学号：1651574 姓名：贾昊霖 班级： 1班

**以 P.324 - P.325 例10.10为基础并进行适当扩展，讨论转换构造函数及类型转换函数的使用，完成下列文档**

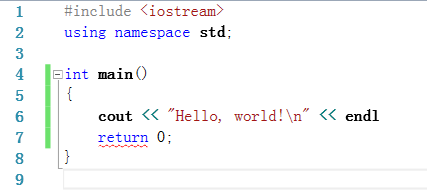
**【填写方法：】**

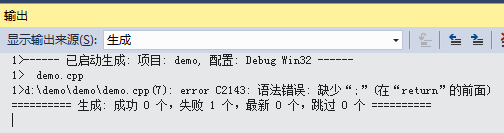
1. 依次用VS2017/Linux编译指定的源程序文件

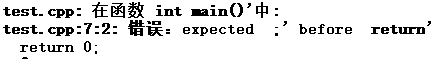
如果编译正确，则对应位置填写运行结果并给出得到此结果的原因解释

如果编译错误，则对应位置填写该行的编译错误提示及错误原因分析

1. 如果编译器报多个错误，填写源程序文件对应行的错误提示即可，示例如下，将红色框截图即可：







1. 如果main函数中某一句错误，则将该句及下面的打印语句全部注释掉，继续观察其余正确语句的运行结果（示例如下）

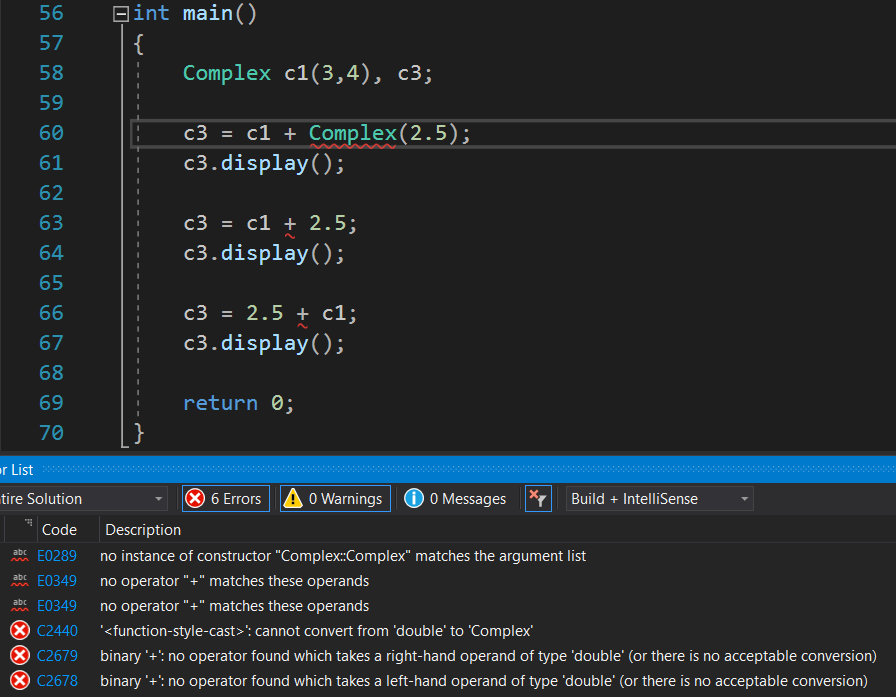
 

1. **用蓝色加粗字体填写**
2. 不需要填写的部分可以删除（例如：某句正确，则错误部分不填，可以删除）

**【10-b1-1.cpp ：】**无转换构造函数、无类型转换函数、友元方式实现复数+

* c3 = c1 + Complex(2.5)

编译错误，VS2017下：



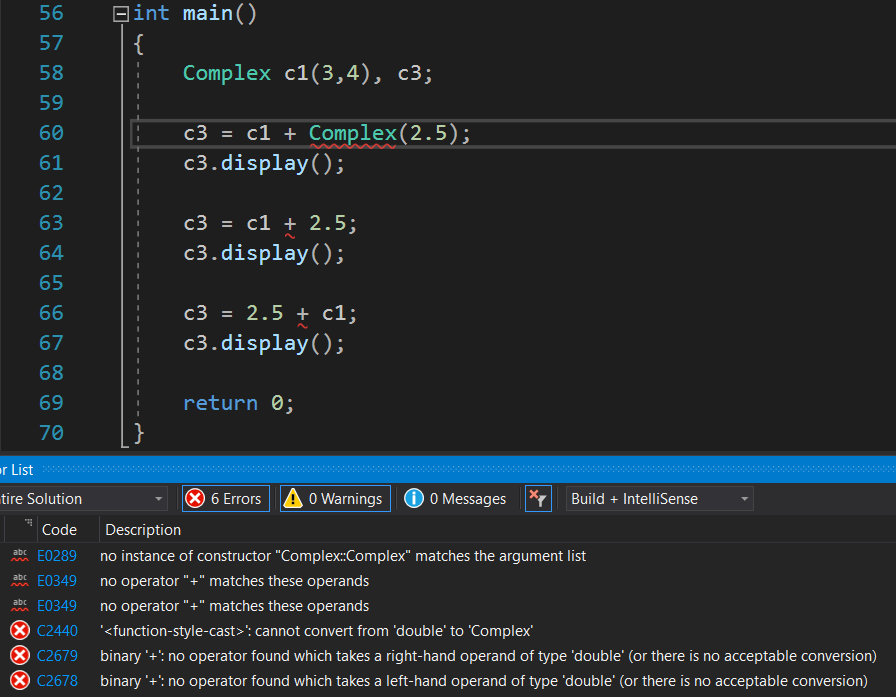
Linux下：



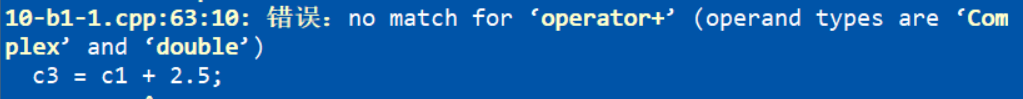
错误原因分析：\_\_\_\_\_\_\_没有写转换构造函数\_\_\_\_\_\_\_\_\_\_\_\_

