目标：

编写C++程序来达到：编写复数类，实现复数的加法、减法、乘法、除法、前置“++”、后置“++”、前置“--”、后置“--”等运算符重载，提供验证程序并为每种运算符重载至少提供3个测试案例，实现复数的整数指数运算与实数指数运算。

原理：

构建Complex类 以及 在计算复数的实数指数次幂的时候用的结构体 如下：

class Complex

{

private:

double Real;

double Imagination;

public:

Complex(double R = 0, double I = 0):Real(R), Imagination(I){}

Complex operator + (Complex &);

Complex operator - (Complex &);

Complex operator \* (Complex &);

Complex operator / (Complex &);

Complex operator + (const double &);

Complex operator - (const double &);

Complex operator \* (const double &);

Complex operator / (const double &);

Complex & operator ++ ();

Complex operator ++ (int);

Complex & operator -- ();

Complex operator -- (int);

friend Complex operator + (const double &, Complex &);

friend Complex operator - (const double &, Complex &);

friend Complex operator \* (const double &, Complex &);

friend Complex operator / (const double &, Complex &);

bool operator == (Complex &);

bool operator == (const double &);

friend bool operator == (const double &, Complex &);

friend ostream & operator << (ostream &, const Complex &);

friend Complex Power(Complex & root, const double & times);

friend Complex Power(Complex & root, const int & times);

double Get\_Real();

double Get\_Imagination();

void Print();

Power\_Form\_Of\_Complex To\_Power\_Form();

friend Complex To\_Normal\_Form(Power\_Form\_Of\_Complex &);

};

struct Power\_Form\_Of\_Complex

{

double Rho\_Squared;

double Theta;

};

在复数类中：

* 实现了运算符 + - \* / 前++ 后++ 前-- 后-- 输出流<< 判断相等的== 的重载
* 实现了实数复数混合运算（包括它们的任意次序）
* 实现了复数的任意指数次幂的运算（包括整数次和实数次，包括正数、负数和0次）
* 实现了实数复数判定相等的函数
* 实现了普通形式和指数形式的换算
* 实现了正常书写形式输出（如2i， 1-2i这样的形式）（在print函数以及输出流中）
* 避免了除以零的程序崩溃（会返回

#define Divided\_By\_Zero\_Num Complex(DBL\_MAX, DBL\_MAX) 包括在除法（除以实数和复数两种情况）以及指数次数幂时）

