1. My algorithm uses a divide and conquer approach to find the shortest distance between two points in a two dimensional plane. If there are less than three or equal to three points, it will not use the divide and conquer algorithm but uses the brute force approach.
2. Major codes:

Input coords:

1.0 2.0 3.0

4.5 6.7 8.9

-1.2 0.5

7.8 9.1 2.3

ClosestPairResult closestPair(vector<Coords> &points, int left, int right){

if(right - left <=3) {

//use brute force if there are less than or equal to 3 points

ClosestPairResult result;

result.distance = DBL\_MAX;

for (int i = left; i < right; i++) {

for (int j = i + 1; j < right; j++) {

double dist = distance(points[i], points[j]);

if (dist < result.distance) {

result.i = i;

result.j = j;

result.distance = dist;

}

}

}

return result;

}

int mid = (left + right) / 2;

ClosestPairResult leftResult = closestPair(points, left, mid);

ClosestPairResult rightResult = closestPair(points, mid, right);

ClosestPairResult result = (leftResult.distance < rightResult.distance) ? leftResult : rightResult;

//store the points within the strip

vector<Coords> strip;

for(int i = left; i <right; i++){

double x\_distance = fabs(points[i].x - points[mid].x);

if (x\_distance < result.distance){

strip.push\_back(points[i]);

}

}

sort(strip.begin(), strip.end(), compareY);

//find minDistance in strip vector if found

int stripSize = strip.size();

for (int i = 0; i <stripSize; i++){

for(int j = i + 1; j < stripSize && (strip[j].y - strip[i].y ) < result.distance; j++){

double dist = distance(strip[i], strip[j]);

if (dist < result.distance) {

result.i = i;

result.j = j;

result.distance = dist;

}

}

}

return result;

}

Over here, I use the divide conquer approach given by a list of coords to find the closest two points and then return it as a result of this format: [i j D] (i is the index of point P[i]; j is the index of point P[j]; D is the minimum distance between P[i] and P[j] in a structure of ClosestPairResult which I will be writing to the output file.

double closestPairBruteForce(vector<Coords>& point){

int n = point.size();

double minDistance = DBL\_MAX;

for (int i = 0; i < n; i++){

for (int j = 0; j < n; j++){

minDistance = min(minDistance, distance (point[i], point[j]));

}

}

return minDistance;

}

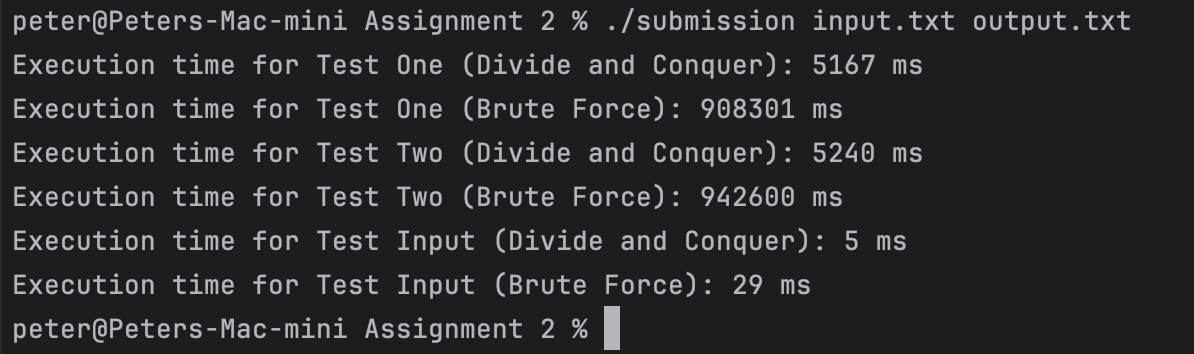
Here is the brute force approach which outputs the same result.

Output:

0 1 2.66271

The minimum distance is 2.6666.

1. Running results of three test cases:



1. There aren't any known bugs.