* c3 = c1 + 2.5

编译错误，VS2017下：



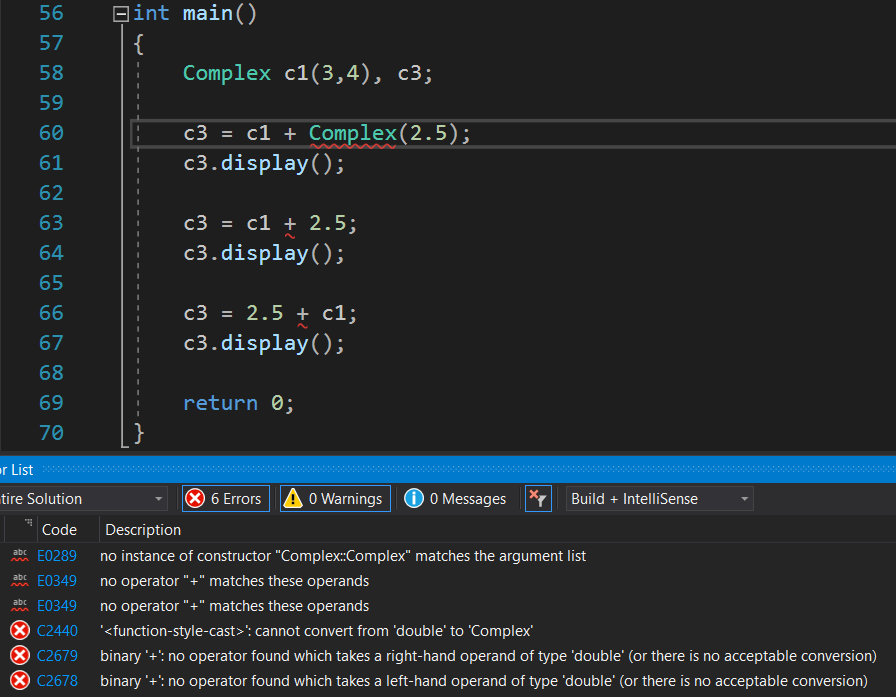
Linux下：



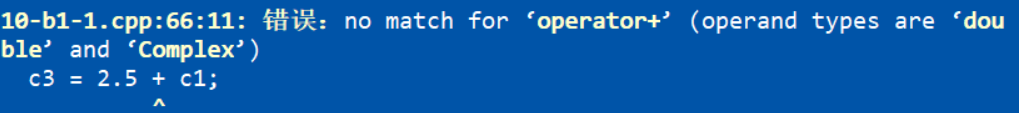
错误原因分析：\_\_\_\_\_\_\_\_\_\_\_没有写转换构造函数\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

* c3 = 2.5 + c1

编译错误，VS2017下：



Linux下：

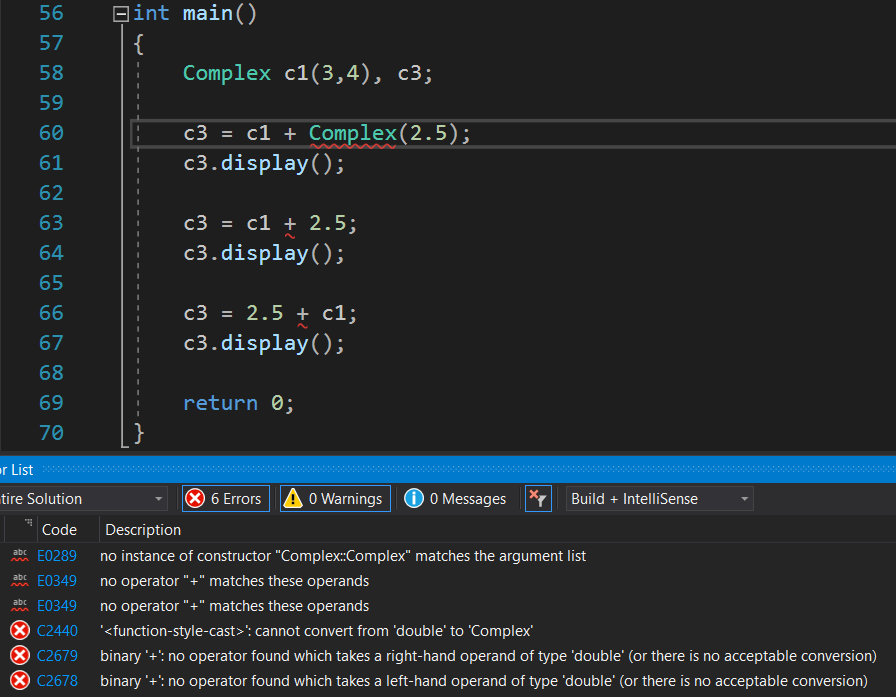


错误原因分析：\_\_\_\_\_\_\_\_\_\_没有写转换构造函数\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

**【10-b1-2.cpp ：】**无转换构造函数、无类型转换函数、成员方式实现复数+

* c3 = c1 + Complex(2.5)