**如何编程和运行：**

* 软件构件库部分：

编写Complex类以及其中函数的具体实现。

* 构件库测试部分：

编写所需要的测试函数，并且在其中内置样例。

* 构件库应用部分：

编写main函数，在函数中提供选择自动和手动测试的功能。

结果：

以下测试数据均在64位操作系统下用VS 2017 测试得到。

自动测试程序运行结果：

------------------------------------------------------------

输入 1 进行手动测试，输入 2 进行自动测试！

------------------------------------------------------------

2

------------------------------------------------------------

第一轮测试开始！

------------------------------------------------------------

(1 + 3i)+(0.999 + 10i)=1.999 + 13i

(-96.2 + 35.6i)+(96.2 + 3i)=38.6i

(12.3 - 63.2i)+(15.965 + 63.3i)=28.265 + 0.1i

(1 + 31.3i)-(1 + 31.3i)=0

(5 + 26i)-(1 + 96.525i)=4 - 70.525i

(96.2 + 2.6i)-(19586 + 9.25i)=-19489.8 - 6.65i

(1 + 3i)\*(-3 + 1i)=-6 - 8i

(-5.625 + 2.66i)\*(0)=-0

(-96.2 + 2.6i)\*(-19586 - 9.25i)=1.8842e+06 - 50033.8i

(1 + 31.3i)/(0)=1.79769e+308 + 1.79769e+308i

(-5 + 2.6i)/(1.69 - 9.6i)=-0.351625 - 0.458933i

(36.2 + 2.6i)/(3.62 + 0.26i)=10

原始数据为 3.26 + 15i ，之后计算 3.26 + 15i ++ = 3.26 + 15i ，现在计算结束后为 4.26 + 16i

原始数据为 4.26 + 16i ，之后计算 ++ 4.26 + 16i = 5.26 + 17i ，现在计算结束后为 5.26 + 17i

原始数据为 5.26 + 17i ，之后计算 5.26 + 17i -- = 5.26 + 17i ，现在计算结束后为 4.26 + 16i

原始数据为 4.26 + 16i ，之后计算 -- 4.26 + 16i = 3.26 + 15i ，现在计算结束后为 3.26 + 15i

原始数据为 1.36 - 96i ，之后计算 1.36 - 96i ++ = 1.36 - 96i ，现在计算结束后为 2.36 - 95i

原始数据为 2.36 - 95i ，之后计算 ++ 2.36 - 95i = 3.36 - 94i ，现在计算结束后为 3.36 - 94i

原始数据为 3.36 - 94i ，之后计算 3.36 - 94i -- = 3.36 - 94i ，现在计算结束后为 2.36 - 95i

原始数据为 2.36 - 95i ，之后计算 -- 2.36 - 95i = 1.36 - 96i ，现在计算结束后为 1.36 - 96i

原始数据为 2 ，之后计算 2 ++ = 2 ，现在计算结束后为 3 + 1i

原始数据为 3 + 1i ，之后计算 ++ 3 + 1i = 4 + 2i ，现在计算结束后为 4 + 2i

原始数据为 4 + 2i ，之后计算 4 + 2i -- = 4 + 2i ，现在计算结束后为 3 + 1i

原始数据为 3 + 1i ，之后计算 -- 3 + 1i = 2 ，现在计算结束后为 2

下面开始计算复数的指数次幂运算！

原来的数为： 3.6 + 9.562i

它的1次方为: 3.6 + 9.562i

它的2次方为: -78.4718 + 68.8464i

它的3次方为: -940.808 - 502.501i

它的4次方为: 1418 - 10805i

它的5次方为: 108422 - 25339.1i

它的0次方为: 1

它的-1次方为: 0.0344855 - 0.0915972i

它的-2次方为: -0.0072008 - 0.00631754i

它的-3次方为: -0.000826992 + 0.00044171i

它的-4次方为: 1.19402e-05 + 9.09827e-05i

它的-5次方为: 8.74552e-06 + 2.04389e-06i

它的0.5次方为: 3.14478 + 0.572378i

它的0.1次方为: 1.26082 + 0.0454187i

它的9.6次方为: -4.65229e+09 - 1.51668e+09i

它的-0.5次方为: 0.307791 - 0.0560208i

原来的数为： 9 - 35i

它的1次方为: 9 - 35i

它的2次方为: -1144 - 630i

它的3次方为: -32346 + 34370i

它的4次方为: 911836 + 1.44144e+06i

它的5次方为: 5.86569e+07 - 1.89413e+07i

它的0次方为: 1

它的-1次方为: 0.00689127 + 0.0267994i

它的-2次方为: -0.000670718 + 0.000369364i

它的-3次方为: -1.45208e-05 - 1.54294e-05i

它的-4次方为: 3.13433e-07

它的-5次方为: 1.54384e-08

它的0.5次方为: 5.964 - 0.754527i

它的0.1次方为: 1.43107 - 0.0360261i

它的9.6次方为: -6.76981e+14 - 6.0022e+14i

它的-0.5次方为: 0.165031 + 0.0208787i

原来的数为： -1 - 2i

它的1次方为: -1 - 2i

它的2次方为: -3 + 4i

它的3次方为: 11 + 2i

它的4次方为: -7 - 24i

它的5次方为: -41 + 38i

它的0次方为: 1

它的-1次方为: -0.2 + 0.4i

它的-2次方为: -0.12 - 0.16i

它的-3次方为: 0.088 - 0.016i

它的-4次方为: -0.0112 + 0.0384i

它的-5次方为: -0.01312 - 0.01216i

它的0.5次方为: 1.45535 + 0.343561i

它的0.1次方为: 1.08263 + 0.0502321i

它的9.6次方为: -585.273 - 2188.01i

它的-0.5次方为: 0.650851 - 0.153645i

原来的数为： 2i

它的1次方为: 2i

它的2次方为: -4

它的3次方为: - 8i

它的4次方为: 16

它的5次方为: 32i

它的0次方为: 1

它的-1次方为: - 0.5i

它的-2次方为: -0.25

它的-3次方为: 0.125i

它的-4次方为: 0.0625

它的-5次方为: - 0.03125i

它的0.5次方为: 1 + 1i

它的0.1次方为: 1.05858 + 0.167662i

它的9.6次方为: -627.835 + 456.149i

它的-0.5次方为: 0.5 - 0.5i

原来的数为： 3

它的1次方为: 3

它的2次方为: 9

它的3次方为: 27

它的4次方为: 81

它的5次方为: 243

它的0次方为: 1

它的-1次方为: 0.333333

它的-2次方为: 0.111111

它的-3次方为: 0.037037

它的-4次方为: 0.0123457

它的-5次方为: 0.00411523

它的0.5次方为: 1.73205

它的0.1次方为: 1.11612

它的9.6次方为: 38050.8

它的-0.5次方为: 0.57735

------------------------------------------------------------

下面即将进行第二轮测试！

请按任意键继续. . .

