Programming Tutorial (Advanced)
Bauhaus-Universität Weimar

Task 1

Let's do some geometry. Implement the following things:

1. A class `Point2D`. A `Point2D` has an *x-coordinate* and a *y-coordinate*.

2. A class `Line2D`. A `Line2D` contains two objects of `Point2D` for start and end of the line. In addition, `Line2D` should contain a method `getLength()` which returns the length of the line. If the line is parallel to one of the axis, computing the length is trivial. Otherwise you can use the pythagorean theorem to compute the length.

3. An interface `iForm`. `iForm` contains methods for computing the *area* and *circumference* of a geometric object.

4. A class `Rectangle` implementing `iForm`. Its constructor should get as input two objects of `Line2D`.

5. A class `Triangle` implementing `iForm`. Its constructor should get as input three objects of `Line2D`. Implement a method `isValid()` to check if the lines given as input form a valid triangle. Additionally, you will need a value h to be able to perform all desired computations. Obtaining h out of the lines of a valid triangle works as follows:

$$\alpha = acos(\frac{0.5 * len(a)^2 - 0.5 * len(b)^2 + 0.5 * len(c)^2}{len(b) * len(c)})$$

$$\beta = acos(\frac{0.5 * len(a)^2 - 0.5 * len(b)^2 + 0.5 * len(c)^2}{len(a) * len(c)})$$

$$\gamma = -\beta - \alpha + 180$$

$$h = len(b) * sin(\gamma)$$

6. A class `Driver` containing a *main*-Method where you call all your methods and print the outputs into the console.

The following commands might be helpful:

- `Math.sqrt(double d)` = Square-root of d
- `Math.pow(double a, double b)` = a^b
- `Math.acos(double d)` = $acos(d)$
- `Math.sin(double d)` = $sin(d)$