**一程序代码**

#include<iostream>

using namespace std;

class Complex {

double real;

double imag;

public:

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

{

real = r; imag = i;

}

void print();

friend Complex operator\*(Complex co1, Complex co2);

};

Complex operator\*(Complex co1, Complex co2)

{

Complex temp;

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

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

return temp;

}

void Complex::print()

{

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

}

int main()

{

float a, b, c, d;

cout << "please input real and imag:" << endl;

cin >> a >> b;

cout << "please input real and imag:" << endl;

cin >> c >> d;

Complex com1(a, b), com2(c, d), total1;

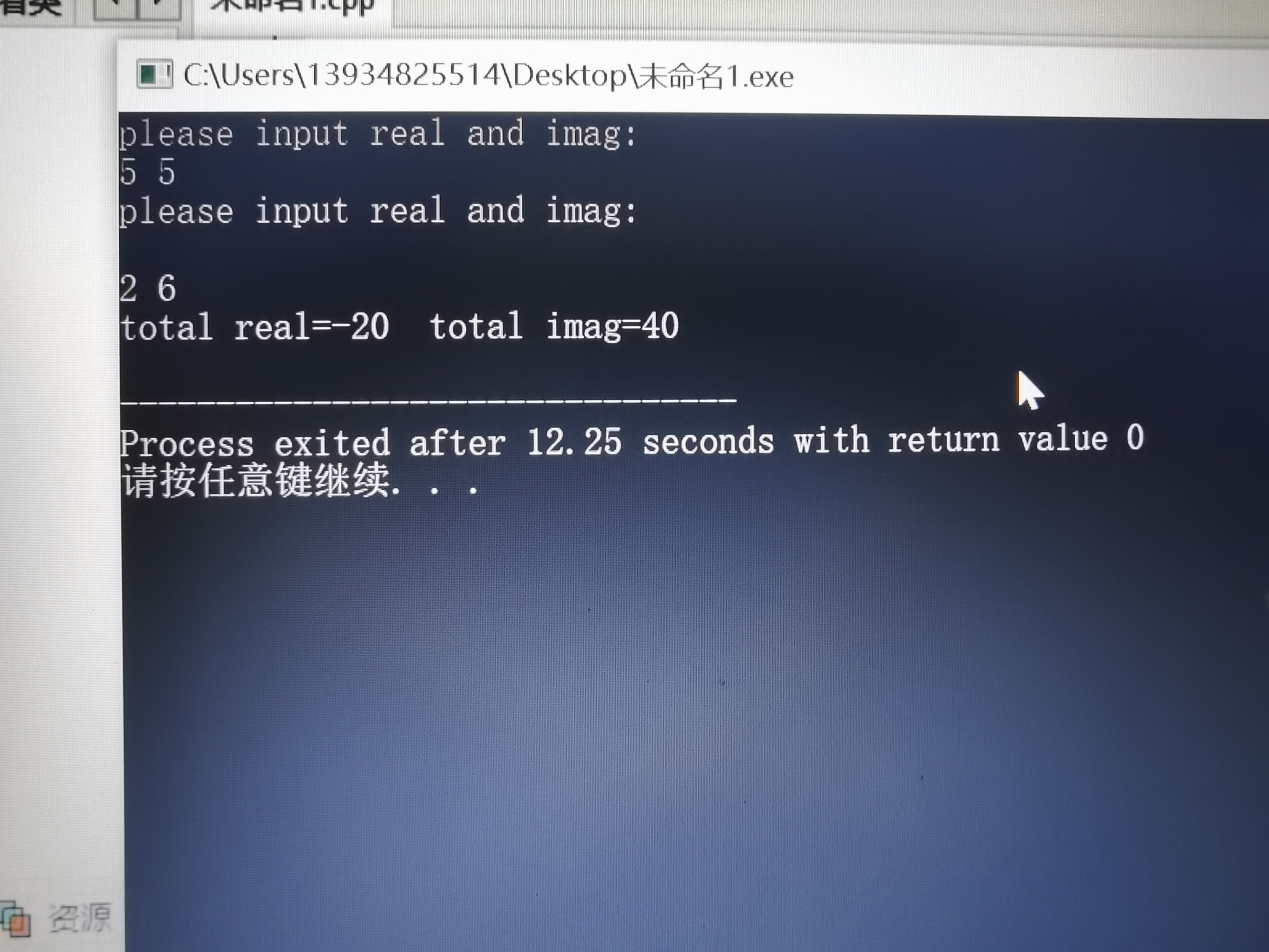
total1 = com1 \* com2;

total1.print();

return 0;

}

**二结果**



**三心得体会**

**1.友元函数既可以是不属于任何类的非成员函数，也可以是另一个类的成员函数，友元函数不是当前类的成员函数，是独立于当前类的外部函数，但是它可以访问该类的所以成员，包括私有成员，公有成员和保护成员**

**2.在类中声明友元函数时，需要加上关键字friend，在类外则不用**

**3.只能对已有的c++运算符进行重载，不能由用户自定义，**

**重载不能改变运算符的操作对象的个数，不能改变优先级，实验中这是我出现的问题**

Copyright ©2021-2099 lvlei. All rights reserved