代码1

#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;

temp.imag = co1.imag + co2.imag;

return temp;

}

void Complex::print()

{

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

}

int main()

{

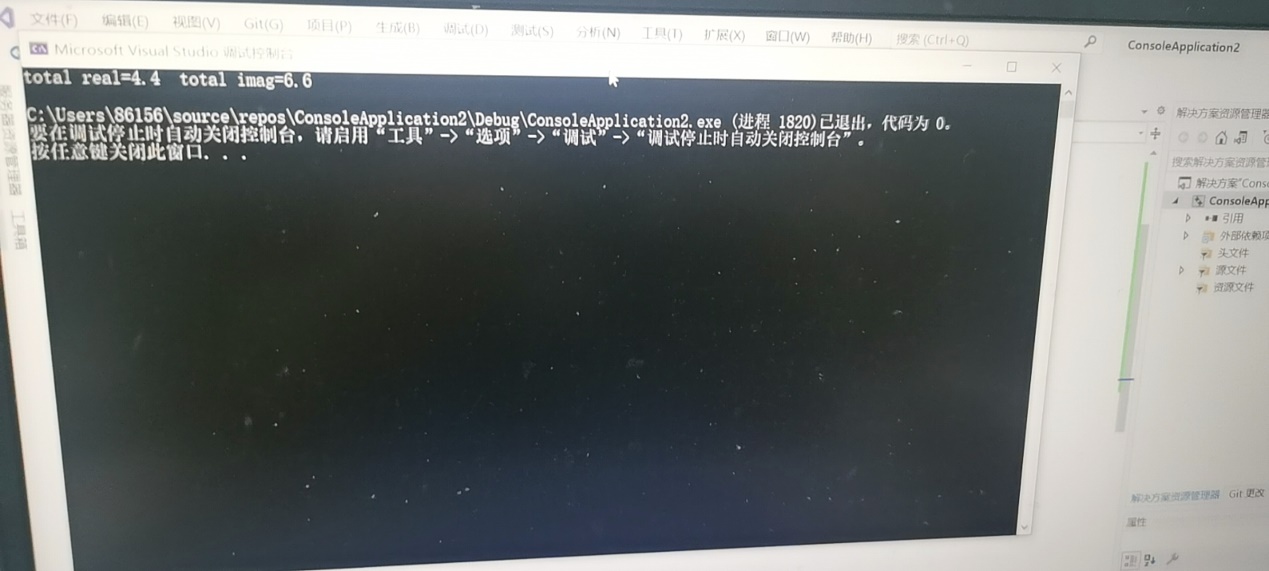
Complex com1(1.1, 2.2), com2(3.3, 4.4), total1;

total1 = com1 + com2;

total1.print();

return 0;

}



代码2

#include <iostream>

using namespace std;

class complex {

public:

complex() {

real = 0.0;

imag = 0.0;

}

complex(double r\_, double i\_) {

real = r\_;

imag = i\_;

}

void print() {

if (imag >= 0)

cout << real << "+" << imag << "i" << endl;

else

cout << real << imag << "i" << endl;

}

complex multiply(const complex& z2) {

complex result;

result.real = real \* z2.real - imag \* z2.imag;

result.imag = real \* z2.imag + imag \* z2.real;

return result;

}

private:

// real and imaginary part

double real, imag;

};

int main() {

complex c1(1, 4), c2(3.3, -4.13);

complex c3 ;

c3 = c1.multiply(c2);

c1.print();

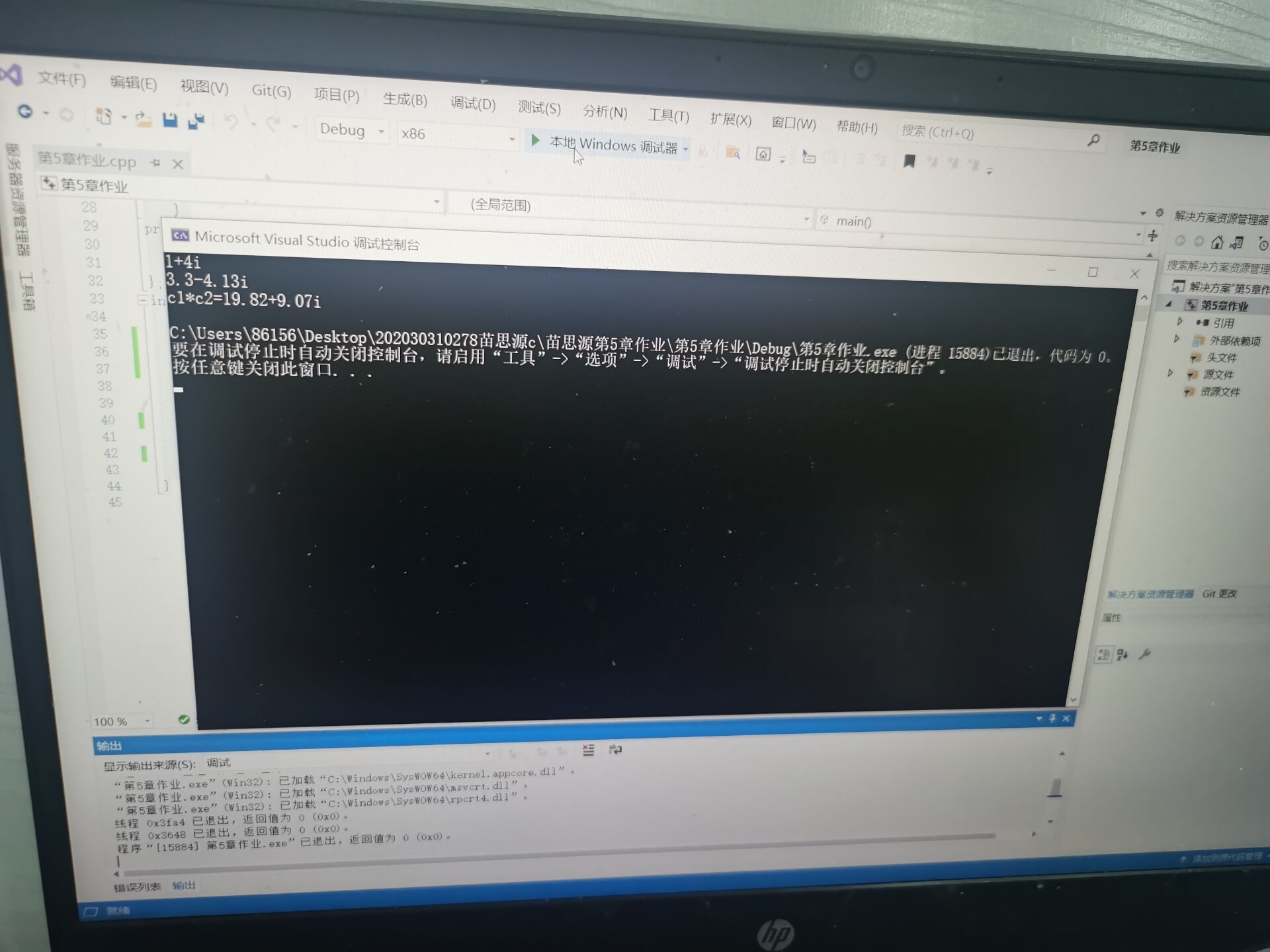
c2.print();

cout << "c1\*c2=";

c3.print();

return 0;

}



感悟

通过本次实验我掌握C++语言多态性的基本概念；

掌握运算符重载函数的声明和定义方法；

我无比的相信，这些知识一定会在不远的将来让我受益匪浅。