Assignment Triangle CSIS-1400

Learning Objective:

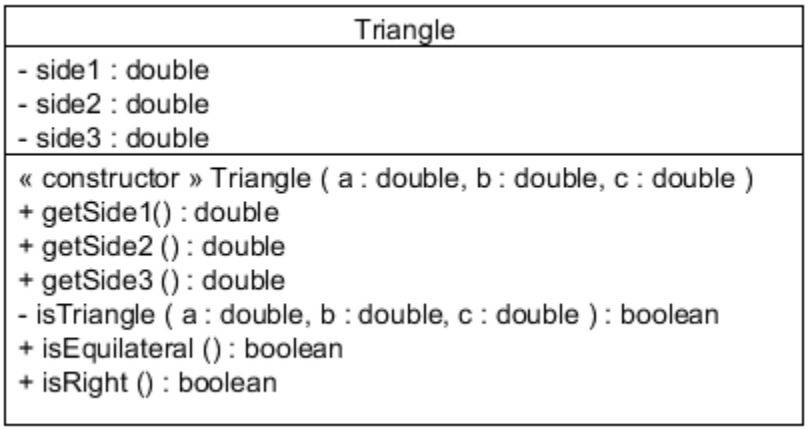
- Declare constructors and methods with multiple parameters
- Use the `if-else` statement to provide a second path of execution
- Practice the use of logical operators `(&&, ||)`
- Formulate algorithms that return the correct results for a described methods

Description:

In this assignment you'll write a class that represents a triangle.
Create a new project that includes 2 files: Triangle.java and TriangleApp.java
Include a comment on top of each source code file that include the author, the date the code was written, and the assignment number

Ad Triangle:

The class Triangle should implement the members exactly as specified in the class diagram below.
It has 3 attributes that represent the three sides of a triangle, it has a constructor with three parameters, and it has 6 methods. 3 of them are get methods that expose the values of the triangle sides (no set methods; triangles of this class cannot be resized once they have been created). The other three methods are: isTriangle, isEquilateral, and isRight.
Read Ch 5.8 to learn about the use of logical operators `(&&, ||)`. Then use them to implement the methods above.
Hint: The class Triangle should include **no** print statements.



Ad **`Constructor:`**

The class Triangle has one constructor that takes 3 parameters to initialize the three sides.
The argument values should only be used if they for a valid triangle. Otherwise all sides should be initialized with 1d.
To find out whether 3 sides make a valid triangle the constructor should call the private method isTriangle (see below)

FYI: Typically you might throw an exception in such a situation. We haven't covered exceptions yet, so for now we just use default values (in our case 1d)

Ad **`isTriangle:`**

The method isTriangle has three parameters . It determines whether the three parameter values make a triangle and returns true or false respectively. (E.g. 3 sides of length 2, 3, and 7 do **not** make a triangle, however, 3,3, and 4 do)
This method isTriangle will be called by the constructor (see above).
Caveat: arguments passed to the method could be zero or negative.

Ad **`isEquilateral:`**

The method isEquilateral does not need any parameters. It takes the values of the fields to find out whether the given triangle happens to be an equilateral triangle. It returns the corresponding boolean value.

Ad **`isRight:`**

The method isRight also takes the field values to find out whether the given triangle is a right triangle. It returns the corresponding boolean value.
Caveat: we do not know which of the 3 sides is the longest

Ad TriangleApp

1. Read in three sides from the user and create an instance of type Triangle based on the user provided values.
   The values should be read in next to the prompts and each prompt is displayed in a separate line (see output)
2. Display information about the triangle (NOT the local variables). First there is the word Triangle followed by the field values side1, side2, and side3 comma separated in parenthesis.
   If the triangle is right or equilateral the corresponding message is displayed
   If the triangle is neither right nor equilateral no additional message is displayed (see output)

Output samples:

output1:
```
Side1: 3
Side2: 5
Side3: 4

Triangle(3.0, 5.0, 4.0) is right
```

output2:
```
Side1: 4.5
Side2: 4.5
Side3: 4.5

Triangle(4.5, 4.5, 4.5) is equilateral
```

output3:
```
Side1: 4
Side2: 4
Side3: 5

Triangle(4.0, 4.0, 5.0)
```

output4:
```
Side1: -2
Side2: 2
Side3: 1

Triangle(1.0, 1.0, 1.0) is equilateral
```

Turning in:

Zip up your project (include all source code files)

Turn it in via Canvas.

output4:
```
Side1: 1
Side2: 2
Side3: 3

Triangle(1.0, 1.0, 1.0) is equilateral
```