Assignment 3 – Experimentation and Efficiency

Assigned: Tuesday February 5, 2013
Due: Tuesday February 12, 2013 11:59 PM
Assignment Type: Individual

Problem Description

This assignment is a departure from the previous assignments in that the focus is not on program construction, but rather on analysis, experimentation, and applying the scientific method. The provided zip file for this assignment (A3.zip) contains the following files.

- Clock.java - A Java class that you will use to measure running time.
- RunningTime.class - A Java class for which you have to experimentally discover the running time of a method.
- RunningTimeClient.java - A Java class that illustrates basic timing relevant to RunningTime. This file is for illustration purposes only. You can modify it if you wish, or you can create your own.
- SortTrials.class - A Java class for which you have to experimentally identify the five sorts that are implemented.
- SortTrialsClient.java - A Java class that illustrates basic calls to public methods of SortTrials. This file is for illustration purposes only. You can modify it if you wish, or you can create your own.

There are two parts to this assignment.

Part 1 - Experimental discovery of running time

You must apply the scientific method to empirically discover the big-Oh running time of a given method. The method is encapsulated in `RunningTime.timeTrial(int N, int seed)`. The parameter `N` is used to model the problem size. The parameter `seed` is used to select a particular private method, internal to the class, to actually perform the work. Thus, the value of `seed` will differentiate your results from those of your classmates (at least some of them, anyway). The seed you are to use for your assignment is posted as a comment in your assignment grading area.

You must write a lab report that describes your process in discovering the big-Oh running time of `RunningTime.timeTrial()`. The lab report must be well written and exhibit a high degree of professionalism. Your report must contain the source code that you wrote and the data that was generated. The data must be described in both tabular and graphical formats.

You must model your lab report on the examples provided at the following two sites.

- http://www.writing.engr.psu.edu/workbooks/laboratory.html
- http://www.ncsu.edu/labwrite/res/labreport/res-sample-labrep.html

Part 2 - Experimental identification of sorting algorithms

You must apply the scientific method to empirically identify the sorting algorithms implemented in five public methods of `SortTrials.java`. The methods are simply named `sort1`, `sort2`, `sort3`, `sort4`, and `sort5`. For each method you must decide, based on your experimental results, which of the following sorting algorithms the method is implementing: mergesort, quicksort with no randomization, randomized quicksort, insertion sort, or selection sort.

You must write a lab report that describes your process in identifying the sorting algorithms. The lab report must be well written and exhibit a high degree of professionalism. Your report must contain the source code that you wrote and the data that was generated. The data must be described in both tabular and graphical formats.

You must model your lab report on the examples provided at the following two sites.

- http://www.writing.engr.psu.edu/workbooks/laboratory.html
- http://www.ncsu.edu/labwrite/res/labreport/res-sample-labrep.html

Assignment Submission

Part 1 and Part 2 must each have its own separate lab report. Each lab report must be in PDF format. You must create a zip file that contains both these PDF files, along with all the source code files that you created. You must upload this single zip file to Canvas no later than the date and time indicated.