

§ 14. 运算符重载 - 转换构造和类型转换函数的使用



要求:

- 1、完成本文档中所有的题目并写出分析、运行结果
- 2、无特殊说明，均使用 **VS2019 + Linux** 编译
- 3、直接在本文件上作答，**写出答案/截图（不允许手写、不允许手写拍照）**即可；填写答案时，为适应所填内容或贴图，**允许调整**页面的字体大小、颜色、文本框的位置等
★ 在保证一页一题的前提下，具体页面布局可以自行发挥，简单易读即可
- 4、转换为pdf后提交
- 5、**4月22日前**网上提交本次作业（在“实验报告”中提交）



§ 14. 运算符重载 - 转换构造和类型转换函数的使用

填写要求:

1、依次用VS2019/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

2、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  #include <iostream>
2  using namespace std;
3
4  int main()
5  {
6      cout << "Hello, world!\n" << endl;
7      return 0;
8  }
9
```

输出

显示输出来源(S): 生成

1>----- 已启动生成: 项目: demo, 配置: Debug Win32 -----

1> demo.cpp

1>d:\demo\demo\demo.cpp (7): error C2143: 语法错误: 缺少“;” (在“return”的前面)

===== 生成: 成功 0 个, 失败 1 个, 最新 0 个, 跳过 0 个 =====

test.cpp: 在函数 int main() 中:

test.cpp:7:2: 错误: expected ';' before 'return'

return 0;

