CS 456: Advanced Algorithms
Programming Assignment #02
Total Points: 150

Assigned Date : Tuesday, October 27, 2015
Due Date   : Tuesday, November 10, 2015 @ 05:59:59 p.m.

Overview

For your second programmming assignment, you will implement three versions of the **Convex Hull Problem** – Graham Scan (see p. 1029 of the CLRS book), Jarvis March (see p. 1037 of the CLRS book), and QuickHull (see p.195 of the Levitin book) – and empirically validate their asymptotic runtime behavior using **computer generated results**. The convex hull algorithms run at different complexities with one of the algorithms even exhibiting a different complexity based on average case and one on worse case. 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 <u>worst case</u> complexity of the quickhull algorithm?
c. How does the measured run time correspond to the theoretical asymptotic bounds?

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.
- 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.

Input Specification

- Your code must be capable of handling two types of input:

    1. Input read from a file
    2. Randomly generate a graph of a given size

    This option should be prompted to the user immediately upon execution. If file is selected, prompt user for filename. If random is selected, prompt user for the set size.

- For testing purposes, two files – **small.txt** and **large.txt** – are provided herewith. In addition, You are required to generate a file named **worstcase.txt** that contains a set used to demonstrate worst case time complexity of the quickhull algorithm. The given files will be used to test the accuracy of your program. It would be beneficial to test these files on your program beforehand.

- All input files should follow a format that has each line containing only one point with the x-coordinate listed first followed by a space and the y-coordinate.

Measurements

- After a set is read, either through a file or random generation, it is important that the same set is ran by all three algorithms. This is important because the runtime of the jarvis march is dependent on the location of the points as well as the number of points. It will be important to track the number of input points as well as the number of output points.

Output Specification

- For output, generate a file named **set[numberofpoints].txt**. The top three lines should include the times of the three algorithms followed by the output set in the same format as the input set. e.g. 500 points found in file set500.txt.

- Be sure to test enough different size sets to accurately graph behavior.

Deliverables

The due date of this assignment is <mark>Tuesday, November 10, 2015 @ 05:59:59 p.m.</mark> A dropbox will be opened for submission on Moodle before the due date. A complete solution comprises of:

- A report **in PDF format** that includes the followings:
    - Motivation and background of the experiment
    - Testing Plan and Test Results
    - Problems Encountered/Key insights
    - Justification of your observations. You must be able to justify and/or argue the empirical asymptotic behavior you are observing
    - Conclusion and performance comparison. Be sure to discuss the worst case quickhull file you created.

- A compressed tarball of the directory containing your source codes. Absolutely **<u>do not</u>** include any executables with your submission. To create a compressed tarball of the directory source, use the following command: `tar -zcvf name111-pr3.tar.gz source/`. Obviously, change the name to your last name and 111 to the last three digits of your SIUE ID.
- **worstcase.txt** your file used to demonstrate worst case behavior in quickhull

Tentative Grading Rubric

- Styling: Compiles, easy to read, properly indented, etc.., [**7 points**]
- Input Specification [**8 points**]
- Output Specification [**10 points**]
- Algorithmic Correctness [**69 points**]
    - Graham Scan: runs in $O(nlogn)$[**23 points**]
    - Jarvis's March: runs in $O(nh)$ [**23 points**]
    - Quickhull: runs average case $O(nlogn)$, worse case $O(n^2)$ [**23 points**]
- Report Specifications [**56 points**]

Sample input file

small.txt
```
6 6
7 9
5 0
0 1
4 9
7 4
8 5
2 6
8 3
8 1
```

Sample output file

set10.txt
```
Graham Scan: .0912 s
Jarvis March: .124s
Quick Hull: .065s
5 0
8 1
8 5
7 9
4 9
2 6
0 1
```

This assignment was designed by the course TA David Shovlin. Any questions regarding the assignment should be brought to Dr. Gamage's attention. TA will not honor any direct attempts by students to communicate with him regarding the assignment.