

# TCSS143

## Fundamentals of Object-Oriented Programming-Theory and Application

### Programming Assignment 1

DUE: Tuesday, September 30, 2014 by 11:59 p.m.

The purpose of this programming project is to demonstrate understanding of methods (calling, parameters, and return values), the for loop, primitive data types, standardized documentation, and basic java program structure.

Write a program that prompts the user to enter the population values of 5 towns. After the 5 populations have been entered, the program should display a horizontal bar graph comparing each population. The bars in the graph should be made of asterisks "\*" where each asterisk equals 1,000 people. An example interaction is as follows (**bold print is what the user enters**):

```
Enter the population of town 1: 12549 [Enter]
Enter the population of town 2: 3578 [Enter]
Enter the population of town 3: 5023 [Enter]
Enter the population of town 4: 1732 [Enter]
Enter the population of town 5: 10522 [Enter]
```

POPULATION GRAPH:

```
Town 1: *****
Town 2: ***
Town 3: *****
Town 4: *
Town 5: *****
```

#### DETAILS

Name your class Assign\_1 (Save as Assign\_1.java).

You are required to follow the rules for correct documentation usage and identifier names as described in the "Documentation Requirements.pdf" file posted on the Canvas website for 143.

You are required to create and use at least 2 methods: **getPopulation** & **drawPopulationBar**

**getPopulation** should receive an integer used to represent or identify the town on which a population is to be entered and a scanner to System.in. It should prompt the user for this town and return the integer value the user enters. As an example, one of the calls to this method might be:

```
town2Population = getPopulation(2, console); // Enter the population of town 2: 3578 [Enter]
```

**drawPopulationBar** should be passed both the integer representing the town and its population value. It should display the town number and the bar of asterisks based on the population value as seen above. As an example, one of the calls to this method might be:

```
drawPopulationBar(3, town3Population); // Output would be: Town 3: *****
```

Without the use of arrays, you will have at very least 5 variables for the 5 populations, 5 separate calls to getPopulation and 5 separate calls to drawPopulationBar. If you choose to solve this using an array, your reward will be much less effort in typing 😊 and let's throw in 1% added to your score if your code reflects only one line for each method call.