编译错误，VS2017下：



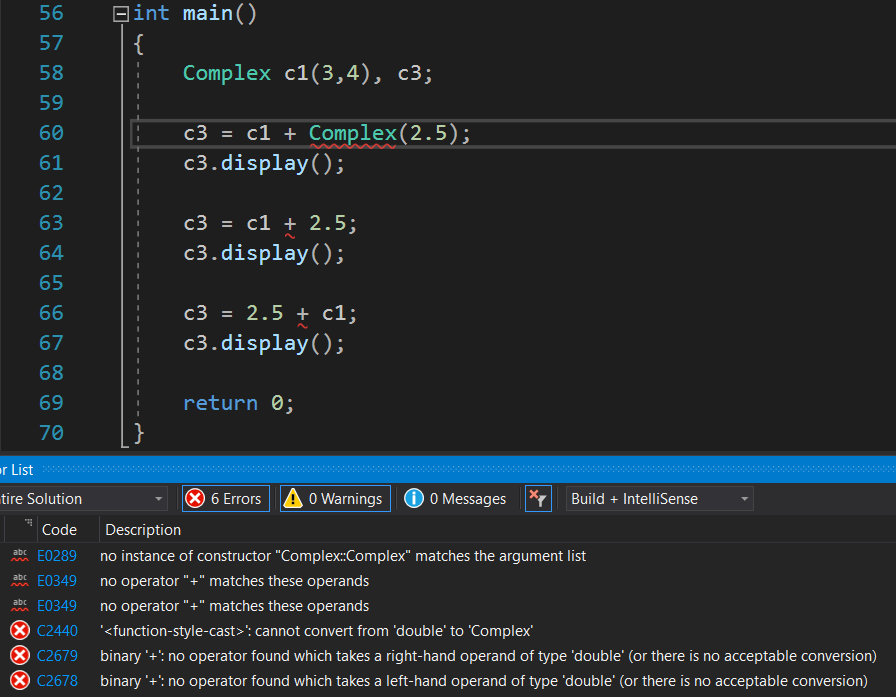
Linux下：



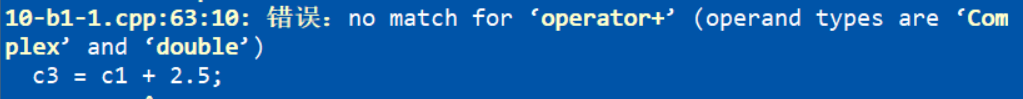
错误原因分析：\_\_\_\_\_\_\_没有写转换构造函数\_\_\_\_\_\_\_\_\_\_\_\_

* c3 = c1 + 2.5

编译错误，VS2017下：



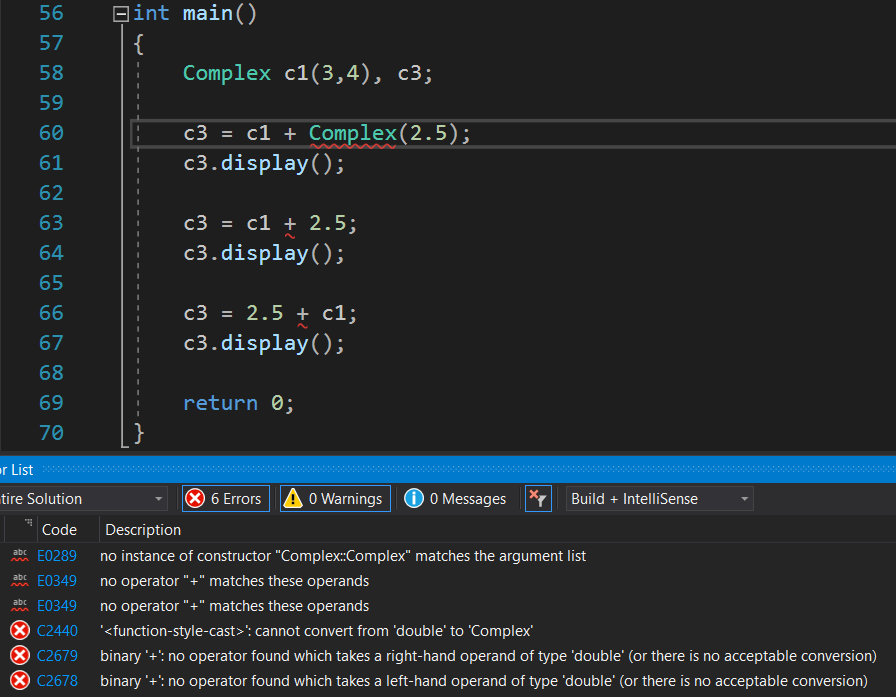
Linux下：



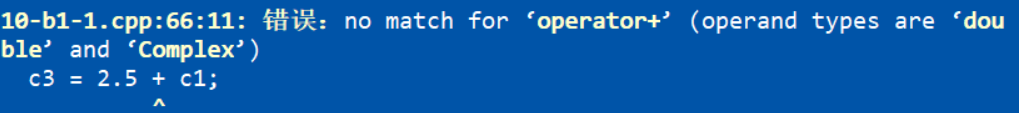
错误原因分析：\_\_\_\_\_\_\_\_\_\_\_没有写转换构造函数\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

