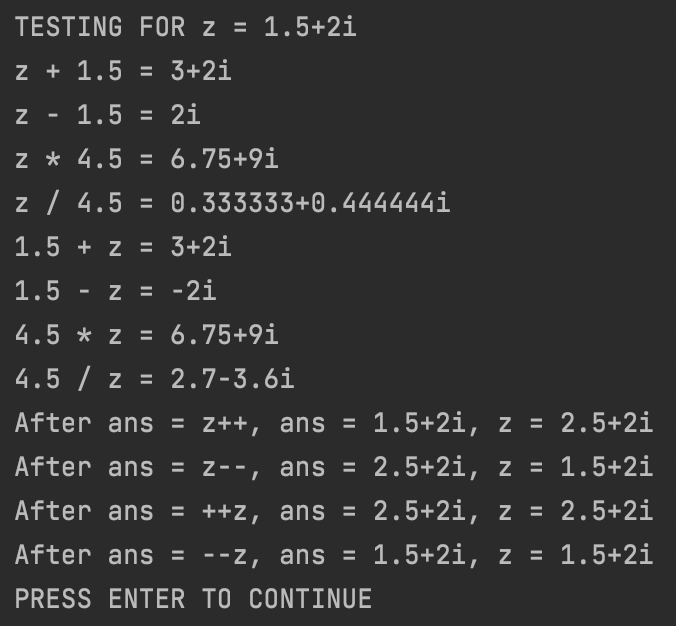
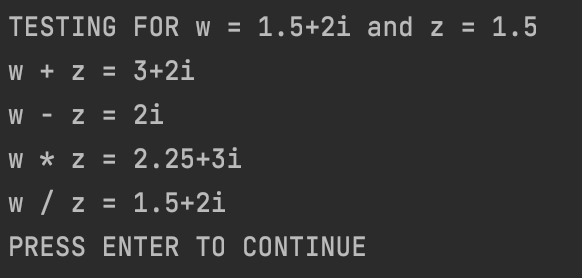
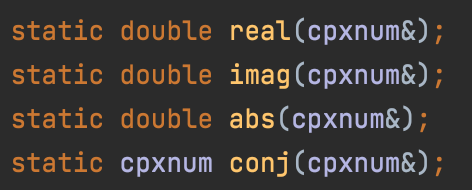
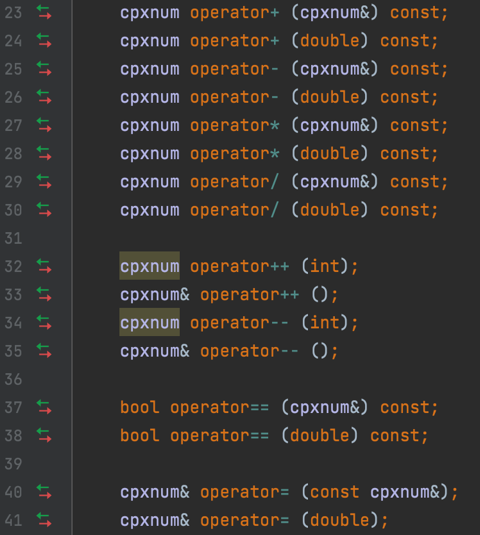
面向对象程序设计基础作业五 设计文档

1. 模型部分
   1. 功能简述

本程序内包含一个复数类cpxnum，可以实现复数的储存、计算工作（详见b）。在运行时，本程序将对包含的复数类进行测试（详见2），并输出结果。下图为测试过程中两个样例输出。

* 1. 数据结构

本程序包含复数类cpxnum，有两个double型私有参数real\_part和imaginary\_part，分别存储复数的实数部分和虚数部分。cpxnum内有四个公有接口，分别用于调用本复数的实部、虚部、模长和共轭。另外，cpxnum类重载了如上左图所示的操作符，在c. 算法部分将一一说明。

* 1. 算法
     1. 自增、自减操作符
        1. 前、前

此操作符可以使被操作数的实部增加/减少1，虚部不变。用在表达式中时，先改变被操作数的值，再获取改变后被操作数的值。

* + - 1. 后、后

此操作符可以使被操作数的实部增加/减少1，虚部不变。用在表达式中时，先获取被操作数的值，再改变被操作数的值。

* + 1. 四则运算符

这些操作符的功能和它们数学上的功能相一致。本程序中的cpxnum类允许复数和实数混合运算，且顺序可交换。

特别注明：本程序没有进行“除以零”判断。鉴于计算过程的“除以零”异常会被正常抛出，在实际使用时和实数运算中处理除以零错误的方式是一样的。

* + 1. 赋值操作符

本程序为cpxnum类重载了符值操作符=，支持将复数或实数赋值给cpxnum类实例。当将实数赋值给cpxnum类时，cpxnum类会将虚部设置为零。

* + 1. 逻辑运算符

本程序为cpxnum类重载了逻辑运算符==，支持与另一个复数或实数相比较，返回二者是否相等。

其他的逻辑运算符没有重载，原因是复数不可比较大小。

* + 1. 插入流运算符 (std::ostream) <<

为了更便捷地输出复数，本程序为std::ostream类重载了针对复数的插入流运算符。被调用时，会以a+bi的形式输出复数结果；如果a或b等于零，对应的部分不会显示。如果a与b都等于零，会显示0.如果a或b等于正负1，不会输出此多余的1.

