Turtle Class Explorer Lab Report

<table>
<thead>
<tr>
<th>Name</th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>ID</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Lab Section</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**Task 1: Understanding the behavior of Turtle class**

1. Try the **step**, **turn**, and **setPaint** buttons and describe what they do. Draw a simple drawing (a square, a simple polygon, ...) with more than one color and copy the code from the **Console** into your report.

2. Experiment with **showTurtle**, **hideTurtle**, **setPenState(PEN_UP)**, and **setPenState(PEN_DOWN)** functions and describe what they do.

3. Try the **Triangle**, **Square**, and **Hexagon** buttons. Study the code that performs the drawings. Now create a regular pentagon and include a snapshot and the code in your report.

4. Create the following drawing and include the code and a full snapshot in your report. You always get the code by copying the transcript in the Console. Hint: Plan your work first, write down what you want to do, the order in which you want to do things. Remember to put the pen down or up as needed.

**Task 2: Create your own Turtle drawing**

5. Write down the sequence of the function calls that need to be made to create your drawing. Include your drawing and design in your lab report. These can be sketched by hand.

6. Run the program again and check that it creates your drawing correctly. Include both the code and a snapshot in your report.
Task 3: Read Turtle class code

Include the highlighted copy of the TurtleX.java with your lab report.

7. Pick two static member data and two static member functions and write down what you think is their purpose.

8. Highlight in yellow the names and types/classes of all non-static member data. Explain the reason for selecting each type of class for each member data item.

9. Highlight in another color the headers of all get non-static member function - specify the color in your report.

10. Highlight in a fourth color the headers of the remaining non-static member functions and the constructor header.