```
c3 = c1 + Complex(2.5); //假设此句错误
c3.display();
```

```
// c3 = c1 + Complex(2.5);
// c3.display();
```

3、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 (示例见上图)

4、用蓝色加粗字体填写

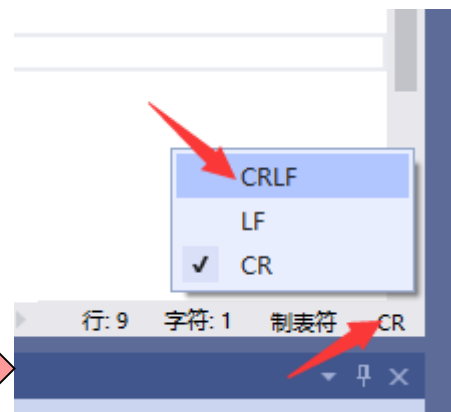
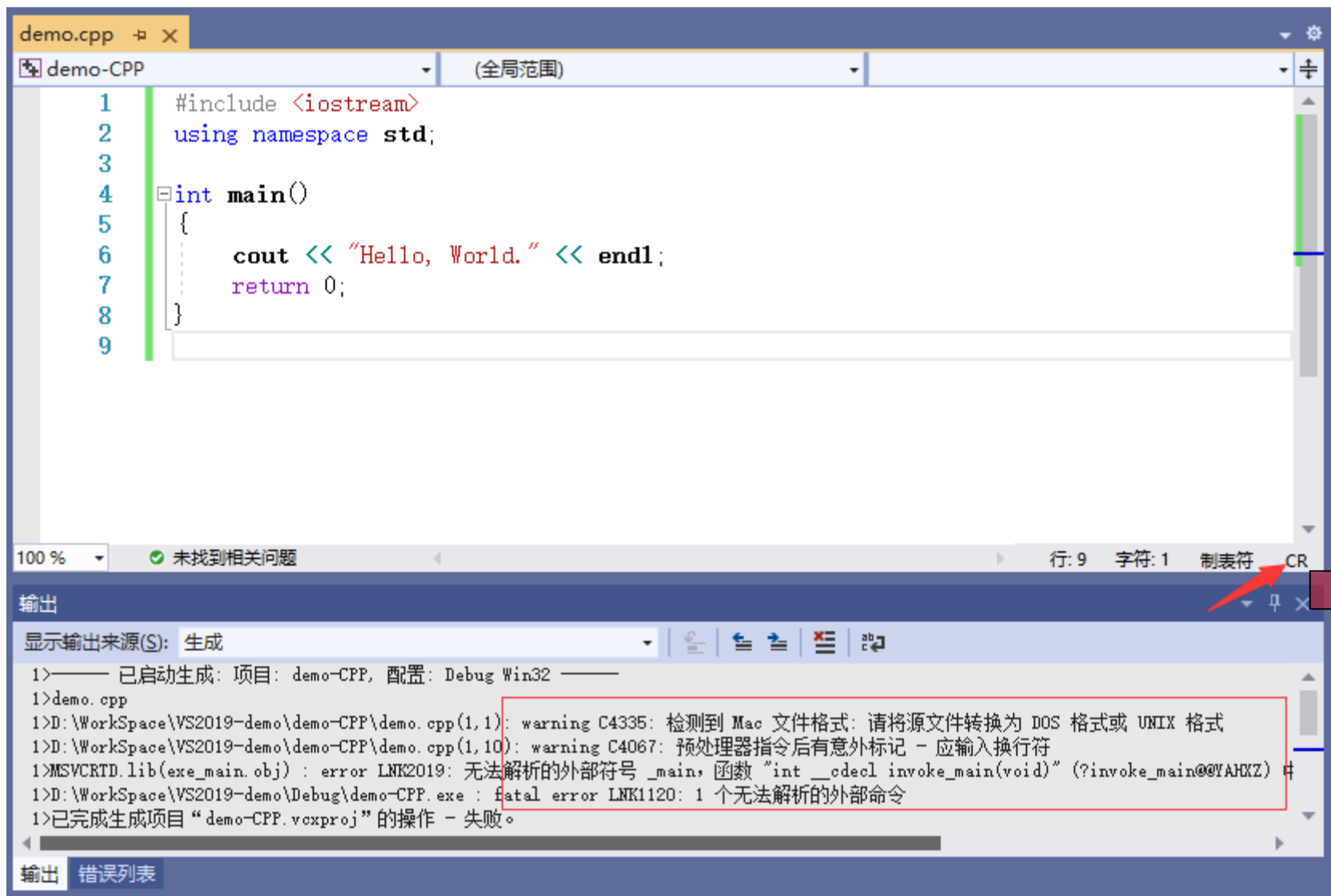
5、不需要填写的部分可以删除 (例如: 某句正确, 则错误部分不填, 或填写/即可)



§ 14. 运算符重载 - 转换构造和类型转换函数的使用

附：用WPS等其他第三方软件打开PPT，将代码复制到VS2019中后，如果出现类似下面的**编译报错**，则观察源程序编辑窗

的右下角是否为CR，如果是，单击CR，在弹出中选择CRLF，再次CTRL+F5运行即可





§ 14. 运算符重载 - 转换构造和类型转换函数的使用

1. 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)

编译正确, 运行结果: _____

编译错误, VS2019下: _____ 0: “<function-style-cast>”: 无法从“double”转换为“Complex”

Linux下: _____ 无构造函数可以接受源类型, 或构造函数重载决策不明确

错误原因分析: _____ 2.5无法转成类对象, 没有一参构造函数, 无法调用友元函数实现‘+’的重载_____

c3 = c1 + 2.5

编译正确, 运行结 _____

编译错误, VS2019 _____ 二进制“+”: 没有找到接受“double”类型的右操作数的运算符(或没有可接受的转换)

Linux下: _____ \可删除横线后贴图/

错误原因分析: _____ 2.5无法转成类对象, 无法调用友元函数实现‘+’的重载, 重载运算符的函数是对象加对象_____

c3 = 2.5 + c1

编译正确, 运行结 _____

编译错误 _____ 二进制“+”: 没有找到接受“double”类型的左操作数的运算符(或没有可接受的转换)

Linux下: _____ \可删除横线后贴图/

错误原因分析: _____ 2.5无法转成类对象, 无法调用友元函数实现‘+’的重载,

重载运算符的函数是对象加对象_____

- 1、无 **double+复数** 的重载
- 2、无 **复数转double** 的类型转换函数, 也无法理解为 double+
- 3、无 **double转复数** 的转换构造函数及 **复数+复数** 的重载, 也无法理解为 复数+

★ 每个输出的不匹配项可删除 (例: 若本项编译正确, 则编译错误的几行内容直接删除即可)

★ 分析正确/错误原因时, 仿课件P. 45~46的样式, 需要将正确/错误原因交待清楚

★ 后续页面要求相同

2.5 + c1,
因为没有定义**double+复数**的重载,
因此c1被转换为double
(隐式调用类型转换函数),
再double相加, 得5.5



§ 14. 运算符重载 - 转换构造和类型转换函数的使用

2. 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)

编译正确, 运行结果

能得到此结果的原因

编译错误, V " <function-style-cast> ": 无法从"double"转换为"Complex"
L

错误原因分析: 2.5无法转成类对象, 没有一参构造函数, 无法调用成员函数实现'+'的重载

c3 = c1 + 2.5

编译正确, 运行结果: , 能得到此结果的原因:

编译错误

二进制"+": 没有找到接受"double"类型的右操作数的运算符(或没有可接受的转换)

错误原因分析: 2.5无法转成类对象, 无法调用成员函数实现'+'的重载, 重载运算符的函数是对象加对象

c3 = 2.5 + c1

编译正确, 运行结果

编译错误, VS2019

Linux下

没有与这些操作数匹配的 "+" 运算符

错误原因分析: 2.5无法转成类对象, 无法调用成员函数实现'+'的重载, 重载运算符的函数是对象加对象



§ 14. 运算符重载 - 转换构造和类型转换函数的使用

3. 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)
```