第二轮测试开始！

------------------------------------------------------------

(25 + 0.6i) + (952) 的结果计算为 977 + 0.6i

(25 + 0.6i) + (3.5) 的结果计算为 28.5 + 0.6i

(0) + (952) 的结果计算为 952

(15.2 - 958625i) + (-35 + 3.8i) 的结果计算为 -19.8 - 958621i

(15.2 - 958625i) + (-0.0006) 的结果计算为 15.1994 - 958625i

(3.02) + (-35 + 3.8i) 的结果计算为 -31.98 + 3.8i

(99.3i) + (584.6 - 2.6i) 的结果计算为 584.6 + 96.7i

(99.3i) + (925.636) 的结果计算为 925.636 + 99.3i

(9.65) + (584.6 - 2.6i) 的结果计算为 594.25 - 2.6i

(25 + 0.6i) - (952) 的结果计算为 -927 + 0.6i

(25 + 0.6i) - (3.5) 的结果计算为 21.5 + 0.6i

(0) - (952) 的结果计算为 -952

(15.2 - 958625i) - (-35 + 3.8i) 的结果计算为 50.2 - 958629i

(15.2 - 958625i) - (-0.0006) 的结果计算为 15.2006 - 958625i

(3.02) - (-35 + 3.8i) 的结果计算为 38.02 - 3.8i

(99.3i) - (584.6 - 2.6i) 的结果计算为 -584.6 + 101.9i

(99.3i) - (925.636) 的结果计算为 -925.636 + 99.3i

(9.65) - (584.6 - 2.6i) 的结果计算为 -574.95 + 2.6i

(25 + 0.6i) \* (952) 的结果计算为 23800 + 571.2i

(25 + 0.6i) \* (3.5) 的结果计算为 87.5 + 2.1i

(0) \* (952) 的结果计算为 0

(15.2 - 958625i) \* (-35 + 3.8i) 的结果计算为 3.64224e+06 + 3.35519e+07i

(15.2 - 958625i) \* (-0.0006) 的结果计算为 -0.00912 + 575.175i

(3.02) \* (-35 + 3.8i) 的结果计算为 -105.7 + 11.476i

(99.3i) \* (584.6 - 2.6i) 的结果计算为 258.18 + 58050.8i

(99.3i) \* (925.636) 的结果计算为 91915.6i

(9.65) \* (584.6 - 2.6i) 的结果计算为 5641.39 - 25.09i

(25 + 0.6i) / (952) 的结果计算为 0.0262605 + 0.000630252i

(25 + 0.6i) / (3.5) 的结果计算为 7.14286 + 0.171429i

(0) / (952) 的结果计算为 0

(15.2 - 958625i) / (-35 + 3.8i) 的结果计算为 -2939.48 + 27070.1i

(15.2 - 958625i) / (-0.0006) 的结果计算为 -25333.3 + 1.59771e+09i

(3.02) / (-35 + 3.8i) 的结果计算为 -0.0852804 - 0.00925902i

(99.3i) / (584.6 - 2.6i) 的结果计算为 -0.000755434 + 0.169856i

(99.3i) / (925.636) 的结果计算为 0.107278i

(9.65) / (584.6 - 2.6i) 的结果计算为 0.0165067 + 7.34132e-05i

((25 + 0.6i) ++) \* (-- (952)) 的结果计算为 23775.6 + 545.6i

((26 + 1.6i) ++) \* (-- (3.5)) 的结果计算为 65 + 4i

((0) ++) \* (-- (951 - 1i)) 的结果计算为 0

((15.2 - 958625i) ++) \* (-- (-35 + 3.8i)) 的结果计算为 2.6836e+06 + 3.45105e+07i

((16.2 - 958624i) ++) \* (-- (-0.0006)) 的结果计算为 -16.2097 + 959199i

((3.02) ++) \* (-- (-36 + 2.8i)) 的结果计算为 -111.74 + 5.436i

((99.3i) ++) \* (-- (584.6 - 2.6i)) 的结果计算为 357.48 + 57951.5i

((1 + 100.3i) ++) \* (-- (925.636)) 的结果计算为 924.636 + 92741i

