CS1026a: Assignment 1

Due: Wednesday, October 9th, 9pm
Weight: 5%

Purpose:
To gain experience with

- Basic programming constructs in Java
- Writing and invoking methods in Java
- Using DrJava to create a complete program

Task:

In this assignment, you will write a **complete** program in Java that uses the Turtle class to draw shapes on the screen. Your task is to use combinations of squares, rectangles and equilateral triangles to draw a figure (house, car, spaceship, cat, dog, parakeet, airplane, etc) on the screen. You must use:

- At least 2 rectangles of different sizes
- At least 2 squares of different sizes
- At least 2 triangles of different sizes
- At least two colors other than black
- At least two pen widths

You can make the figure look anyway you like so long as it uses these shapes and meets the requirements. Other shapes may be used in your house. Be creative!

Functional Specifications:

1. It is assumed that you have added the method drawSquare with the header

public void drawSquare(int width)

to your *Turtle.java* already. It must be as described in the Lecture Notes: your turtle must start at the top left hand corner, facing north, and finish in the same place, facing north.

2. You will add two new methods to the Turtle class in your file *Turtle.java:*

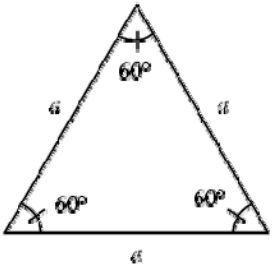
- A method drawRectangle with the header:

  public void drawRectangle(int width, int height)

  that draws a rectangle on the screen with the width and height passed in as integer parameters. Your turtle must start at the top left hand corner, facing north, and finish in the same place, facing north.

- A method drawEquilateral(int length) with the heather:

  public void drawEquilateral(int length)



that draws an equilateral triangle on the screen, with the length of each side passed in as the integer parameter. An equilateral triangle has all three sides of the same length and all three internal angles of 60 degrees. Your turtle must start at one vertex of the triangle, facing north, and finish at the same point, facing north. An equilateral triangle looks like this:

3. Write a complete Java program that makes use of your methods drawSquare, drawRectangle and drawEquilateral to draw a house on the screen.

4. Your Java program will be in a class called TurtleArt as shown below. Using the definitions pane of DrJava, you will add Java statements that actually draw the required options. The program will be stored in a file called *TurtleArt.java*.

```
import java.awt.Color;
public class TurtleArt
{
        public static void main (String[] args)
        {
                /* Insert your code here */
        }
}
```

5. The figure will be in its own world, and you may use as many turtles as you like to draw it.

6. You may use any of the methods of the Turtle class that are already defined. You may draw lines as well as the shapes. You may write other methods to help you if you like.

7. You can be as creative as you want, although there are no marks for "degree of difficulty".

*****Important! You must follow all the specifications above for your program*****

Non-functional Specifications:

1. Include brief comments in your code that explain what each method is doing.
2. Assignments are to be done individually and must be your own work. Software will be used to detect cheating.
3. Use Java conventions and good Java programming techniques, for example:
   i. Meaningful variable names
   ii. Conventions for naming variables and constants
   iii. Readability: indentation, white space, consistency

What You Will Be Marked On:

1. Functional specifications:
   - ✓ Are the required methods written according to specifications?
   - ✓ Do they work as specified? Pay attention to details!
   - ✓ Are they called as specified?
   - ✓ Are the objects drawn according to the specifications?
2. Non-functional specifications: as described above
3. Assignment submission: via the OWL, though the assignment submission in OWL.