编译正确，运行结果：_____，能得到此结果的原因：_____

编译错误，**"<function-style-cast>": 无法从"double"转换为"Complex"**

错误原因分析：_____ 2.5无法转成类对象，没有一参构造函数，无法调用友元函数实现'+'的重载_____

```
c3 = c1 + 2.5
```

编译正确，运行结果：_____，能得到此结果的原因：_____

编译**二进制"=": 没有找到接受"double"类型的右操作数的运算符(或没有可接受的转换)**

错误原因分析：__把c1转成double类型之后与2.5相加整体是double类型，不能赋值给左侧对象，没有重载'='运算符，不能进行赋值

```
c3 = 2.5 + c1
```

编译**二进制"=": 没有找到接受"double"类型的右操作数的运算符(或没有可接受的转换)**

Linux ↑: _____ (可删除横线后贴图)

错误原因分析：__把c1转成double类型之后与2.5相加整体是double类型，不能赋值给左侧对象，没有重载'='运算符，不能进行赋值



§ 14. 运算符重载 - 转换构造和类型转换函数的使用

4. 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)
```

编译正确，运行结果：_____，能得到此结果的原因：_____

编译错误，**"<function-style-cast>": 无法从"double"转换为"Complex"**

错误原因分析：_____ 2.5无法转成类对象，没有一参构造函数，无法调用成元函数实现'+'的重载_____

```
c3 = c1 + 2.5
```

编译正确，运行结果：_____，能得到此结果的原因：_____

编译**二进制"=": 没有找到接受"double"类型的右操作数的运算符(或没有可接受的转换)**

错误原因分析：_把c1转成double类型之后与2.5相加整体是double类型，不能赋值给左侧对象，没有重载'='运算符，不能进行赋值