* c3 = 2.5 + c1

编译错误，VS2017下：



Linux下：

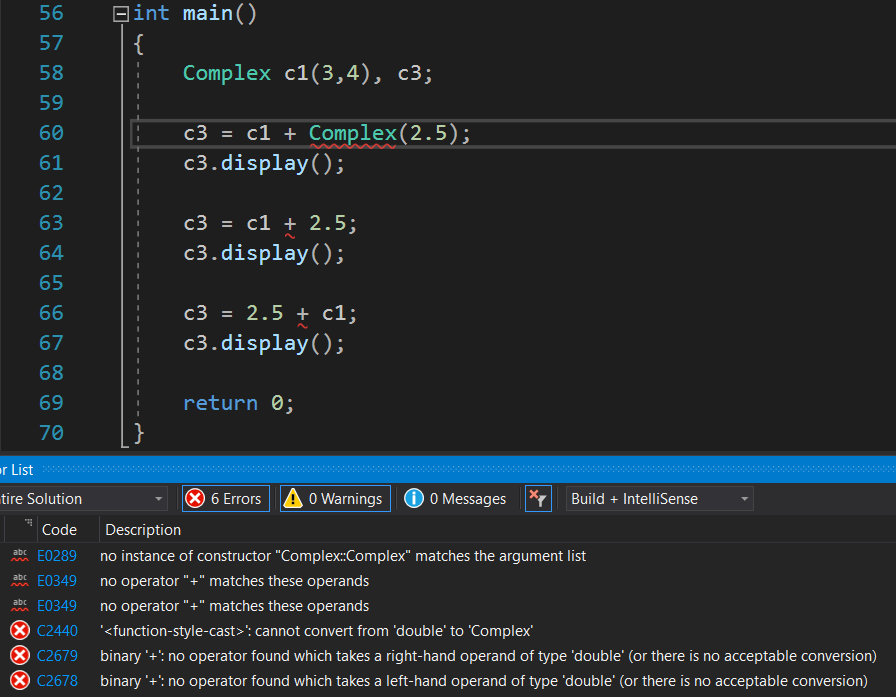


错误原因分析：\_\_\_\_\_\_\_\_\_\_没有写转换构造函数\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

**【10-b1-3.cpp ：】**无转换构造函数、有类型转换函数、友元方式实现复数+

* c3 = c1 + Complex(2.5)

编译错误，VS2017下：



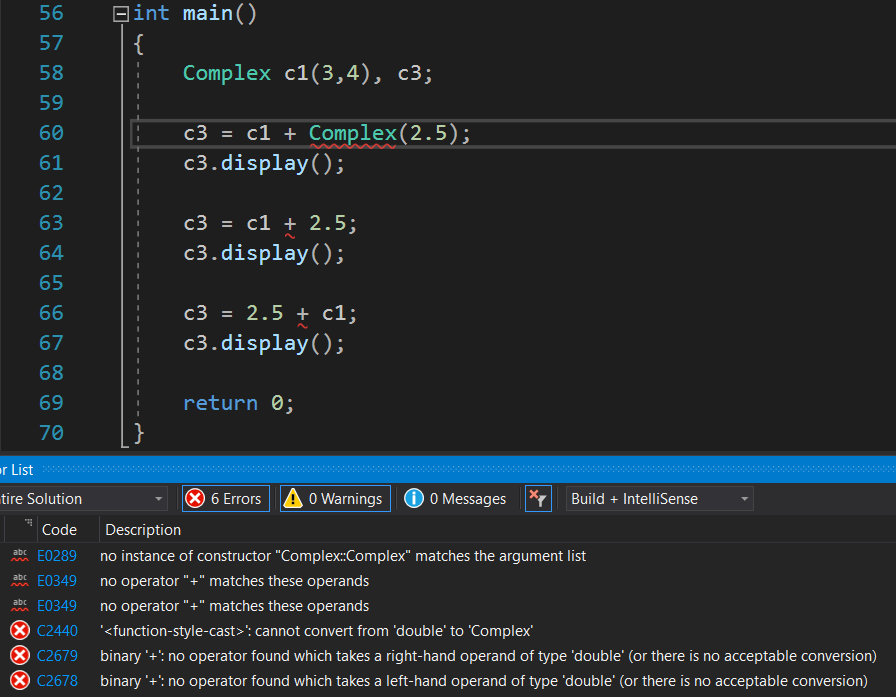
Linux下：



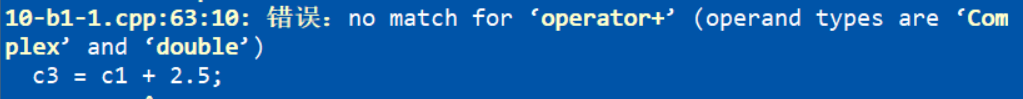
错误原因分析：\_\_\_\_\_\_\_没有写转换构造函数\_\_\_\_\_\_\_\_\_\_\_\_

