

§ 16. 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 **VS2022 + 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、**11月16日18:30前**网上提交本次作业（在“文档作业”中提交）



§ 16. 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VS2022/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

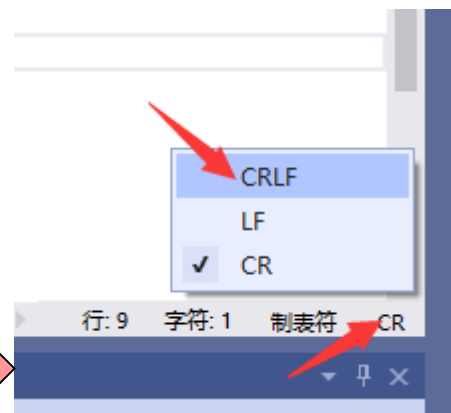
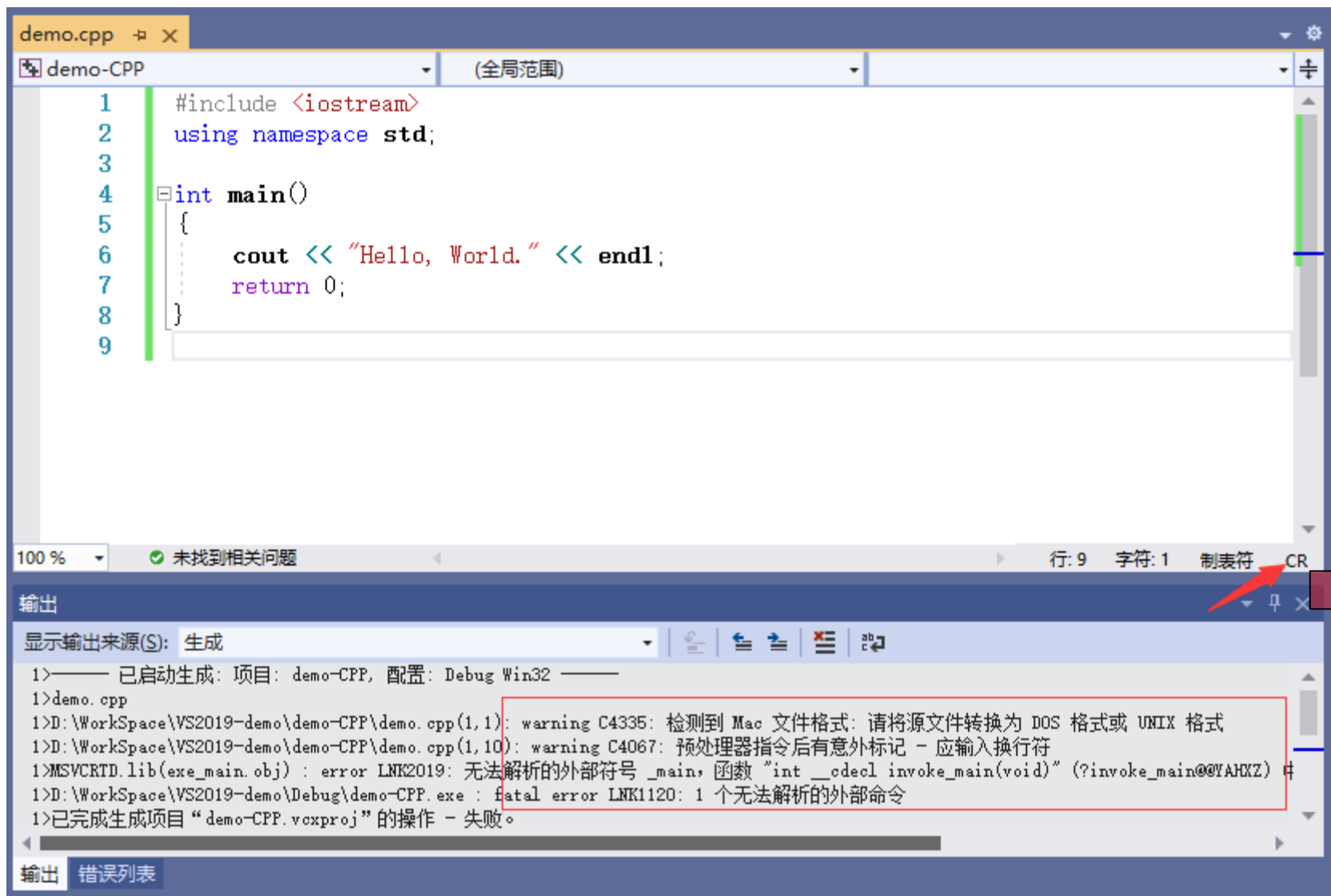
5、不需要填写的部分可以删除（例如：某句正确，则错误部分不填，或填写/即可）



§ 16. 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VS2022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





§ 16. 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)

编译错误, VS2022下:

Linux下: error C2440: "<function-style-cast>": 无法从 "double" 转换为 "Complex"