((9.65) ++) \* (-- (583.6 - 3.6i)) 的结果计算为 5622.09 - 44.39i

(-- (25 + 0.6i)) \* ((952) ++)的结果计算为 22848 - 380.8i

(-- (24 - 0.4i)) \* ((3.5) ++)的结果计算为 80.5 - 4.9i

(-- (0)) \* ((953 + 1i) ++)的结果计算为 -953 - 1i

(-- (15.2 - 958625i)) \* ((-35 + 3.8i) ++)的结果计算为 3.64228e+06 + 3.3552e+07i

(-- (14.2 - 958626i)) \* ((-0.0006) ++)的结果计算为 -0.00792 + 575.176i

(-- (3.02)) \* ((-34 + 4.8i) ++)的结果计算为 -68.68 + 9.696i

(-- (99.3i)) \* ((584.6 - 2.6i) ++)的结果计算为 -329.02 + 57468.8i

(-- (-1 + 98.3i)) \* ((925.636) ++)的结果计算为 -1851.27 + 90064.4i

(-- (9.65)) \* ((585.6 - 1.6i) ++)的结果计算为 5065.44 - 13.84i

请按任意键继续. . .

程序验证结果报告：

* 对于64位的计算机

在上述的计算过程中未发现错误。

测试案例：

(1 + 3i)+(0.999 + 10i)=1.999 + 13i

(-96.2 + 35.6i)+(96.2 + 3i)=38.6i

(12.3 - 63.2i)+(15.965 + 63.3i)=28.265 + 0.1i

(1 + 31.3i)-(1 + 31.3i)=0

(5 + 26i)-(1 + 96.525i)=4 - 70.525i

(96.2 + 2.6i)-(19586 + 9.25i)=-19489.8 - 6.65i

(1 + 3i)\*(-3 + 1i)=-6 - 8i

(-5.625 + 2.66i)\*(0)=-0

(-96.2 + 2.6i)\*(-19586 - 9.25i)=1.8842e+06 - 50033.8i

(1 + 31.3i)/(0)=1.79769e+308 + 1.79769e+308i

(-5 + 2.6i)/(1.69 - 9.6i)=-0.351625 - 0.458933i

(36.2 + 2.6i)/(3.62 + 0.26i)=10

原始数据为 3.26 + 15i ，之后计算 3.26 + 15i ++ = 3.26 + 15i ，现在计算结束后为 4.26 + 16i

原始数据为 4.26 + 16i ，之后计算 ++ 4.26 + 16i = 5.26 + 17i ，现在计算结束后为 5.26 + 17i

原始数据为 5.26 + 17i ，之后计算 5.26 + 17i -- = 5.26 + 17i ，现在计算结束后为 4.26 + 16i

原始数据为 4.26 + 16i ，之后计算 -- 4.26 + 16i = 3.26 + 15i ，现在计算结束后为 3.26 + 15i

原始数据为 1.36 - 96i ，之后计算 1.36 - 96i ++ = 1.36 - 96i ，现在计算结束后为 2.36 - 95i

原始数据为 2.36 - 95i ，之后计算 ++ 2.36 - 95i = 3.36 - 94i ，现在计算结束后为 3.36 - 94i

原始数据为 3.36 - 94i ，之后计算 3.36 - 94i -- = 3.36 - 94i ，现在计算结束后为 2.36 - 95i

原始数据为 2.36 - 95i ，之后计算 -- 2.36 - 95i = 1.36 - 96i ，现在计算结束后为 1.36 - 96i

原始数据为 2 ，之后计算 2 ++ = 2 ，现在计算结束后为 3 + 1i

原始数据为 3 + 1i ，之后计算 ++ 3 + 1i = 4 + 2i ，现在计算结束后为 4 + 2i

原始数据为 4 + 2i ，之后计算 4 + 2i -- = 4 + 2i ，现在计算结束后为 3 + 1i

原始数据为 3 + 1i ，之后计算 -- 3 + 1i = 2 ，现在计算结束后为 2

以及Complex(3.6, 9.562), Complex(9, -35), Complex(-1, -2), Complex(0, 2), Complex(3, 0)的1 2 3 4 5 0 -1 -2 -3 -4 -5 0.5 0.1 9.6 -0.5次方

加法减法乘法除法：用正数负数交替检测计算能力以及输出的正确性。

其中除法中包含除以0的情况来检验上述的程序设计思路中屏蔽程序崩溃的作用有效性。

前后++ --中使用整数、负数、小数等情况进行检测。

测试结果含义：程序运行未发现问题。在测试过程中运行正常。

结论：

程序能较好完成设计目的要求，在使用的过程中未发现错误。