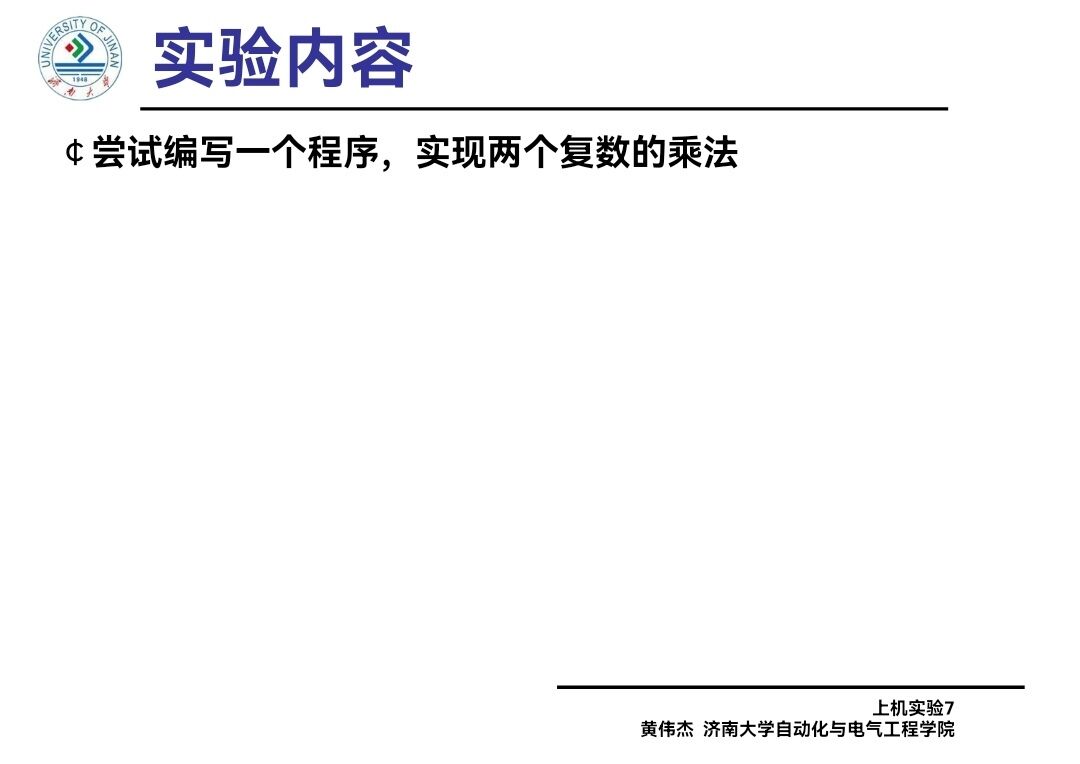
# 上机实验7



# 程序代码

#include<iostream>

using namespace std;

class complex

{

public:

double real;

double imag;

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

{

real = r; imag = i;

}

};

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;

}

int main()

{

complex com1(1, 2), com2(3, 4), s;

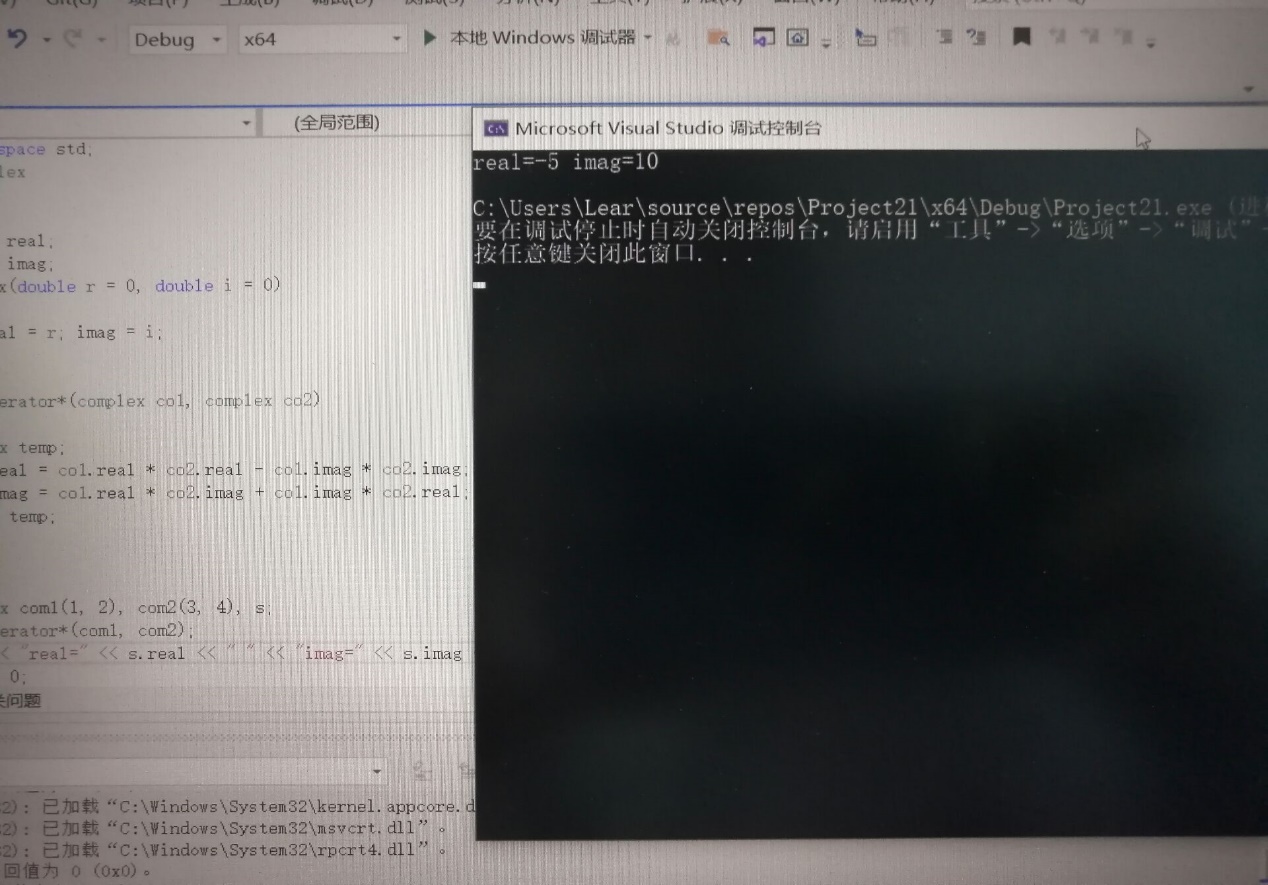
s = operator\*(com1, com2);

cout << "real=" << s.real << " " << "imag=" << s.imag << endl;

return 0;

}

# 程序结果



# 感想心得

实验过程中值得注意的是区分前置和后置运算符的区别:前置运算符先运算后返回;后置运算符先返回后运算。这点很重要,由于之没有注意,所以做出来的值不对,后来改正之后就对了。通过这次实验,我基本掌握了C++语言多态性的基本概念，通过运算符重载实现多态性的方法,学会了运算符重载的成员函数法和友元函数法,基本能够区分单目运算符的前置与后置。

Copyright 2021-2099 Hubing. All rights reserved