**上机七—运算符重载**

**一程序代码**

#include<iostream>

using namespace std;

class complex {

private:

double real;

double imag;

public:

complex(double a, double b)

{

real = a;

imag = b;

cout << "调用complex构造函数" << endl;

}

void print()

{

cout<<real;

if(imag>0) cout<<"+";

if(imag!=0)cout<<imag<<"i"<<endl;

}

complex operator\*(complex c);

};

complex complex::operator\*(complex c)

{

complex temp(0,0);

temp.real = real \* c.real - imag \* c.imag;

temp.imag = real \* c.imag + imag \* c.real;

return temp;

}

int main()

{

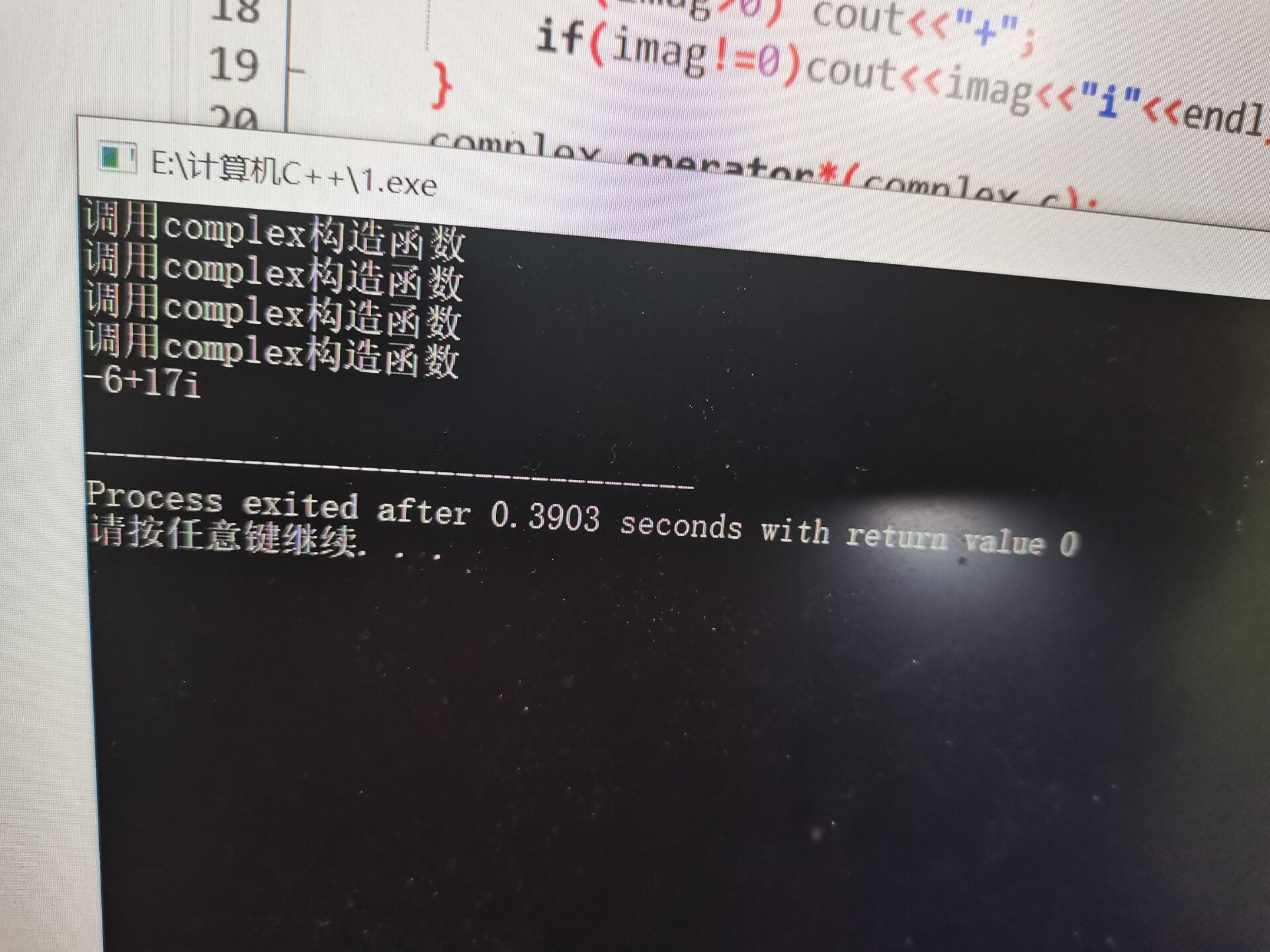
complex co1(2, 3), co2(3, 4), total1(0,0);

total1 = co1 \* co2;

total1.print();

return 0;

}



**二感想心得**

1. 什么时候应用运算符重载函数？

当进行运算的数据的数据类型不是基本数据类型，而是用户自定义数据类型时，需要用到运算符重载。

1. 运算符重载的参数至少有一个是类对象（或类的引用）。
2. 运算符重载函数有：

类外定义的运算符重载函数——缺点是不能直接调用类中的私有和保护数据成员。

友元运算符重载函数

成员运算符重载函数

**Copyright 2021-2099 XiaowenRen. All rights reserved**