**Project 3 Part 3 (full recursive approach O(nlog^2n))**

Name: \_\_\_\_\_\_\_\_\_\_Michael Fatemi\_\_\_\_\_\_\_\_\_ Period: \_\_\_\_\_\_7\_\_\_\_\_\_

Date: \_\_\_\_\_\_\_\_\_\_11/18/2020\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Is your lab name l033?(lowercase L followed by digits 033) \_\_\_\_\_Yes\_\_\_\_\_\_

Did you created a class to store a point? \_\_\_\_\_Yes\_\_\_\_\_\_

Did you use a vector to store the points you generated? \_\_\_\_\_Yes\_\_\_\_\_\_

Did you use at least one iterator to traverse the vector you created? \_\_\_\_\_Yes\_\_\_\_\_\_

Did you sort using the sort method offered by C++? \_\_\_\_\_Yes\_\_\_\_\_\_

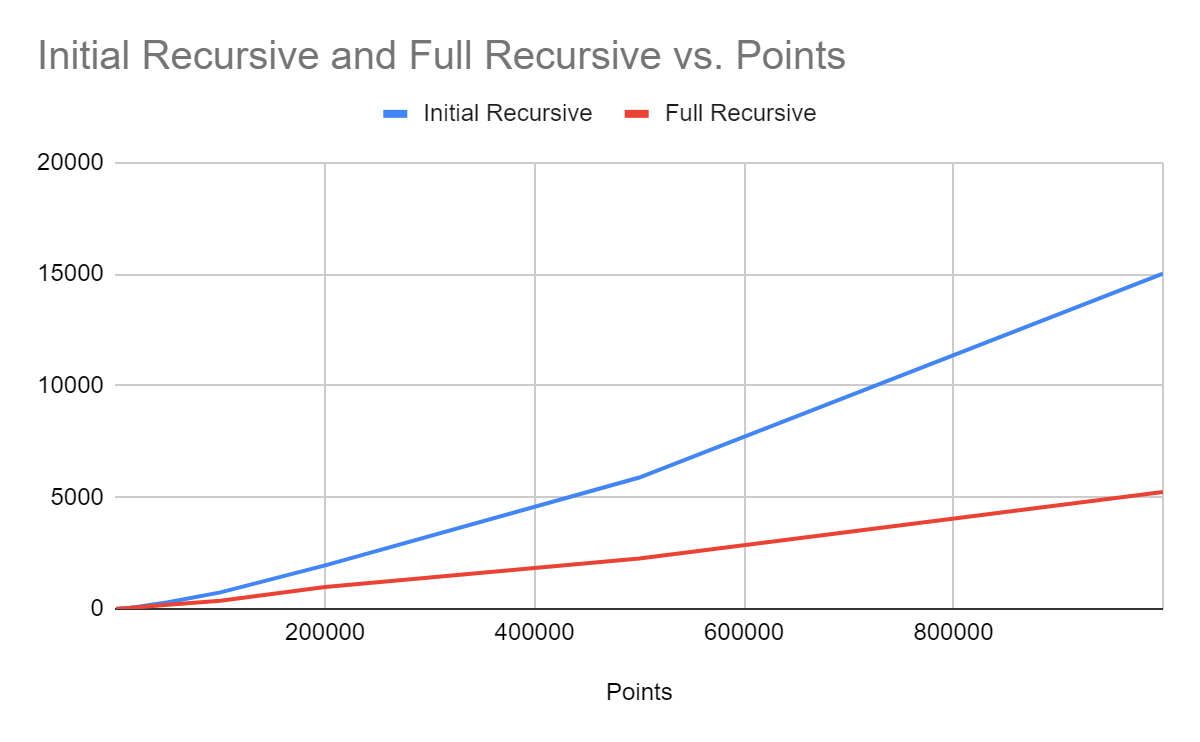
Did you use at least one iterator to traverse the vector you created? \_\_\_\_\_Yes\_\_\_\_\_\_

Does your main contain only 2 calls of: part2() and part3() (NO part1!!)? \_\_\_\_\_Yes\_\_\_\_\_\_

(in main you may also have the part to display results for the 2 methods and them also in the txt file)

1. **Paste here a clear picture of the graph that compares the running times of the “initial recursive” algorithm and “full recursive” algorithm versus number of points. (use 2 different colors for the 2 graphs, colors that can be visible even if you print in black and white). Each point on this graph should be an average of several runs for that size:**

Each of these results is the average of 10 runs.

****

1. **Paste here the content of the results.txt when you run your lab on the content of the file points10k.txt and points100k.dat**

**For 10k:**

Recursive x [1] 34ms (34/cycle)

(0.49999999999999989,0.50000000000000111) (0.49999999999999994,0.50000000000000111) 5.5511151231257827e-017

Recursive Optimized x [1] 33ms (33/cycle)

(0.49999999999999883,0.50000000000000033) (0.49999999999999889,0.50000000000000033) 5.5511151231257827e-017

**For 100k:**

Recursive x [1] 1271ms (1271/cycle)

(0.49999999999973127,0.49999999999889994) (0.49999999999975941,0.49999999999890893) 2.9545963426100509e-014

Recursive Optimized x [1] 480ms (480/cycle)

(0.49999999999973127,0.49999999999889994) (0.49999999999975941,0.49999999999890893) 2.9545963426100509e-014