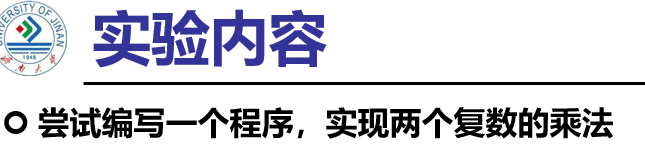
**第七次实验报告**

实验要求：



心得体会：

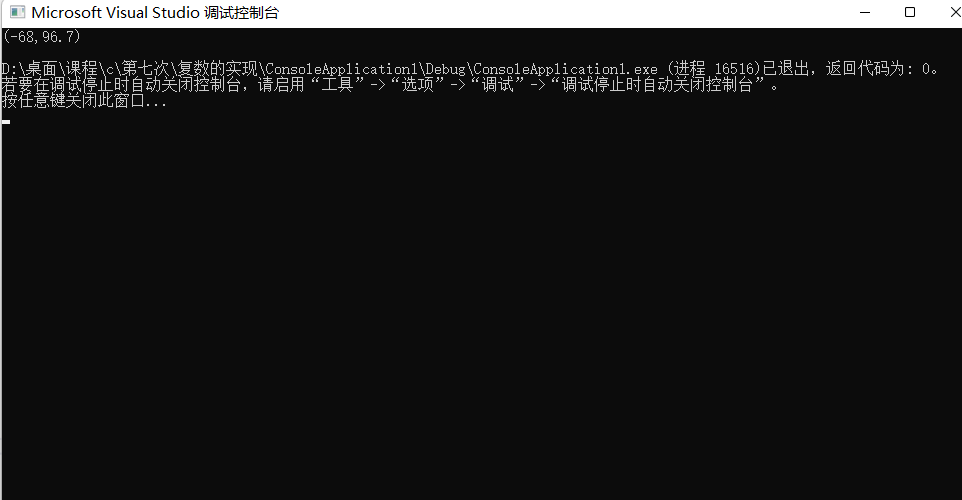
这个我之前编过了，课上就复习了复习，因为是PPT上的内容，这次实验的重点是运算符重载，来实现复数的相加；正常情况我们只能做到对已有的数据类型通过C++本来的运算符进行运算，而不能对用户自定义的数据类型进行重载，而运算符重载则提供了这个方法。这次实验中，因为对于双目运算符而言，成员运算符重载函数的形参表中仅有一个参数，而另一个操作数（左操作数）是隐含的，通过this指针指向的。实际编写过程中有些不适应。我参考课本211页有关复数的代码编写了下面的程序。

实验代码与程序运行结果

|  |
| --- |
| //版权声明  //Copyright 2021-2099 ZhantengZhang.  //202030310284 张展腾  #include <iostream>  using namespace std;  class complex {  private:  float real;  float imag;  public:  complex operator\*(complex p)  {  complex temp;  temp.real = real \* p.real - imag \* p.imag;  temp.imag = real \* p.imag + imag \* p.real;  return temp;  }  complex(float a = 0, float b = 0)  {  real = a;  imag = b;  }  void show()  {  cout << "(" << real << "," << imag << ")"<<endl;  }    };  int main()  {  complex a(4, 6), b(4, -6), c;  c = a \* b;  c.show();  } |

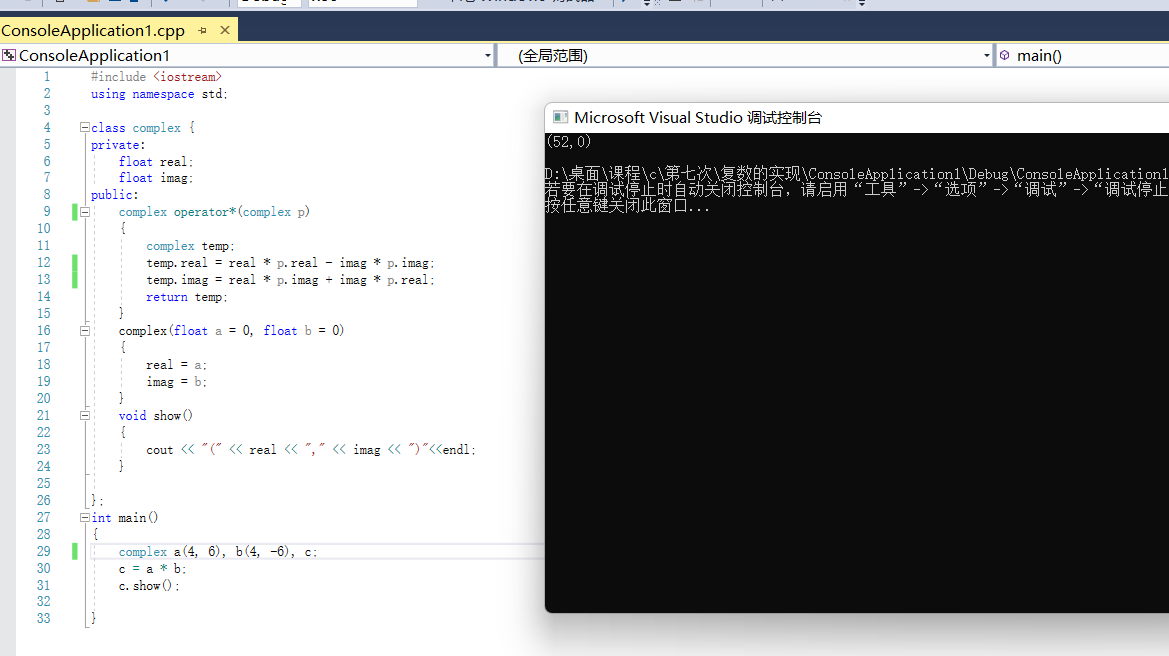
关于复数相关运算和重载符相关运算我在课本211上参考了部分代码。

运行结果如下



将其他的数带入再尝试

运算结果仍正确



版权声明：

Copyright 2021-2099 ZhantengZhang.

202030310284 张展腾