Lab 1 – Readme File

File Includes With This Project:

Main.java PointReader.java PRNG.java XComparator.java YComparator.java XYPoint.java ClosestPairDC.java ClosestPair.java autograder-results.txt control.txt readme.pdf

Algorithms Decisions & Project Issues:

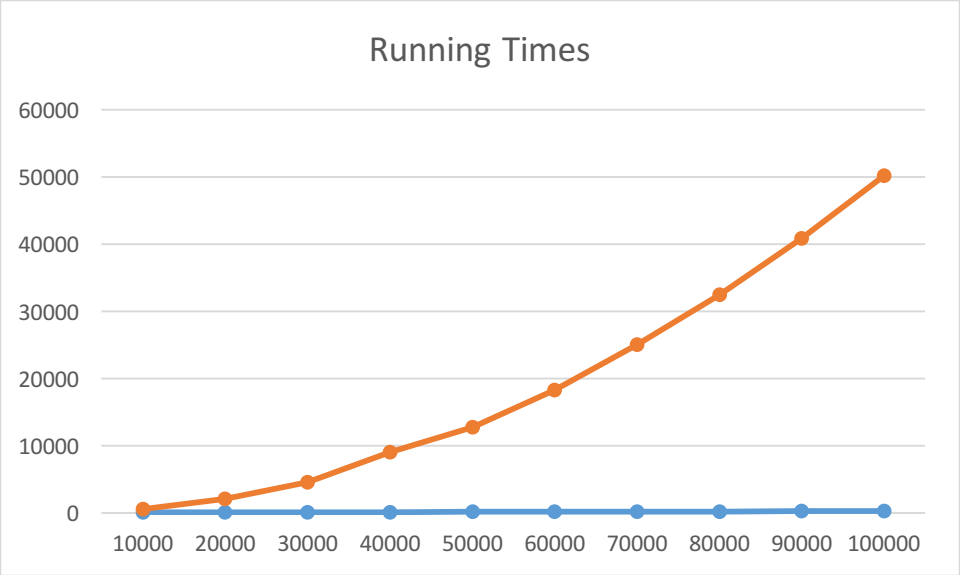
There are two algorithms to solve closest pair problem, one is called “Naive” while the other is “Divide-and-Conquer” algorithm.
We have implemented “Divide-and-Conquer” as followings: Under ClosestPairDC class, we create a new method named findClosestPairHelper(), which will do “Divide-and-Conquer” parts and return minimum distance as well as the closest pair. Under findClosestPair() method, we print the distance and the closest pair.
We have implemented “Naive” algorithm as followings: We compute the distance of two points for C(n,2) times, and compare them to get the minimum distance.

The most difficult part for me is the length of a new array. We I ran the project, it always showed “java.long.ArrayIndexOutOfBoundsException”. Fortunately, after debugging several times, I fixed the problem.

Part Two

Running Times on inputs as the following sizes:

nPoints	10000	20000	30000	40000	50000	60000	70000	80000	90000	100000
DC	38	77	102	113	128	123	172	183	217	224
NAÏVE	526	2040	4503	9021	12731	18233	25053	32498	40822	50140



1000 randomly generated inputs of size 50000

	Maximum Time	Minimum Time	Average Time	Variance
Same Inputs	154	22	31	490
Different Inputs	111	20	28	278

Part Three

Set k = 15000. Run “Divide-and-Conquer” and “Naive” algorithm in a loop for 15000 trials separately.
When input size is 400, the average time for “Divide-and-Conquer” is equal to “Naive” is
When input size is 600, the average time for “Divide-and-Conquer” is less 1 milliseconds than “Naive”.
When input size is 1000, the total time for “Divide-and-Conquer” is 4867 milliseconds; the total time for “Naive” is 76323 milliseconds
When input size is 1100, the total time for “Divide-and-Conquer” is 5585 milliseconds; the total time for “Naive” is 93290
When input size is 1200, the total time for “Divide-and-Conquer” is 5876 milliseconds; the total time for “Naive” is 106848 milliseconds

So, in my opinion, the range of crossover points could be (500,600).