CSC240_Lab02: Writing Java Static Methods

Points: **100** points

Objective:
This assignment will demonstrate your knowledge of static methods and Math class.

Instructions:

- Complete the class attached named **StaticMethodsDemo.java**
- This class has the `main` method and 17 other static methods bodies to be called from inside `main`.
- Therefore, once we run your code, we should see all outputs based on calling those 17 methods.
- Be sure to document your code (add comments on top of your java file).
- In the comments, add your name, date, course, homework number, and statement of problem.
- Once you are done, upload the complete **StaticMethodsDemo.java** through D2L.

Static methods definitions:

- (Each part worth 6 points, 5 points for the answer, and 1 point for adding comments).
- Adding Comments means adding Pre-conditions and Post-conditions

[1] **Write a method called `countsGuests` for the following header. The method should return a welcome message that includes the user's name and visitor number. For example, if the parameters were "Joe" and 5, the returned string would be "Welcome Joe! You are visitor #5."**

[2] **Write a method called `alarm` that prints the string `"Alarm!"` multiple times on separate lines. The method should accept an integer parameter that specifies how many times the string is printed. Print an error message if the parameter is less than 1.**

[3] **Write a method called `sum100` that returns the sum of the integers from 1 to 100, inclusive.**

[4] **Write a method called `maxOfTwo` that accepts two integer parameters and returns the larger of the two.**

[5] **Write a method called `sumRange` that accepts two integer parameters that represent a range. Issue an error message and return zero if the second parameter is less than the first. Otherwise, the method should return the sum of the integers in that range (inclusive).**

[6] **Write a method called `larger` that accepts two floating-point parameters (of type double) and returns true if the first parameter is greater than the second, and false otherwise.**

[7] **Write a method called `countA` that accepts a `String` parameter and returns the number of times the character `'A'` is found in the string.**

[8] **Write a method called `evenlyDivisible` that accepts two integer parameters and returns true if the first parameter is evenly divisible by the second, or vice versa, and false otherwise. Return false if either parameter is zero.**

[9] **Write a method called `average` that accepts two integer parameters and returns their average as a floating point value.**

[10] **Write a method called `average` to accept three integer parameters and return their average.**

[11] **Write a method called `average` to accept four integer parameters and return their average.**

[12] **Write a method called `average` to accept five integer parameters and return their average.**

[13] **Write a method called `multiConcat` that takes a `String` and an integer as parameters. Return a `String` that consists of the string parameter concatenated with itself `count` times, where `count` is the integer parameter. For example, if the parameter values are `"hi"` and `4`, the return value is `"hihihihi"`. Return the original string if the integer parameter is less than 2.**

[14] **Write a method called `isAlpha` that accepts a character parameter and returns true if that character is either an uppercase or lowercase alphabetic letter.**

[15] **Write a method called `reverse` that accepts a `String` parameter and returns a string that contains the characters of the parameter in reverse order. Note that there is a method in the `String` class that performs this operation, but for the sake of this exercise, you are expected to write your own.**

[16] **Write a method called `isIsoceles` that accepts three integer parameters that represent the lengths of the sides of a triangle. The method returns true if the triangle is isosceles but not equilateral (meaning that exactly two of the sides have an equal length), and false otherwise.**

[17] **Write a method called `randomInRange` that accepts two integer parameters representing a range. The method should return a random integer in the specified range (inclusive). Return zero if the first parameter is greater than the second.**