16-b1-1-无转换构造-无类型转换-友元实现+重载.cpp:60:30: 错误: no matching function for call to 'Complex::Complex(double)'
60 | c3 = c1 + Complex(2.5);
| ^~~~~~
| A

错误原因分析: 1、有complex+complex重载, 但是没有正确的转换 2、没有一参构造函数使得complex(2.5)成为complex类 3、没有double到complex的类型转换函数。

c3 = c1 + 2.5

编译错误, VS2022下: error C2679: 二元 "+": 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: 16-b2-1.cpp:63:17: 错误: no match for 'operator+' (operand types are 'Complex' and 'double')
63 | c3 = c1 + 2.5;
| ^~~~~~
| A ~~~~
| |
| Complex double

错误原因分析: 1、无double+complex的重载 2、无double转complex的转换构造函数 3、无complex转double的类型转换函数

c3 = 2.5 + c1

编译错误, VS2022下: error C2678: 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: 16-b2-1.cpp:66:18: 错误: no match for 'operator+' (operand types are 'double' and 'Complex')
66 | c3 = 2.5 + c1;

错误原因分析: 1、无complex+double的重载 2、无double转complex的转换构造函数

3、无complex转double的类型转换函数

★ 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

编译错!!!

- 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+

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



§ 16. 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)
```

编译错误, VS2022下: `error C2440: "<function-style-cast>": 无法从 "double" 转换为 "Complex"`
`60:30: 错误: no matching function for call to 'Complex::Complex(double)'`

Linux下: `c3 = c1 + Complex(2.5);`

错误原因分析: 1、没有一参构造函数使得`Complex(2.5)`成为`Complex`类 2、没有`double`到`Complex`的转换构造函数 3、没有`Complex`到`double`的类型转换函数

```
c3 = c1 + 2.5
```

编译错误, VS2022下: `error C2679: 二元 "+" : 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: `63:17: 错误: no match for 'operator+' (operand types are 'Complex' and 'double')`
`c3 = c1 + 2.5;`

错误原因分析: 1、无`Complex+double`的重载 2、无`double`转`Complex`的转换构造函数 3、无`Complex`转`double`的类型转换函数

```
c3 = 2.5 + c1
```

编译错误, VS2022下: `error C2677: 二进制 "+" : 没有找到接受 "Complex" 类型的全局运算符(或没有可接受的转换)`

Linux下: `66:18: 错误: no match for 'operator+' (operand types are 'double' and 'Complex')`
`c3 = 2.5 + c1;`

错误原因分析: 1、无`double+Complex`的重载 2、无`double`转`Complex`的转换构造函数 3、无`Complex`转`double`的类型转换函数 4、2.5是`double`型, 而成员函数实现复数+需要左加数为`Complex`类



§ 16. 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)
```

编译错误, VS2022下: error C2440: “<function-style-cast>”: 无法从“double”转换为“Complex”

Linux下: 60:30: 错误: no matching function for call to 'Complex::Complex(double)'
c3 = c1 + Complex(2.5);

错误原因分析: 1、没有一参构造函数使得complex(2.5)成为complex类 2、没有double到complex的转换构造函数

```
c3 = c1 + 2.5
```

编译错误, VS2022下:

Linux下: error C2679: 二元“=”: 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)
63:19: 错误: no match for 'operator=' (operand types are 'Complex' and 'double')
c3 = c1 + 2.5;

错误原因分析: c1被隐式转换为double, double相加, 但没有double赋值给complex的=重载

```
c3 = 2.5 + c1
```

编译错误, VS2022下:

Linux下: error C2679: 二元“=”: 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)
66:20: 错误: no match for 'operator=' (operand types are 'Complex' and 'double')
c3 = 2.5 + c1;

错误原因分析: c1被隐式转换为double, double相加, 但没有double赋值给complex的=重载



§ 16. 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)
```

编译错误, VS2022下: error C2440: “<function-style-cast>”: 无法从“double”转换为“Complex”

Linux下: 60:30: 错误: no matching function for call to 'Complex::Complex(double)'
c3 = c1 + Complex(2.5);

错误原因分析: 1、没有一参构造函数使得complex(2.5)成为complex类

```
c3 = c1 + 2.5
```

编译错误, VS2022下: error C2679: 二元“=”: 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: 错误: no match for 'operator=' (operand types are 'Complex' and 'double')
c3 = c1 + 2.5;

错误原因分析: c1被隐式转换为double, double相加, 但没有double赋值给complex的=重载

```
c3 = 2.5 + c1
```

编译错误, VS2022下: error C2679: 二元“=”: 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: 66:20: 错误: no match for 'operator=' (operand types are 'Complex' and 'double')
c3 = 2.5 + c1;

错误原因分析: c1被隐式转换为double转, double相加, 但没有double赋值给complex的=重载



§ 16. 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：[通过显式调用一参转换构造函数，建立无名complex对象，再与c1通过友元方式实现重载的+相加](#)