* c3 = c1 + 2.5

编译错误，VS2017下：



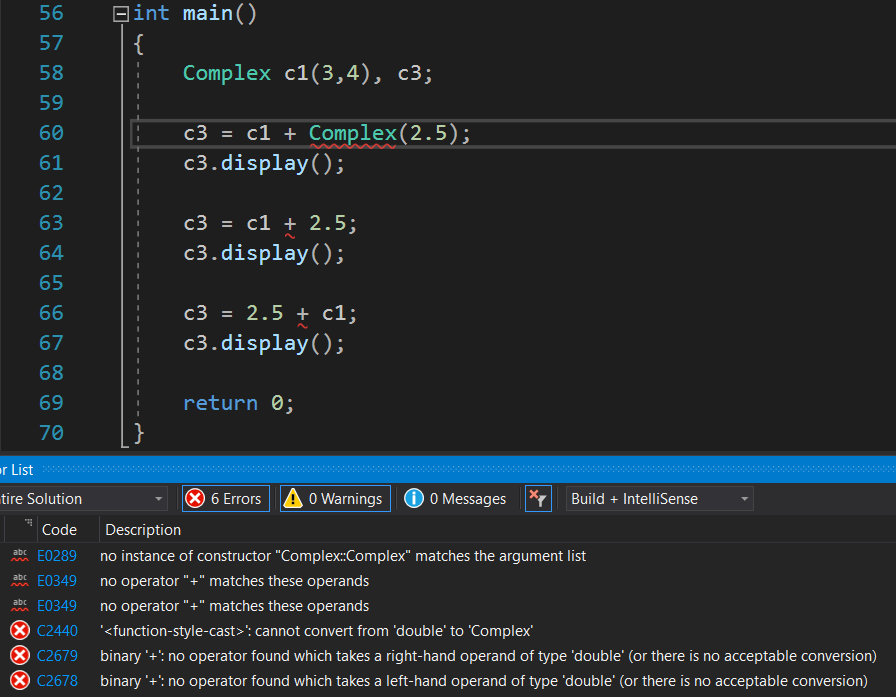
Linux下：



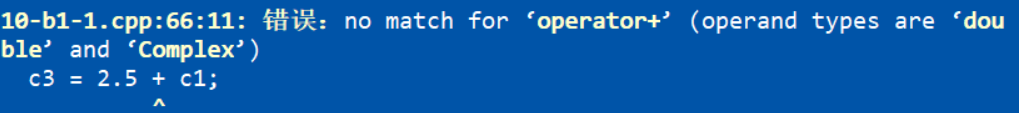
错误原因分析：\_\_\_\_\_\_\_\_\_\_\_没有写转换构造函数\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

* c3 = 2.5 + c1

编译错误，VS2017下：



Linux下：

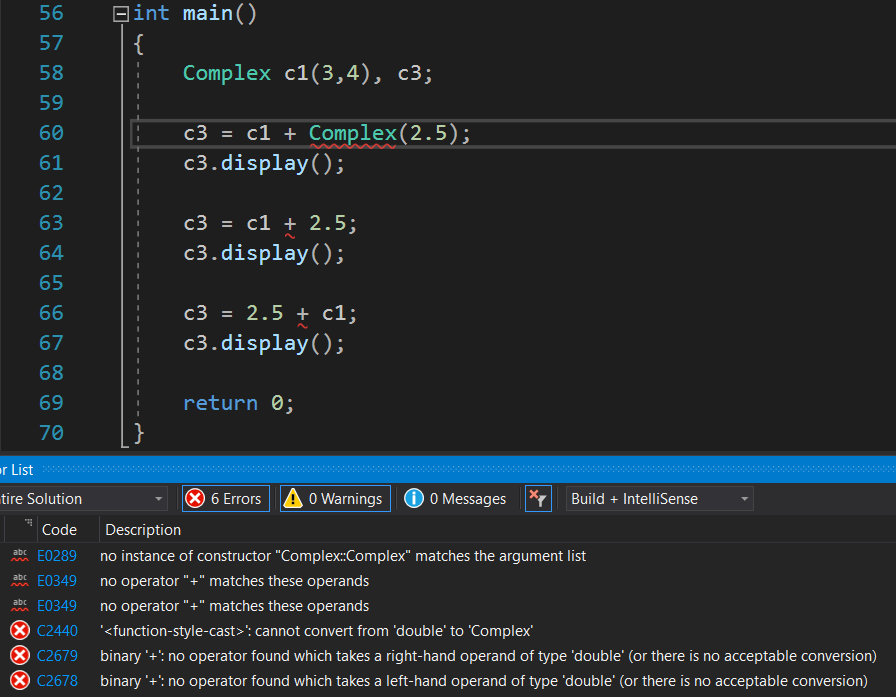


错误原因分析：\_\_\_\_\_\_\_\_\_\_没有写转换构造函数\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

**【10-b1-4.cpp ：】**无转换构造函数、有类型转换函数、成员方式实现复数+

* c3 = c1 + Complex(2.5)

编译错误，VS2017下：



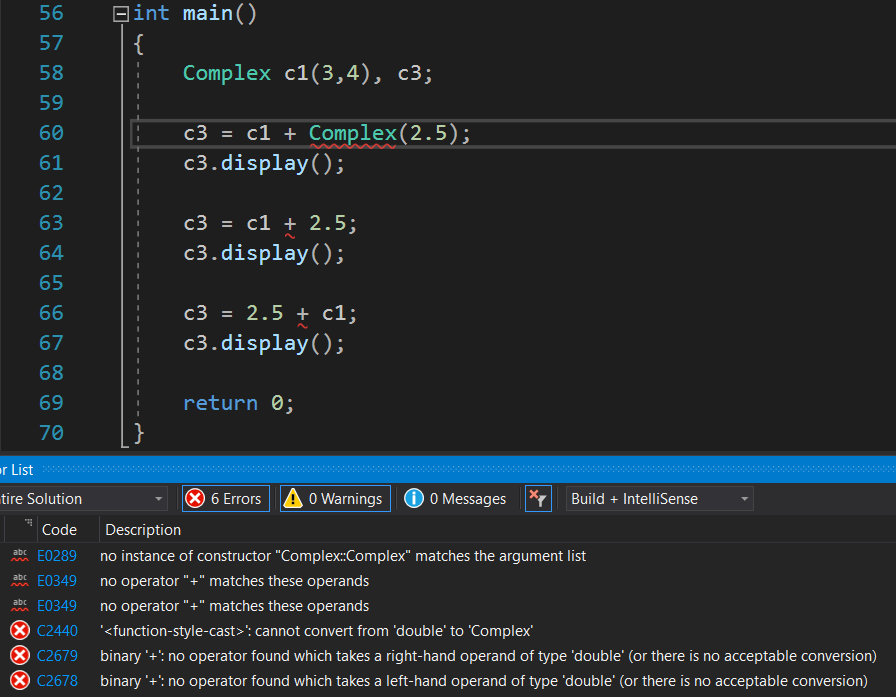
Linux下：



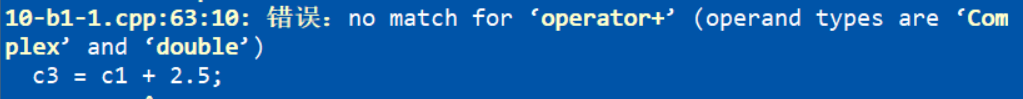
错误原因分析：\_\_\_\_\_\_\_没有写转换构造函数\_\_\_\_\_\_\_\_\_\_\_\_

