Seneca College **Jan 30, 2020**

Applied Arts & Technology

SCHOOL OF COMPUTER STUDIES

JAC444 **Demo Due dates : Jan 31,2020**

Final Code Submission Date: Feb1,2020

Workshop 3

Notes:

i. Each task should be presented during the lab, demo worth 70% of the workshop marks and code uploading worth the other 30%.

ii. At least one of the tasks should be demoed in Jan 31th lab.

iii. Make sure you have all security and check measures in place, like wrong data types etc.

iv. Make your project in proper hierarchy; introduce proper class coherence in your project. Proper packages and your project should be handled by only one main method which should be in a TesterClass.

v. Given output structure is just for student to have a glimpse what the output can look, students are free to make the output better in any way.

Other inputs can be given during demo, so make sure you test your program properly.

Task 1: A complex number is a number in the form *a* + *bi*, where *a* and *b* are real numbers and *i* is 2-1. The numbers a and b are known as the real part and imaginary part of the complex number, respectively. You can perform addition, subtraction, multiplication, and division for complex numbers using the following formulas:

$$a + bi + c + di = (a + c) + (b + d)i$$

$$a + bi - (c + di) = (a - c) + (b - d)i$$

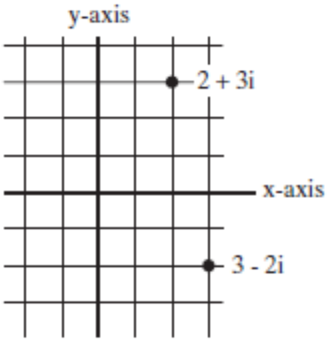
$$(a + bi) * (c + di) = (ac - bd) + (bc + ad)i$$

$$(a + bi)/(c + di) = (ac + bd)/(c^2 + d^2) + (bc - ad)i/(c^2 + d^2)$$

You can also obtain the absolute value for a complex number using the following formula:

$$|a + bi| = \sqrt{a^2 + b^2}$$

(A complex number can be interpreted as a point on a plane by identifying the (*a,b*) values as the coordinates of the point. The absolute value of the complex number corresponds to the distance of the point to the origin, as shown in Figure)



- Design a class named Complex for representing complex numbers.
- The methods
    - add,
    - subtract,
    - multiply,
    - divide

For performing complexnumber operations.

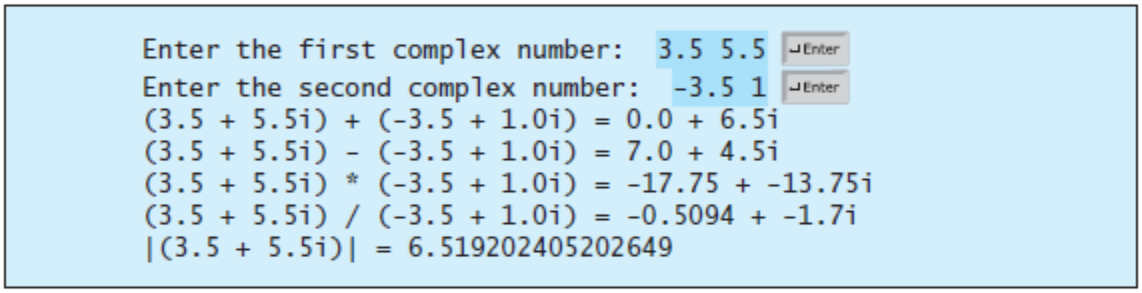
- Override toString method for returning a string representation for a complex number. (The toString method returns (a + bi) as a string. If b is 0, it simply returns a.)
- Your Complex class should also implement the Cloneable interface.

Provide three constructors

- Complex(a, b),
- Complex(a), and
- Complex().

Complex() creates a Complex object for number 0 and Complex(a) creates a Complex object with 0 for b. Also provide the getRealPart() and getImaginaryPart() methods for returning the real and imaginary part of the complex number, respectively.

Write a test program that prompts the user to enter two complex numbers and displays the result of their addition, subtraction, multiplication, division, and absolute value. Here is a sample run:



```
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 + -13.75i
(3.5 + 5.5i) / (-3.5 + 1.0i) = -0.5094 + -1.7i
|(3.5 + 5.5i)| = 6.519202405202649
```

Task 2:

Write a method that returns a random number from a

list of numbers passed in the argument. The method header is specified as follows:

public static int getRandom(int... numbers).

Note that the program should also contain a main method as well.

Task 3.

Design a class named Triangle that extends GeometricObject. The class contains:

Three double data fields named side1, side2, and side3 with default values 1.0 to denote three sides of a triangle.

A no-arg constructor that creates a default triangle.
A constructor that creates a triangle with the specified side1, side2, and side3.
The accessor methods for all three data fields.
A method named getArea() that returns the area of this triangle.
A method named getPerimeter() that returns the perimeter of this triangle.

A method named toString() that returns a string description for the triangle. (You can also use the program given in the class, as a GeometricObject if you need it)

The toString() method is implemented as follows:

return "Triangle: side1 = " + side1 + " side2 = " + side2 + " side3 = " + side3;

Implement the followings: The classes Triangle and GeometricObject. (As I mentioned, you do have access to the GeometricObject alredy, make your assumption if there are any and change the code)Write a test program that prompts the user to enter three sides of the triangle, a color, and a Boolean value to indicate whether the triangle is filled. The program should create a Triangle object with these sides and set the color and filled properties using the input. The program should display the area, perimeter, color, and true or false to indicate whether it is filled or not.