CS 456: Advanced Algorithms
Programming Assignment #01
Total Points: 130

Assigned Date : Tuesday, September 15, 2015
Due Date : Tuesday, September 29, 2015 @ 05:59:59 p.m.

Update: September 25, 2015

You are no longer required to submit your code as part of your solution. Correspondingly, I've modified the assignment lingo to reflect this change. The changes are all logistics related; the overall programming task remains the same.

Overview

For your first programmming assignment, you will implement two versions of **Quicksort** – the regular quicksort and the randomized quicksort (see chapter 7 of the main textbook) – and empirically validate their asympototic runtime behavior for *best case*, *average case*, and *worst case* using **computer generated results**. More specifically, you are expected to think about and address the following questions:

a. At what size n_0 does your implementation start to exhibit asymptotic complexity?
b. What is the characteristic of your input required to generate *average* complexity. How about best case and worst case scenarios? How do you plan to generate the appropriate input?
c. How does the measured run time correspond to the abstract complexity analysis using operation counting (as discussed in class)?
d. How to create your test driver so that it exercises your sort programs.
e. How to create the sorting class so that it will be extensible and reusable for future projects.

Instructions

- This is an individual assignment. **Do your own work**.
- **Start early!!**
- **Take backups of your code often!!**.

- You may use any programming language of your choice. However, you **must** make sure that your code compiles and runs on a typical Linux machine. Absolutely **DO NOT** include executables with your submissions.
  - For this first assignment, you are **NOT** required to submit any code as part of your solution. If for any reason, however, I need to see the code you've used to generate your results, you must be able to demonstrate an execution of your code on a Linux machine. Moreover, if you are being asked to submit code in future assignments, they must be Linux compatible/friendly. Thus, I advice you to program and generate results on a Linux machine, even for the first assignment.
- The report part of your solution must be produced using a word processor. I highly recommend **Latex**. Any figures, graphs, plots, etc., should also be produced using appropriate computer applications. Graphs/plots should be properly labeled. Your final report should be in **PDF** format. No exceptions.
- Follow a good coding standard. Use the Google C++ coding standard found here `http://goo.gl/1rC1o`, if you don't already follow one.
- Total points: [**100 points**]

Deliverables

The due date of this assignment is **Tuesday, September 29, 2015 @ 05:59:59 p.m.** A dropbox will be opened for submission on Moodle before the due date. A complete solution comprises of:

- A report that includes the followings:
  - Motivation and background of the experiment [**10 points**]
  - Pseudocode with Invariants and pre/post conditions. [**20 points**]
  - Testing Plan and Test Results [**20 points**]
  - A correctness proof of your programs [**20 points**]
  - Problems Encountered/Key insights [**10 points**]
  - Justification of your observations. You must be able to justify and/or argue the empirical asymptotic behavior you are observing [**30 points**].
  - Conclusion and performance comparisons [**20 points**]