* c3 = c1 + 2.5

编译错误，VS2017下：



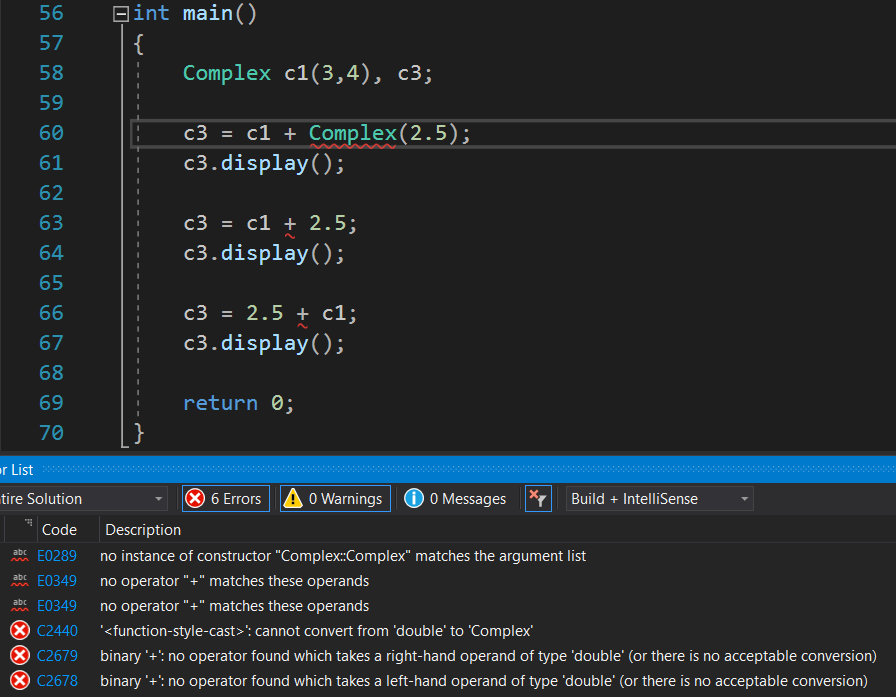
Linux下：



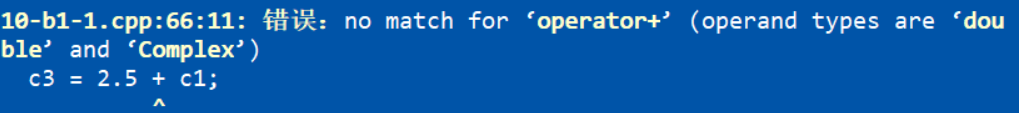
错误原因分析：\_\_\_\_\_\_\_\_\_\_\_没有写转换构造函数\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

* c3 = 2.5 + c1

编译错误，VS2017下：



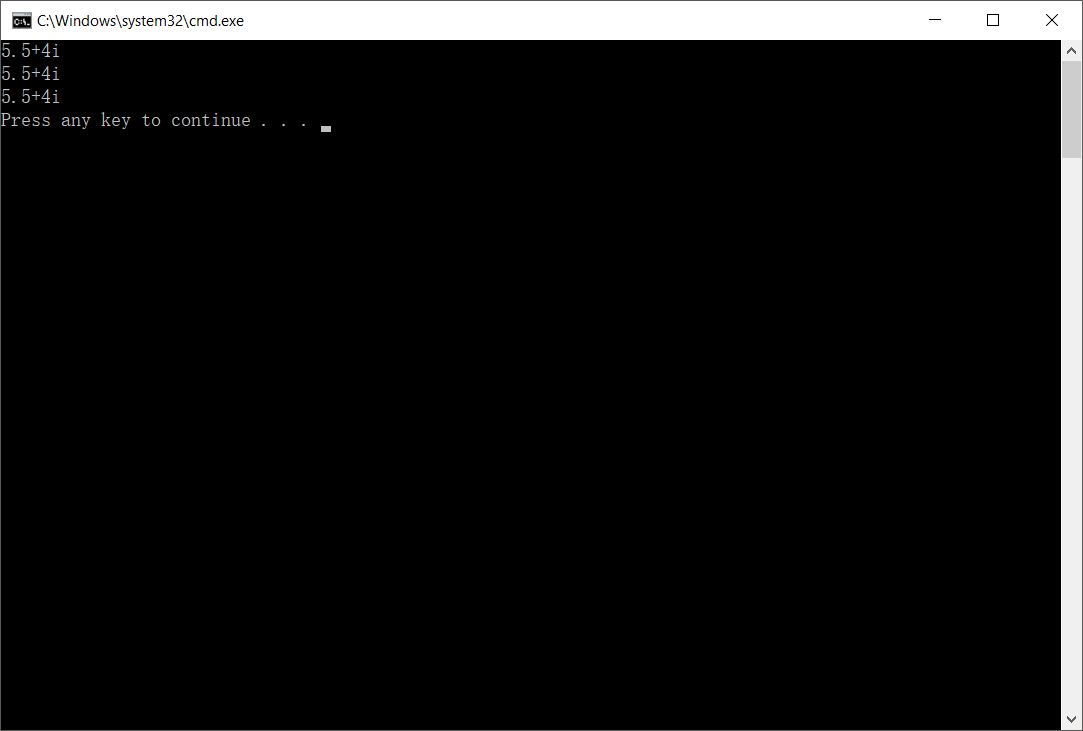
Linux下：



错误原因分析：\_\_\_\_\_\_\_\_\_\_没有写转换构造函数\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

