7.

#include<iostream>

using namespace std;

class Complex {

public:

double real;

double imag;

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

{

real = r; imag = i;

}

};

Complex operator\*(Complex co1, Complex co2)

{

Complex temp;

temp.real = co1.real \* co2.real - co1.imag \* co2.imag;

temp.imag = co1.imag \* co2.real + co2.imag \* co1.real;

return temp;

}

int main()

{

cout << "请分别输入两个复数的实部和虚部。" << endl;

double x[2], y[2];

cin >> x[0] >> y[0] >> x[1] >> y[1];

Complex com1(x[0], y[0]), com2(x[1], y[1]), total;

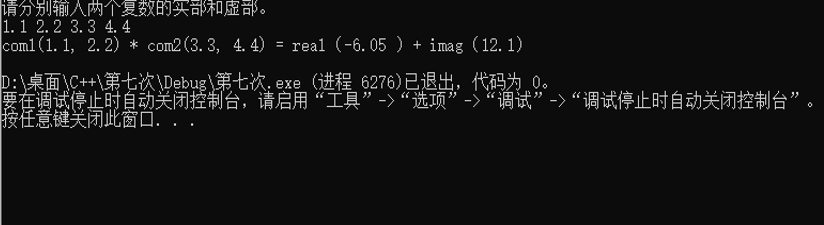
total = com1 \* com2;

cout << "com1(1.1, 2.2) \* com2(3.3, 4.4) = real（" << total.real << " " << "）+ imag（" << total.imag << "）" << endl;

return 0;

}

运行结果：



感想：

1. 掌握了C++语言多态性的基本概念。
2. 明白了运算符重载函数的声明和定义方法。
3. 明白了C++强大的原因之一是支持泛型编程。泛型编程使得算法可以独立于数据类型存在。

Copyright ©2021-2099 hukun. All rights reserved