cout<<"address of "<<&p<<endl;

c1.input();

p->display();

}

Constructor

PolyMorphism

one form many action

Compile time polymorphism

1. function overloading

2. Operator overloading

Runtime Polymorphism

1. Function overriding

virtual functions

function overloading -- it is a process of defining more than one function with same name and different signature

signature can be different in two ways

1. No of arguments

2. type of arguments

Complex1::Complex1() //default constructor

{

cout<<"enter real part"<<endl;

cin>>real;

cout<<"enter imag part"<<endl;

cin>>imag;

}

Complex1::Complex1(int r,int i) // parameterized constructor

{

real=r;

imag=i;

}

Complex1::Complex1(int r) // parameterized constructor

{

real=r;

imag=r;

}