```
c3 = 2.5 + c1
```

编译**二进制"=": 没有找到接受"double"类型的右操作数的运算符(或没有可接受的转换)**

Linux ↑: _____ (可删除横线后贴图)

错误原因分析：_把c1转成double类型之后与2.5相加整体是double类型，不能赋值给左侧对象，没有重载'='运算符，不能进行赋值



§ 14. 运算符重载 - 转换构造和类型转换函数的使用

5. 有转换构造函数、无类型转换函数、友元方式实现复数+

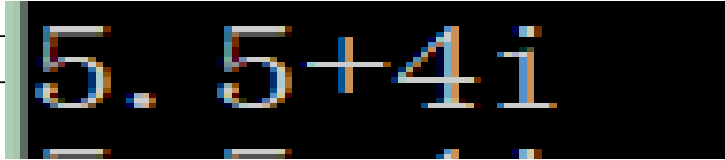
`c3 = c1 + Complex(2.5)`

编译正确，运行结果：_____，能得到此结果的原因：_转换构造函数将2.5转成对象，然后使用重载+函数相加依然为对象，再赋值给c3_

编译错误，VS2019下：_____（可删除横线后贴图）

Linux下：_____（可删除横线后贴图）

错误原因分析：_____



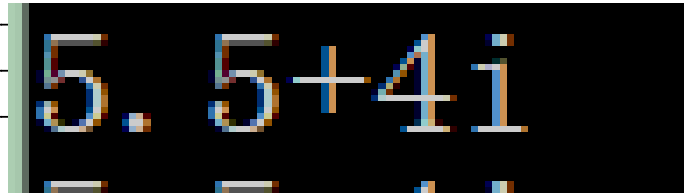
`c3 = c1 + 2.5`

编译正确，运行结果：_____，能得到此结果的原因：_转换构造函数将2.5转成对象，然后使用重载+函数相加依然为对象，再赋值给c3_

编译错误，VS2019下：_____（可删除横线后贴图）

Linux下：_____（可删除横线后贴图）

错误原因分析：_____



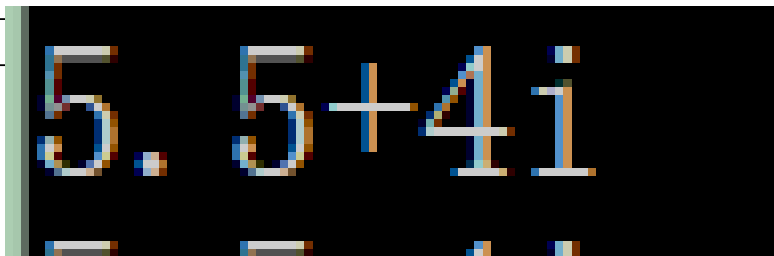
`c3 = 2.5 + c1`

编译正确，运行结果：_____，能得到此结果的原因：_转换构造函数将2.5转成对象，然后使用重载+函数相加依然为对象，再赋值给c3_

编译错误，VS2019下：_____（可删除横线后贴图）

Linux下：_____（可删除横线后贴图）

错误原因分析：_____





§ 14. 运算符重载 - 转换构造和类型转换函数的使用

6. 有转换构造函数、无类型转换函数、成员方式实现复数+

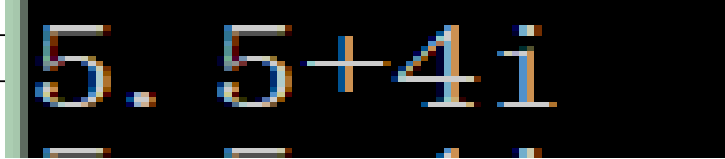
`c3 = c1 + Complex(2.5)`

编译正确，运行结果：_____，能得到此结果的原因：转换构造函数将2.5转成对象，然后使用重载+函数相加依然为对象，再赋值给c3

编译错误，VS2019下：_____（可删除横线后贴图）

Linux下：_____（可删除横线后贴图）

错误原因分析：_____



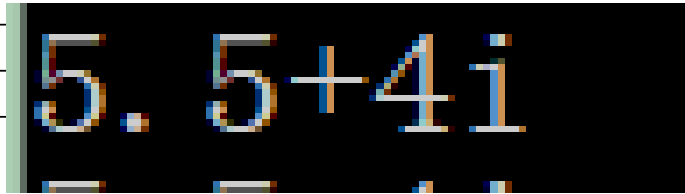
`c3 = c1 + 2.5`

编译正确，运行结果：_____，能得到此结果的原因：转换构造函数将2.5转成对象，然后使用重载+函数相加依然为对象，再赋值给c3

编译错误，VS2019下：_____（可删除横线后贴图）

Linux下：_____（可删除横线后贴图）

错误原因分析：_____



`c3 = 2.5 + c1`

编译正确，
编译错误，
Linux下：_____（可删除横线后贴图）

没有与这些操作数匹配的 "+" 运算符

错误原因分析：_____2.5在前面，且是成员函数形式，不隐式调用转换构造函数，成员函数左侧一定是对象，2.5转不了对象_____



§ 14. 运算符重载 - 转换构造和类型转换函数的使用

7. 有转换构造函数、有类型转换函数、友元方式实现复数+

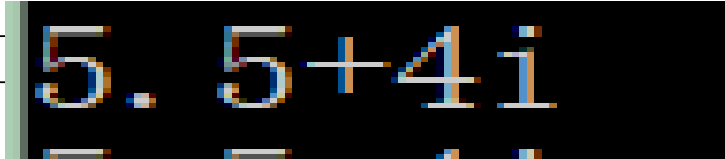
`c3 = c1 + Complex(2.5)`

编译正确，运行结果：_____，能得到此结果的原因：转换构造函数将2.5转成对象，然后使用重载+函数相加依然为对象，再赋值给c3

编译错误，VS2019下：_____（可删除横线后贴图）

