Assignment 10 (Bonus)

Exercise 1 (5pt)

Write a `Book` class that contains the title and author of a book. Implement the `BookShelf` class that stores a `ArrayList` of books.

Add a method to sort the books (the sorting should first follow the title then the author), and a method to find a particular book.

Exercise 2 (7pt)

Write the `Complex` number class. The class must extend `java.lang.Number`.
The conversion to the basic types (e.g., float, double, …) should be done on the real part.

It should provide

- An empty constructor (initialise the number to 0)
- A constructor with one parameter (real number)
- A constructor with 2 parameters (real and imaginary part)

- `re()` that returns the real part
- `im()` that returns the imaginary part
- `conjugate()` that returns the complex conjugate

- addition, subtraction, and multiplication, with complex numbers or doubles (see https://en.wikipedia.org/wiki/Complex_number#Elementary_operations)

Exercise 3 (5pt)

Implement the `GeometricObject` class (or reuse the one from previous assignment), it should provide

- The `x,y` position

- An empty constructor that initialises it at 0,0
- A constructor that takes the 2 fields

- A method `getArea()` that returns the area
- A method `getPerimeter()` that returns the perimeter

Implement also 3 subclasses (add all necessary fields and methods)

- Circle
- Rectangle
- Triangle

Implement setters for the radius width/height and base/height for the 3 geometric figures and throw the appropriate exception if the values are invalid (e.g., negative).

Exercise 4 (6pt)

Implement the `Triangle` class. The constructor should take 3 arguments, corresponding to the length of the sides. In a triangle, the sum of any two sides is grater the the other side.
Create the `IllegalTriangleException` class and make the constructor of the `Triangle` class throw such exception if the object violates the rule.

Example:

`new Triangle(10, 1, 1)`, violates the rule since 1 + 1 < 10

Exercise 5 (7pt)

Write a java program that counts the number of characters (with and without spaces), words, and lines in a file. The filename should be passed as a command-line argument.

Example of call:

```
> java FileCounter file.txt
```

Output

```
1234 characters
1001 characters ignoring spaces
150 words
30 lines
```

Instructions

The solution of the exercises must be provides as a **java** (for the code, do not submit class files), **png** (for eventual screenshot), and **pdf** (for eventual text) files. The **files must be zipped** together before upload.

Assignments not respecting these instructions will be ignored.