Steven Nguyen

Jake Nugent

CIS 368

Assignment 5

OUTPUT:

```
PS C:\gethub\objectOrientedDesignAndProrams\object-oriented-design\assignFive> java .\assignmentF.java
Enter the first complex number: 3.5 5.5
Enter the second complex number: -3.5 1
(3.5 + 5.5i) + (-3.5 + 1.0i) = 0.0 + 6.5i
(3.5 + 5.5i) - (-3.5 + 1.0i) = 7.0 + 4.5i
(3.5 + 5.5i) * (-3.5 + 1.0i) = -17.75 + -15.75i
(3.5 + 5.5i) / (-3.5 + 1.0i) = -0.5094339622641509 + -1.7169811320754718i
|3.5 + 5.5i| = 6.519202405202649
false
3.5
5.5
C1 is not the same as C2
```

COMPLEX IMPLEMENT:

```java
class Complex implements Cloneable, Comparable<Complex> {
    private double a;
    private double b;

    public Complex() {

    }

    public Complex(double a_in) {
        this.a = a_in;
        this.b = 0;
    }

    Complex(double a_in, double b_in) {
        this.a = a_in;
        this.b = b_in;

    }

    public double getRealPart() {
        return a;
    }

    public double getImaginaryPart() {
        return b;
    }

    public Complex add(Complex secondComplex) {
        double newA = a + secondComplex.a;
        double newB = b + secondComplex.b;
        return new Complex(newA, newB);
    }

    public Complex subtract(Complex secondComplex) {
        // stilll needs to be implemented
        double newA = a - secondComplex.a;
        double newB = b - secondComplex.b;
        return new Complex(newA, newB);
    }

    public Complex multiply(Complex secondComplex) {
        double newA = a * secondComplex.a - b * secondComplex.b;
        double newB = b * secondComplex.a + a * secondComplex.b;
        return new Complex(newA, newB);
    }

```

SELF IMPLEMENTED FUNCTIONS:

```java
    public Complex divide(Complex secondComplex) {
        double newA = (a * secondComplex.a + b * secondComplex.b)
                / ((secondComplex.a * secondComplex.a) + (secondComplex.b * secondComplex.b));
        double newB = (b * secondComplex.a - a * secondComplex.b)
                / ((secondComplex.a * secondComplex.a) + (secondComplex.b * secondComplex.b));
        return new Complex(newA, newB);
    }       You, 43 minutes ago • subtract and divide added …
    public double abs() {
        return Math.sqrt(a * a + b * b);
    }
    @Override
    public String toString() {

        if (b != 0) {
            return a + " + " + b + "i";
        }

        return a + "";
    }
    @Override
    public int compareTo(Complex c) {
        if ((c.a == this.a) && (c.b == this.b))
            return 1;
        else {
            return 0;
        }
    }

    @Override
    public Complex clone() {
        return new Complex(this.a, this.b);
    }
}
```