程序代码

#include<iostream>

using namespace std;

class Complex {

double real;

double imag;

public:

Complex(double r = 0.0, double i = 0.0);

void print();

Complex operator\*(Complex c);

};

Complex::Complex(double r, double i)

{

real = r; imag = i;

}

Complex Complex::operator\*(Complex c)

{

Complex temp;

temp.real = real \* c.real;

temp.imag = imag \* c.imag;

return temp;

}

void Complex::print()

{

cout << "total real=" << real << " " << " total imag=" << imag << endl;

}

int main()

{

Complex com1(2.3, 4.6), com2(3.6, 2.8), total1;

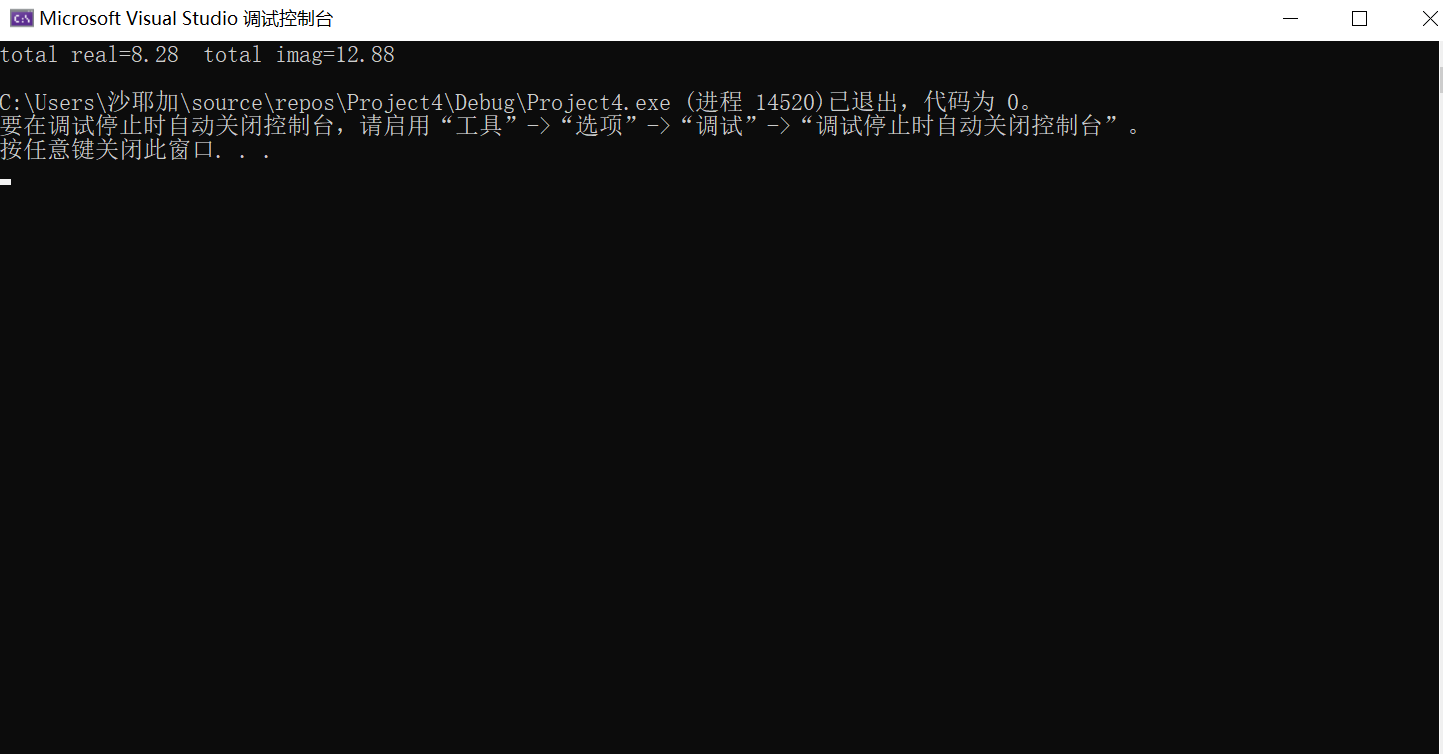
total1 = com1 \* com2;

total1.print();

return 0;

}

程序结果



感想心得

了解了C++语言多态性的基本概念和运算符重载函数的声明和定义方法，从系统实现的角度看，多态性分为两类：静态多态性和动态多态性。

重载函数常用来实现功能类似而所处理的数据类型不同的问题。不能只有函数返回值类型不同。

重载函数是函数的一种特殊情况，为方便使用，C++允许在同一范围中声明几个功能类似的同名函数，

Copyright IMG_2562021-2099 XiaohanYang. All rights reserved.