BLG 336E
Homework 2 Report
150130032 – Baran Kaya

1)Master Theorem

- Master theorem is a method for solving complexity of algorithms. It has 3 cases and all of them requires different conditions.
- a -> number of divides (log(n)), b -> number of parts for divide

2)Problem

- Divide full vector into 2 parts and calculate these parts distances. However for calculating these parts distance's divide these parts into 2 until there is less then 3 element vectors. Then calculate these small vector's point distance with brute force and return these result to the upper levels.

3)Algorithm

Pseudo-code

```
Read file and put values into the pointVector
Call closestPair(pointVector) function

ClosestPair(pointVector)
   Px = sorted pointVector by x-coor.                                  O(nlogn)
   Py = sorted pointVector by y-coor.                                  O(nlogn)
   Return closestPairRec(Px, Py)

ClosestPairRec(Px, Py)
   If(size of Px <= 3)
      Solve with brute force and return min dist. Point pair.
      (Size 2 and 3 have different solition in the code)               O(2)|O(3)
   Divide Px from middle, left part is Qx right part is Rx

   Call closestPairRec(Qx, Py)                                         T(n/2)
   Call closestPairRec(Rx, Py)                                         T(n/2)

   Find min distance of Qx and Rx results and assign it to the delta

   Construct S vector which (Point's x coor – mid point's x coor < delta)   O(15logn)
   Sort S vector by y-coor.                                            O(logn)
   Check every point in S and compare it with next 15 points in S

   If there is smaller distance Point pair in S
      Return this pair in S
   Else if Qx result < Rx result
      Return Qx result's pair
   Else
      Return Rx result's pair                                          Total: O(nlogn)
```

Complexity

$T(n) = T(n/2) + T(n/2) + f(n)$
$F(n) = n \rightarrow$ Merging

4)Results

-Number of calculations and run times does not change with respect to N. For example 1000 to 5000 it should be 5N but time and calculations increase with ~7N.

Data 1000:
- Distance: 16.9115
- Calculations: 37953
- Run times: 44017600 ns, 44030400 ns, 44016000 ns, 43530700 ns

Data 5000:
- Distance: 37.3631
- Calculations: 274457
- Run times: 547966700 ns, 550512100 ns, 550950100 ns, 550626600 ns

Data 10000:
- Distance: 30.2655
- Calculations: 613972
- Run times: 1804497800 ns, 1805189700 ns, 1789779300 ns, 1780081000 ns

Data 25000:
- Distance: 0
- Calculations: 434996
- Run times: 7275643400 ns, 7272807300 ns, 7244920900 ns, 7268265400 ns

Compiling on SSH: g++ -std=c++11 150130032.cpp -o B --> It needs C++11.

150130032
Baran KAYA