CO322: Data Structure and Algorithms

Lab1: Simple sorting algorithms

Aim: Aim of this laboratory is to understand the time complexity of simple sorting algorithms.
Objective:

- Check your ability to implement simple sorting algorithms using nested loops.
- Check your coding skills. (indentations, comments, use of proper variable names etc.).
- Check your analytical abilities.

This laboratory you can do in your leisure and submit what is requested.
Deadline: 3rd February 2015

Work:
Here you are supposed to measure and analyze the performance of three simple sorting algorithms: *Insertion sort*, *selection sort* and *bubble sort.* For this there are 3 main tasks:

Task 1: Implement *insertion sorting*, *selection sorting* and *bubble sorting* using Java. You should implement them as methods using the provided skeleton code in `CompareSorting.java` file for your implementation.
You can use the provided `postCondition` method to test whether your sorting is correct. (see what Wiki has to say about postconditions http://en.wikipedia.org/wiki/Postcondition (last visited 25/1/15))

Task 2: generate 1000-400000 random sets of integers (between 0-1000) in steps of 10000.
The following website discuss about generating random numbers in Java:
http://stackoverflow.com/questions/5887709/getting-random-numbers-in-java (last visited 25/1/15)

Task 3: run each algorithm on the same data set and calculate sorting time. Here the Java `System.currentTimeMillis()` may help you.
http://download.oracle.com/javase/1.5.0/docs/api/java/lang/System.html#currentTimeMillis()

Task 4: plot the data using `matlab`, `gnu plot` or any other tools. And prepare a small report including the graph, the observations and compare the results you got with the theoretical complexity values.

Discussion: in terms of the theoretical analysis all the above algorithms are $O(n^2)$. Does the numbers you measure agree? Suppose for a one data set you run an algorithm; say the bubble sort; twice will you see the same time? If not why? What if you take the average by repeating the experiment? Your report should consider these aspects.

Submission: Please submit the modified `CompareSorting.java` file and the report including the graph of your plot and observations to CMS (report is preferred in pdf format). **Strictly no late submissions.**