1. 验证部分

本程序的验证部分使用如下测试样例：

|  |  |  |  |
| --- | --- | --- | --- |
| 复数编号 | 值 | | 代表的等价类 |
| 实部 | 虚部 |
| z­1 | 1.5 | 2 | 普通复数 |
| z­2 | 1.5 | 0 | 实数 |
| z­3 | 0 | 0 | 零 |
| z­4 | 0 | 1.5 | 纯虚数 |
| z5 | -1 | -1 | 实部与虚部为负数的普通复数 |

* 1. 仅涉及一个复数的运算符测试

对于上表每一个复数，分别做如下测试：输出它的值；复数与实数四则运算；实数与复数四则运算；自增（两种）；自减（两种）。这些测试可以验证重载的四则运算操作符·（· 为 之一）是否可用于复数·实数和实数·复数两种情况，验证自增、自减操作符是否能改变复数的值，比较前自增（减）和后自增（减）操作符的结果差异，以及测试插入流运算符的输出是否符合格式。

结果见*Table 1*。

* 1. 二元运算符的测试

对于上表每两个不相同的复数，分别做如下测试：输出它们的值；复数与复数四则运算。这些测试主要是为了测试复数之间重载的四则运算符。

结果见*Table 2*。

Table 1 涉及一个复数的运算符验证结果

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 序号 | cout << z | z+1.5 | z-1.5 | z\*1.5 | z / 1.5 | z++的值 | ++z的值 | z--的值 | --z的值 |
|  |  | 1.5+z | 1.5-z | 1.5\*z | 1.5 / z | z++后z的值 | ++z后z的值 | z--后z的值 | --z后z的值 |
| z­1 | 1.5+2i | 3+2i | 2i | 6.75+9i | 0.33+0.44i | 1.5+2i | 2.5+2i | 2.5+2i | 1.5+2i |
|  |  | 3+2i | -2i | 6.75+9i | 0.33+0.44i | 2.5+2i | 1.5+2i | 2.5+2i | 1.5+2i |
| z­2 | 1.5 | 3 | 0 | 6.75 | 0.33 | 1.5 | 2.5 | 2.5 | 1.5 |
|  |  | 3 | 0 | 6.75 | 4.5 | 2.5 | 1.5 | 2.5 | 1.5 |
| z­3 | 0 | 1.5 | -1.5 | 0 | 0 | 0 | 1 | 1 | 0 |
|  |  | 1.5 | 1.5 | 0 | NAN[[1]](#footnote-1) | 1 | 0 | 1 | 0 |
| z­4 | 1.5i | 1.5+1.5i | -1.5+1.5i | 6.75i | 0.33i | 1.5i | 1+1.5i | 1+1.5i | 1.5i |
|  |  | 1.5+1.5i | 1.5-1.5i | 6.75i | -4.5i | 1+1.5i | 1.5i | 1+1.5i | 1.5i |
| z5 | -1-i | 0.5-i | -2.5-i | -4.5-4.5i | -0.22-0.22i | -1-i | -i | -i | -1-i |
|  |  | 0.5-i | 2.5+i | -4.5-4.5i | -3.18+3.18i | -i | -1-i | -i | -1-i |

Table 2 涉及两个复数的运算符验证结果

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 第一操作数 | 第二操作数 | 相加 | 相减 | 相乘 | 相除 |
| z­1 | z­2 | 3+2i | 2i | 2.25+3i | 1.5+2i |
| z­3 | 1.5+2i | 1.5+2i | 0 | NAN |
| z­4 | 1.5+3.5i | 1.5+0.5i | -3+2.25i | 2-1.5i |
| z5 | 0.5+i | 2.5+3i | 0.5-3.5i | -2.47-0.35i |
| z­2 | z­1 | 3+2i | -2i | 2.25+3i | 0.9-1.2i |
| z­3 | 1.5 | 1.5 | 0 | NAN |
| z­4 | 1.5+1.5i | 1.5-1.5i | 2.25i | -1.5i |
| z5 | 0.5-i | 2.5+i | -1.5-1.5i | -1.06+1.06i |
| z­3 | z1 | 1.5+2i | -1.5-2i | 0 | 0 |
| z­2 | 1.5 | -1.5 | 0 | 0 |
| z­4 | 1.5i | -1.5i | 0 | 0 |
| z5 | -1-i | 1+i | 0 | 0 |
| z­4 | z1 | 1.5+3.5i | -1.5-0.5i | -3+2.25i | 1.2+0.9i |
| z­2 | 1.5+1.5i | -1.5+1.5i | 2.25i | 1.5i |
| z­3 | 1.5i | 1.5i | 0 | NAN |
| z5 | -1+0.5i | 1+2.5i | 1.5-1.5i | -1.06-1.06i |
| z5 | z1 | 0.5+i | -2.5-3i | 0.5-3.5i | -1.4+0.2i |
| z­2 | 0.5-i | -2.5-i | -1.5-1.5i | -1-i |
| z­3 | -1-i | -1-i | 0 | NAN |
| z4 | -1+0.5i | -1-2.5i | 1.5-1.5i | -1+i |

1. NAN为除以零异常，是本程序意料之内。下文同。详见1.c.ii. [↑](#footnote-ref-1)