上机实验7：尝试编写一个程序，实现两个复数的乘法

#include<iostream>

using namespace std;

class Complex {

public:

double real;

double image;

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

real = r;

image = i;

}

};

Complex operator\*(Complex co1, Complex co2) {

Complex temp;

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

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

return temp;

}

int main() {

Complex com1(1.1, 2.2), com2(3.3, 4.4), total1, total2;

total1 = operator\*(com1, com2)；

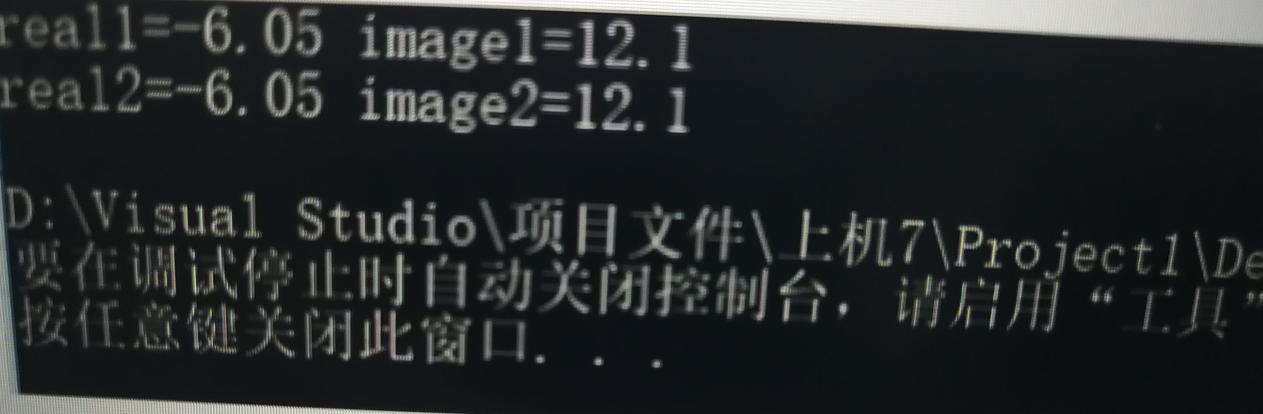
cout << "real1=" << total1.real <<" " << "image1=" << total1.image << endl;

total2 = com1 \* com2;

cout << "real2=" << total2.real << " " << "image2=" << total2.image << endl;

return 0;

}



1. 实验心得总结：当基类构造函数不带参数时，派生类不一定需要定义构造函数；然而当基类的构造函数哪怕只带有一个参数，它所有的派生类都必须定义构造函数，甚至所定义的派生类构造函数的函数体可能为空，它仅仅起参数的传递作用。