#include <iostream>  
  
using namespace std;  
  
class matrixmul  
{  
private:  
    double real;  
    double imag;  
public:  
    void output();//打印输出  
    void initSet(double re,double im);//初始化  
    matrixmul matrixMul(matrixmul Z1,matrixmul Z2);//函数返回值是multirxmul，所以此函数类型为matrixmal  
};  
  
matrixmul matrixmul::matrixMul(matrixmul Z1,matrixmul Z2)  
{  
    double temp1,temp2,temp3;  
    matrixmul result;  
    temp1=Z1.real\*Z2.imag;  
    temp2=Z1.imag\*Z2.real;  
    temp3=(Z1.imag+Z1.real)\*(Z2.real-Z2.imag);  
    result.real=temp1+temp3-temp2;  
    result.imag=temp1+temp2;  
  
    return result;  
}  
  
void matrixmul::initSet(double re,double im)  
{  
    real=re;  
    imag=im;  
}  
  
void matrixmul::output()  
{  
    if(imag>0)  
    {  
        cout<<real<<"+"<<imag<<"i"<<endl;  
    }  
    else if(imag==0)  
    {  
        cout<<real<<endl;  
    }  
    else if(imag<0)  
    {  
        cout<<real<<imag<<"i"<<endl;  
    }  
}  
  
int main()  
{  
    matrixmul Z1,Z2,Z3,result;  
    Z1.initSet(1,-2);  
    Z2.initSet(3,-4);  
    cout<<"第一个复数Z1为："<<endl;  
    Z1.output();  
    cout<<"第二个复数Z2为："<<endl;  
    Z2.output();  
    result=Z3.matrixMul(Z1,Z2);  
    cout<<"两复数相乘结果的复数为："<<endl;  
    result.output();  
    return 0;  
}

心得体会

C++是计算机程序设计的重要理论基础，在我们以后的学习和工作中都有着十分重要的地位。要学好这种语言，仅仅学习课本上的知识是不够的，还要经常自己动手，有较强的实践能力。  
只有多动手，经常编写程序，才能发现我们学习上的漏洞和自己的不足，并在实践中解决这些问题，不断提高自己转化知识的能力。