CS2123 Data Structures
Assignment 1: Runtime Table

Completing the Program (14 points)

This program prints a table of runtimes (these are displayed in seconds) for given functions on arrays. Here's some example tables showing the difference between the best case and worst case runtime of insertion sort:

```
Insertion Sort (sorted order)
Test size Test #1  Test #2  Test #3  Test #4  Average  Increase
4000      0.004    0.004    0.004    0.005    0.004    N/A
8000      0.006    0.006    0.006    0.006    0.006    1.412
16000     0.009    0.009    0.010    0.009    0.009    1.542
32000     0.018    0.022    0.018    0.018    0.019    2.054
64000     0.036    0.036    0.037    0.035    0.036    1.895

Insertion Sort (reverse sorted order)
Test size Test #1  Test #2  Test #3  Test #4  Average  Increase
4000      0.029    0.017    0.016    0.017    0.020    N/A
8000      0.057    0.057    0.057    0.059    0.058    2.911
16000     0.211    0.211    0.211    0.211    0.211    3.670
32000     0.824    0.824    0.820    0.821    0.822    3.897
```

Note your runtimes will vary a bit since the test data is randomly generated.

The program tests different array sizes to establish a relationship between input size and runtime. It tests each array size multiple times and then takes an average of the runtimes. We also output how much the average runtime increased relative to the previous average. This is calculated by dividing the current average by the previous average (output "N/A" for the first increase value).

You are given a partial implementation in the file "DriverRuntimeTable.java". Specifically you are tasked to complete the functions below the heading "Functions for finding and printing runtime" (there will also be a "TODO" next to anything you need to change/complete). No other functions should be changed.

Using the Program (6 points)

After you have the program completed, you should use it to help determine the asymptotic runtimes of the three mystery functions (i.e., mysteryRuntime1, mysteryRuntime2, mysteryRuntime3).

Be sure to also examine the code of the mystery functions to confirm/improve your estimations.

Fill in the following table in the provided java file:

```
/*
TODO: Give your asymptotic estimates for the runtimes of the following 3 functions:
```

```
mysteryRuntime1:  O(   )
mysteryRuntime2:  O(   )
mysteryRuntime3:  O(   )
*/
```

Deliverables:

Your solution should be submitted as "DriverRuntimeTable.java". Be sure to fill in the runtimes described above. To receive full credit, your code must compile and execute.

Upload this file to Canvas under Assignment 1. **Do not zip your file**.