**【10-b1-5.cpp ：】**有转换构造函数、无类型转换函数、友元方式实现复数+

编译正确，运行结果：

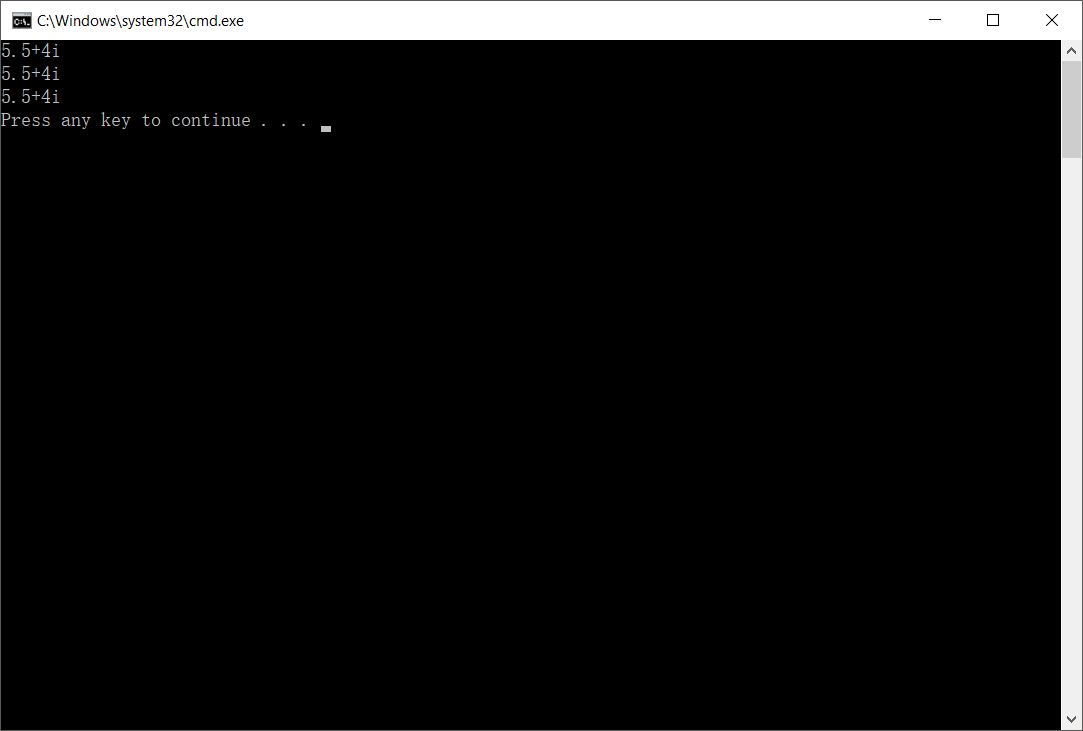
，

能得到此结果的原因：\_\_\_\_2.5强制转换为Complex，Complex+Complex得到Complex,复制给左值\_\_\_\_\_\_

**【10-b1-6.cpp ：】**有转换构造函数、无类型转换函数、成员方式实现复数+

* c3 = c1 + Complex(2.5)

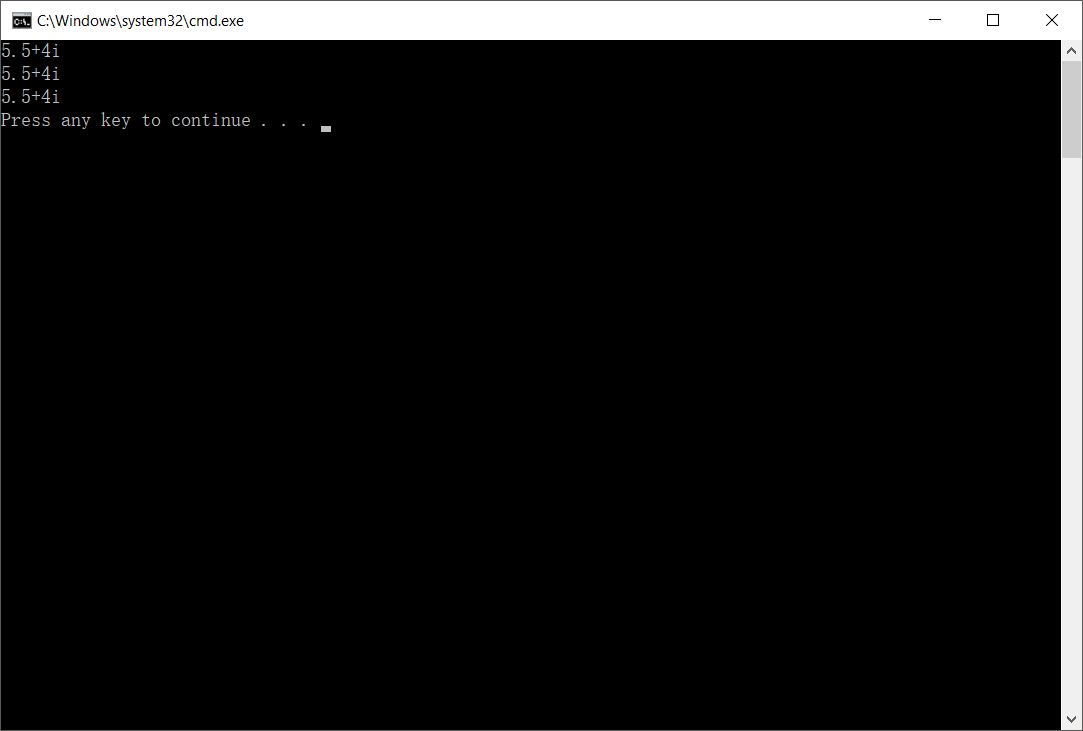
编译正确，运行结果：

，

能得到此结果的原因：\_\_\_2.5强制转换为Complex，Complex+Complex得到Complex,复制给左值

* c3 = c1 + 2.5

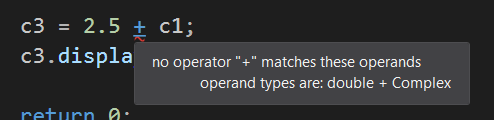
编译正确，运行结果：

，

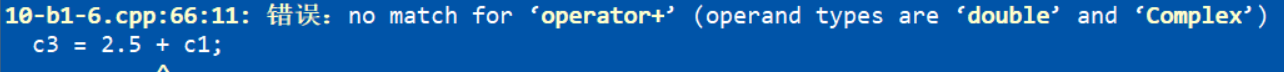
能得到此结果的原因：\_\_2.5强制转换为Complex，Complex+Complex得到Complex,复制给左值