```
C:\> Microsoft Visual Studio 调试控制台  
5.5+4i
```

```
[u2152041@oop ~]$ ./test  
5.5+4i
```

```
c3 = c1 + 2.5
```

编译正确，运行结果：，能得到此结果的原因：[隐式调用转换构造函数转换double数据，建立无名complex对象，再与c1通过友元方式实现重载的+相加](#)

```
C:\> Microsoft V  
5.5+4i
```

```
[u2152041@oop ~]$ ./test  
5.5+4i
```

```
c3 = 2.5 + c1
```

编译正确，运行结果：，能得到此结果的原因：[隐式调用转换构造函数转换double数据，建立无名complex对象，再与c1通过友元方式实现重载的+相加](#)

```
C:\> Microsoft Visu:  
5.5+4i
```

```
[u2152041@oop ~]$ ./test  
5.5+4i
```




§ 16. 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：[通过显式调用一参转换构造函数，建立无名complex对象，再与c1通过成员方式重载的+相加](#)

```
c:\Microsoft Visual Studio i
5.5+4i
```

```
[u2152041@oop ~]$ ./tes
5.5+4i
```

```
c3 = c1 + 2.5
```

编译正确，运行结果：，能得到此结果的原因：[隐式调用转换构造函数转换double数据，建立无名complex对象，再与c1通过友元方式实现重载的+相加](#)

```
c:\Microsoft Visual Studi
5.5+4i
```

```
[u2152041@oop ~]$ ./test
5.5+4i
```

```
c3 = 2.5 + c1
```

编译错误，VS2022下：

error C2677: 二进制“+”：没有找到接受“Complex”类型的全局运算符(或没有可接受的转换)

Linux下：66:18: 错误: no match for 'operator+' (operand types are 'double' and 'Complex')

```
c3 = 2.5 + c1;
```

错误原因分析：1、+的优先级是从左往右，所以没有调用隐式转换 2、成员函数实现的复数+要求左侧必须是complex类 3、没有double+complex的+重载

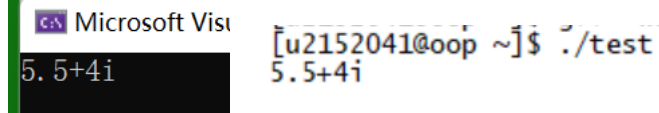


§ 16. 运算符重载 - 转换构造和类型转换函数的使用

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：通过显式调用一参转换构造函数，建立无名complex对象，再与c1通过成员方式重载的+相加



```
[u2152041@oop ~]$ ./test
5.5+4i
```

```
c3 = c1 + 2.5
```

编译错误，VS2022下：error C2666: “operator +” : 2 个重载有相似的转换

Linux下：63:17: 错误: ambiguous overload for ‘operator+’ (operand types are ‘Complex’ and ‘double’)
c3 = c1 + 2.5;

错误原因分析：+既可以调用类型转换函数将complex转换为double，也可以隐式调用转换构造函数建立无名complex类，且优先级相同，使得编译器报错

```
c3 = 2.5 + c1
```

编译错误，VS2022下：error C2666: “+” : 2 个重载有相似的转换

Linux下：66:18: 错误: ambiguous overload for ‘operator+’ (operand types are ‘double’ and ‘Complex’)
c3 = 2.5 + c1;

错误原因分析：+既可以调用类型转换函数将complex转换为double，也可以隐式调用转换构造函数建立无名complex类，且优先级相同，使得编译器报错



§ 16. 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：[通过显式调用一参转换构造函数，建立无名complex对象，再与c1通过成员方式重载的+相加](#)

```
Microsoft [u2152041@oop ~]$ ./test
5.5+4i
5.5+4i
```

```
c3 = c1 + 2.5
```

编译错误，VS2022下：

```
error C2666: "Complex::operator +": 2 个重载有相似的转换
```

Linux下: **63:17: 错误: ambiguous overload for 'operator+' (operand types are 'Complex' and 'double')**
`c3 = c1 + 2.5;`

错误原因分析：[+既可以调用类型转换函数将complex转换为double，也可以隐式调用转换构造函数建立无名complex类，且优先级相同，使得编译器报错](#)

```
c3 = 2.5 + c1
```

编译正确，运行结果：，能得到此结果的原因：[先调用类型转换函数，然后再调用一参转换构造函数，实现了成员方式+](#)

```
Microsoft [u2152041@oop ~]$ ./test
5.5+0i
5.5+0i
```



§ 16. 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？

存在 `double+complex` 的+重载，且返回为 `complex` 类，可以赋值给 `c3`

2、运行结果是多少？

```
Microsoft [u2152041@oop ~]$ ./test
5.5+4i 5.5+4i
```

3、为什么和8的结果不同？

因为8中，在类型转换函数后实现了 `double` 相加成为实部，在调用转换构造函数时虚部默认为0，而此题（即9）中将 `complex` 的虚部默认为 `double+complex` 类的结果的虚部



§ 16. 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：输出a+b时隐式调用了类型转换函数，使得输出值为复数在复平面上的点到原点的距离

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 const & c);`

改动第36行，原内容：`ostream& operator<<(ostream& out, Complex & c)`

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