Linux下：_____（可删除横线后贴图）

错误原因分析：



`c3 = c1 + 2.5`

编译正确，i

编译错误，v "operator +": 2 个重载有相似的转换

错误原因分析：___c1+2.5存在二义性，1. 先转换构造函数，然后调用重载+的函数。2. 先类型转换函数，然后double与double相加___

`c3 = 2.5 + c1`

编译正确，i

编译错误，v "operator +": 2 个重载有相似的转换

错误原因分析：___2.5+c1存在二义性，1. 先转换构造函数，然后调用重载+的函数。2. 先类型转换函数，然后double与double相加___



§ 14. 运算符重载 - 转换构造和类型转换函数的使用

8. 有转换构造函数、有类型转换函数、成员方式实现复数+

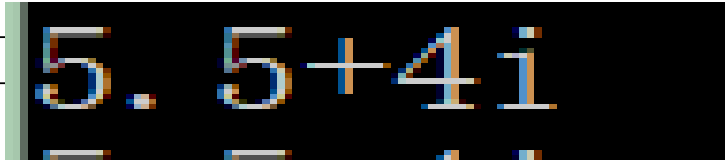
`c3 = c1 + Complex(2.5)`

编译正确，运行结果：_____，能得到此结果的原因：转换构造函数将2.5转成对象，然后使用重载+函数相加依然为对象，再赋值给c3

编译错误，VS2019下：_____（可删除横线后贴图）

Linux下：_____（可删除横线后贴图）

错误原因分析：



`c3 = c1 + 2.5`

编译正确，i

编译错误，v "operator +": 2 个重载有相似的转换

错误原因分析：_____c1+2.5存在二义性，1. 先转换构造函数，然后调用重载+的函数。2. 先类型转换函数，然后double与double相加_____

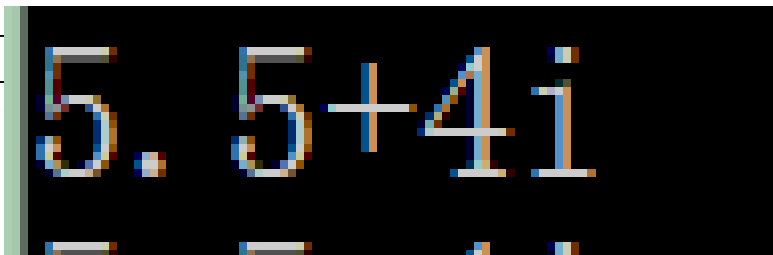
`c3 = 2.5 + c1`

编译正确，运行结果：_____，能得到此结果的原因：_____成员函数的形式前面第一个参数不会隐式调用转换构造函数，那么这种情况下，只有一种处理办法，即c1调用类型转换函数，然后与前面的2.5相加，得到整体为double类型数据，然后此时没有限制了，就可以调用转换构造函数转成对象，然后赋值给c3。所以正确_____

编译错误，VS2019下：_____（可删除横线后贴图）

Linux下：_____（可删除横线后贴图）

错误原因分析



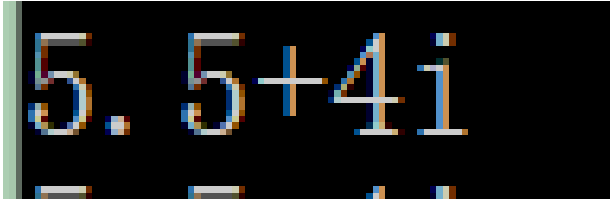


§ 14. 运算符重载 - 转换构造和类型转换函数的使用

9. 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？



2、运行结果是多少？

3、为什么和8的结果不同？

这个是正确的，因为这个有严格匹配，则直接匹配对应的友元全局函数即重载+的函数，则不会去选择转换然后二义性了。



§ 14. 运算符重载 - 转换构造和类型转换函数的使用

10. 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：_a+b本身已经返回对象了，但是直接调用类型转换函数变成double数据然后再输出。_____

仅允许改动两行，使程序输出与期望值相同：

改动第__33__行，原内容： `friend ostream & operator<<(ostream &out, Complex& c)`

新内容：

```
friend ostream & operator<<(ostream &out, Complex c);
```

改动第__36__行，原内容：

```
ostream &operator<<(ostream &out, Complex& c)
```

新内容：

```
ostream &operator<<(ostream &out, Complex c)
```