# 实验七 运算符重载

程序代码：

#include<iostream>

using namespace std;

class Complex {

public:

double real, img;

Complex(double r = 0 ,double i = 0)

{

real = r;

img = i;

}

};

Complex operator\*(Complex a, Complex b)

{

Complex temp;

temp.real = a.real \* b.real-a.img\*b.img;

temp.img = a.real \* b.img + a.img \* b.real;

return temp;

}

int main()

{

double a,b,c,d;

cout << "请输入两个复数的实部和虚部" << endl;

cin >> a >> b >> c >> d;

Complex A(a, b);

Complex B(c, d);

Complex temp;

temp = A \* B;

cout << "乘积的实部是：" << temp.real << endl;

cout << "乘积的虚部是：" << temp.img << endl;

return 0;

}

运行结果：



心得体会：

运算符重载实现了c++的多态性，它是通过创建运算符重载函数来实现的。

1.c++中有少数运算符不能重载，例如：. \* :: sizeof ？：

2.c++中只能对已有的c++运算符进行重载，不允许用户自己定义新的运算符。

3.重载的功能应与原有的功能相似。

Copyright ©2021-2099 Tianhang Li. All rights reserved