* c3 = 2.5 + c1

编译错误，VS2017下：



Linux下：

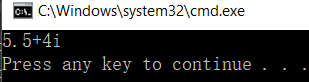


错误原因分析：\_\_\_+是Complex类的成员函数，而2.5不是Complex类，不能调用这个函数 \_\_\_\_\_

**【10-b1-7.cpp ：】**有转换构造函数、有类型转换函数、友元方式实现复数+

* c3 = c1 + Complex(2.5)

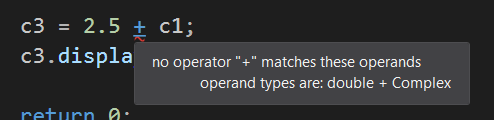
编译正确，运行结果：



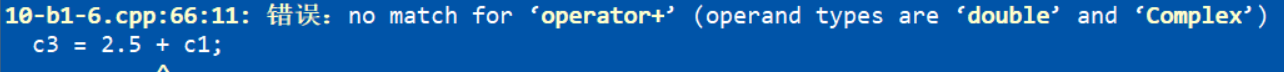
，能得到此结果的原因：\_\_2.5强制转换为Complex，Complex+Complex得到Complex,复制给左值\_

* c3 = c1 + 2.5

编译错误，VS2017下：



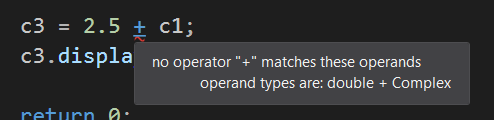
Linux下：



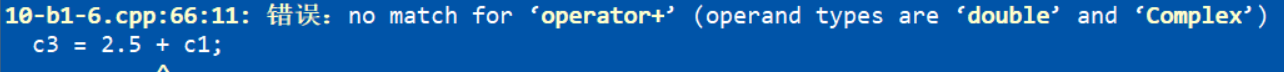
错误原因分析：\_\_+是Complex类的成员函数，而2.5不是Complex类，不能调用这个函数\_\_\_

* c3 = 2.5 + c1

编译错误，VS2017下：



Linux下：

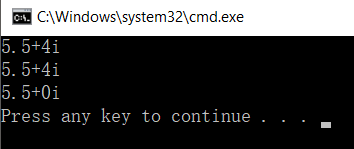


错误原因分析：\_\_+是Complex类的成员函数，而2.5不是Complex类，不能调用这个函数\_

**【10-b1-8.cpp ：】**有转换构造函数、有类型转换函数、成员方式实现复数+

* c3 = c1 + Complex(2.5)

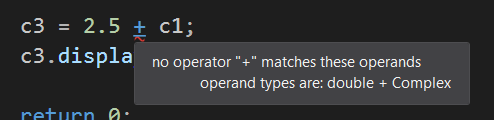
编译正确，运行结果：

，

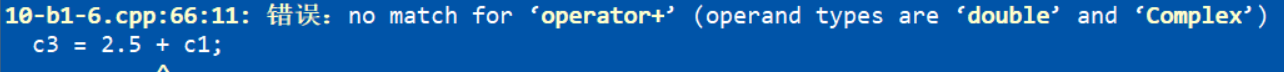
能得到此结果的原因：\_2.5强制转换为Complex，Complex+Complex得到Complex,复制给左值\_

* c3 = c1 + 2.5

编译错误，VS2017下：



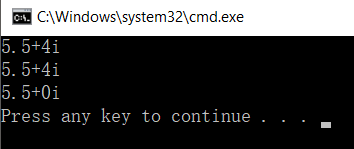
Linux下：



错误原因分析：+是Complex类的成员函数，而2.5不是Complex类，不能调用这个函数\_\_

* c3 = 2.5 + c1

编译正确，运行结果：

 ，

能得到此结果的原因：\_\_2.5强制转换为Complex，Complex+Complex得到Complex,复制给左值\_

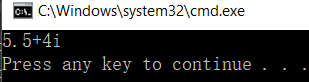
**【10-b1-9.cpp ：】**有转换构造函数、有类型转换函数、成员方式实现复数+、另有double+Complex

* 仅讨论语句c3 = 2.5 + c1，回答下列问题

1. 为什么编译不错？

没有二义性，在第一次寻找严格匹配的时候就找到对应double+Complex的函数，不再继续向下匹配。

1. 运行结果是多少？



1. 为什么和10-b1-8结果不同？

10-b1-8是通过内部转换匹配的，匹配到了double+double，Complex先强制转换为double

**【10-b1-10.cpp ：】**单独讨论有类型转换的情况下，cout重载的输出结果与期望值不同

* 目前main函数中第4个输出语句与期望值不同，原因是：\_\_常对象不能调用普通成员函数\_
* 仅允许改动**两行**，使程序输出与期望值相同：

改动第\_\_\_\_\_32\_\_\_\_\_行，原内容：\_friend ostream & operator<<(ostream & stream, Complex &c);

新内容：\_ friend ostream & operator<<(ostream & stream, const Complex &c);

改动第\_\_\_\_35\_\_\_\_\_\_行，原内容：\_ostream &operator<<(ostream &out, Complex &c) \_\_

新内容：\_ ostream &operator<<(ostream &out, const Complex &c)\_