

TCSS142 Introduction to Object-Oriented Programming

Programming Assignment 2

DUE: Monday, July 7, 2014 by 12 midnight

In our previous programming assignment (1b), we learn how to reduce redundancy by utilizing static methods and identifying the structure of the shapes to be drawn and also the redundancy within the shapes. We will use this same strategy to complete the following programming assignment. However, with the new tools we have learned this week, we will also implement the use of print statements, for loops, and a class wide named constant (Java's "final") that will be used to determine the displayed size of the image drawn.

The image to be produced is that of the world famous Seattle Space Needle (just up the road a piece!). Your program should produce this image in varying sizes base upon your class constant value (Remember, each time you want to adjust the size, you will need to edit your program file to update the value that is assigned to the class constant, save this change by clicking the Save button, re-compile, and then run the program.

See pages 2 - 4 of this assignment (programming description) for illustrations of varying sized Space Needles.

The majority of your output will be produced (one character at a time) through the use of for loops. This is an expectation that will be a major portion of your grade

Your final program should include a brief javadoc statement to clarify each method such as:

```
/**
 * needleRoof method outputs the roof over the restaurant
 * This method can also be used to display the Space Needle base.
 */
public static void needleRoof ( ) {

    etc.
```

Because loops can be confusing, you should document each loop with 1 or 2 lines of non-javadoc just above the loop.

You should also include multiline non-javadoc documentation at the beginning to identify the course, file name, programming assignment number, the due date, and the instructor. This is followed by a line of space and then just before the class declaration, javadoc to give a brief description of what the program does, an author tag for your name and the current date as a javadoc version tag, such as:

```
/*
 * Course:          TCSS142 – Introduction to Object-Oriented Programming Summer 2014
 * File Name:       SeattleSpaceNeedle.java
 * Assignment:      2
 * Due Date:        July 7, 2014
 * Instructor:      Mr. Schuessler
 */

/**
 * This program draws an image of the Space Needle...
 *
 * @author your name
 * @version 2014 July 5
 */
```

Your main method should not have any print or println statements, only calls to the various methods which produce the portions of the Space Needle. Don't forget to look for repeated patterns!

Save your program as SeattleSpaceNeedle.java **...this implies the name of your Class. Don't forget!**

Grading:

Program Compiles:	30%
Correct Output:	20%
Documentation (text and proper/consistent indentation):	20%
Proper use of methods to reduce redundancy, use of a class constant for sizing the image, and frequent use of for loops to print out repeated characters:	<u>30%</u>
	100%

Good Luck and Start Early!

```
public static final int SIZE = 3;
```

The figure is composed of several distinct parts:

- Hat:** A wide, flat-topped hat made of slashes (/ and \) and colons (:).
- Face:** A simple face with two vertical lines for eyes and a small vertical line for a nose.
- Shirt:** A rectangular body with a border of percent signs (%) and a central area of horizontal lines.
- Bottom:** A wide, flat base made of slashes (/ and \) and colons (:).

SIZE of 2:

SIZE of 1:

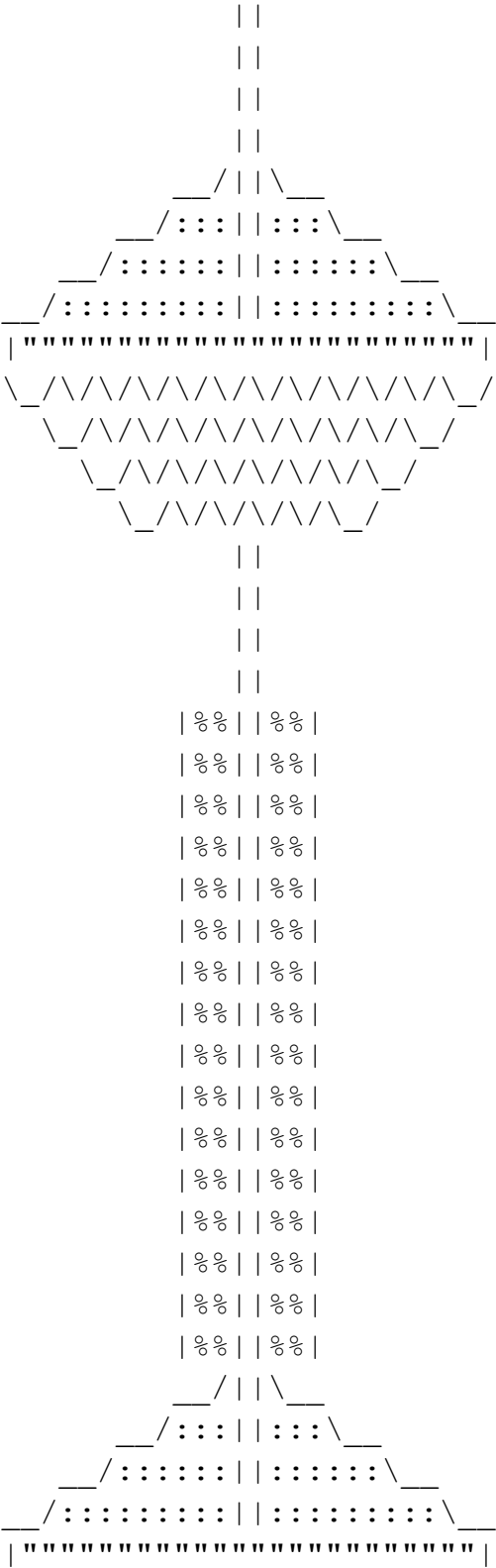
```
//Look at that little feller. He thinks he's so big!
```

The diagram shows a 4x4 grid of circles. The symbols inside the circles are as follows:

- Row 1: Top-left circle has a vertical bar. Top-right circle has a vertical bar. Second circle from left has a slanted line. Third circle from left has a slanted line.
- Row 2: All four circles contain the Korean character 'ㅇ' (ong).
- Row 3: All four circles contain the Korean character 'ㅇ' (ong).
- Row 4: All four circles contain the Korean character 'ㅇ' (ong).

continued--->

SIZE of 4:



continued--->

SIZE of 6:

