**Project 3 Part 4 (randomized approach average O(n))**

Name: Anup Bagali Period: 7

Date: 12/13/2020

Is your lab name l034?(lowercase L followed by digits 034) yes

Did you created a class to store a point? yes

Did you use a vector to store the points you read? yes

Did you sort using the sort method offered by C++? yes

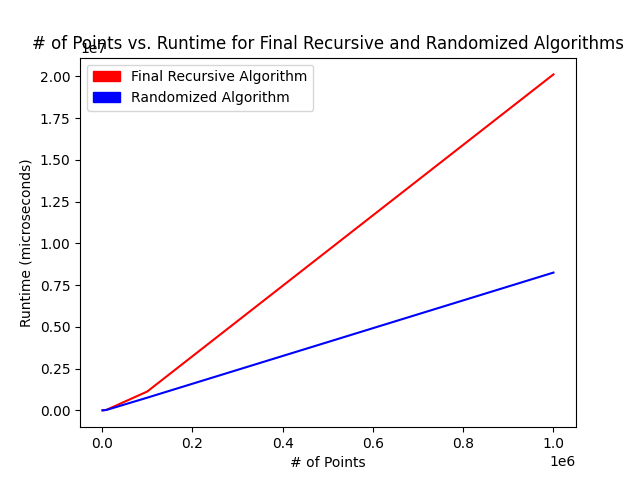
Does your main contain only 2 calls of: part3() and part4() (NO part1/2!!)? yes

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

Did you use an unordered\_map for your dictionary? yes

Did you implement the Knuth algorithm to randomize the points? yes

1. **Paste here a clear picture of the graph that compares the running times of the “full recursive” algorithm and “randomized” 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:**

****

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:**

**Full Recursive (0.50000000000000122, 0.49999999999999978) (0.50000000000000122, 0.49999999999999983) Distance: 5.55111512312578270e-17 Time: 29593  
microseconds   
Randomized (0.49999999999999883, 0.50000000000000033) (0.49999999999999889, 0.50000000000000033) Distance: 5.55111512312578270e-17 Time: 38656 mic  
roseconds  
  
  
For 100k:**

**Full Recursive (0.49999999999973127, 0.49999999999889994) (0.49999999999975941, 0.49999999999890893) Distance: 2.95459634261005094e-14 Time: 10356  
86 microseconds   
Randomized (0.49999999999973127, 0.49999999999889994) (0.49999999999975941, 0.49999999999890893) Distance: 2.95459634261005094e-14 Time: 615342 mi  
croseconds**