Problem : Given n points in a plane, find the pair that is closest.

- The problem falls under the discipline of Computational geometry.

- Brute force method will take $O(n^2)$ time.

Say, $P = (x_1, y_1)$ $\quad Q = (x_2, y_2)$

$$PQ = \sqrt{(x_2 - x_1)^2 + (y_2 - y_1)^2}$$

- Shamos & Hoey Solved the Same problem more efficiently using Divide & Conquer

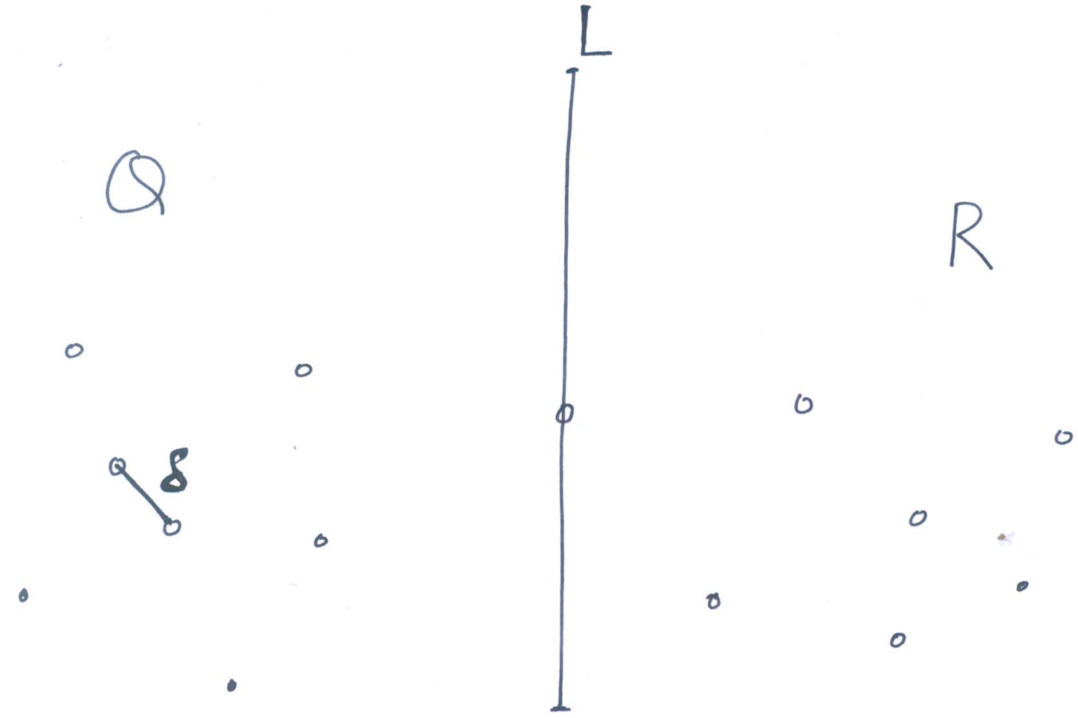
Notation

$$P = \{p_1, p_2, p_3, \ldots, p_n\}$$

is the set of points.

- $d(p_i, p_j)$ denotes the Euclidian distance between p_i and p_j
- (x_i, y_i) denotes the coordinates of the point p_i
- We assume that no two points have the same x coordinates or the same y coordinate.
- If not, we can enforce this by slight rotation of the plane.

- $P = \{p_1, p_2, p_3, \ldots, p_n\}$ is the set of points.
- P_x denotes the list wherein all the points have been sorted by x coordinate.
- P_y denotes the list wherein all the points have been sorted by y coordinate.
- Q denotes the set of points in the Ist $\lceil \frac{n}{2} \rceil$ positions of P_x.
- R denotes the set of points in the 2nd $\lfloor \frac{n}{2} \rfloor$ positions of P_x



- Recursively determine the closest pair of points in Q and R.
- Let (q_0^*, q_1^*) are the closest pts in Q.
- Let (r_0^*, r_1^*) are the closest pts in R.

Let $\delta = \text{Min}\left[d(q_0^*, q_1^*),\ d(r_0^*, r_1^*)\right]$

Determine whether $\exists q \in Q \wedge \exists r \in R$ s.t $d(q, r) < \delta$

If No, the closest pair is in Q or R

If Yes, the closest pair is across Q and R

Lemma : If $\exists q \in Q \wedge \exists r \in R$ s.t $d(q,r) < \delta$, then each of q and r lies within a distance of δ from L.

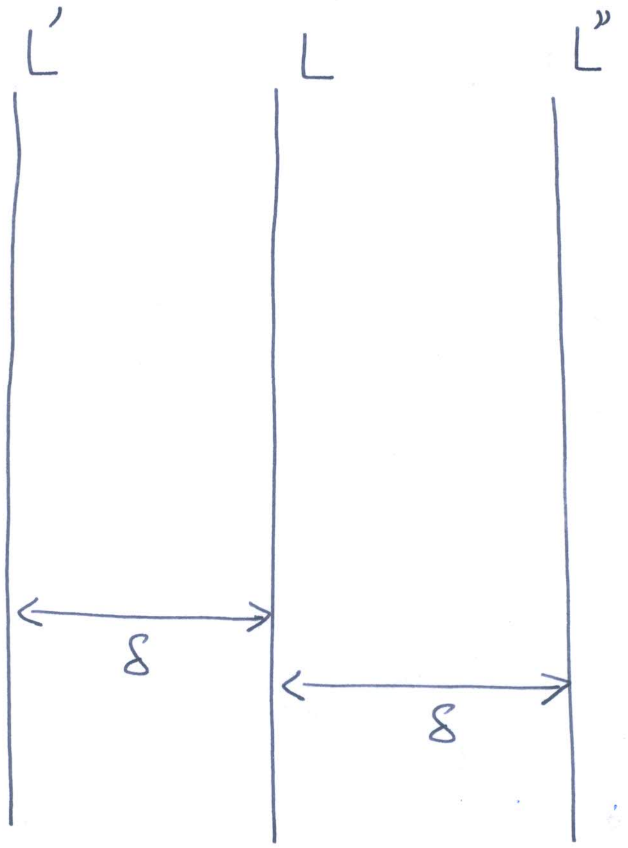
Proof : • Suppose q and r exist and $q = (q_x, q_y)$, $r = (r_x, r_y)$

• Suppose the equation of line L is $x = x^*$

Now $q_x \leq x^* \leq r_x$

• $x^* - q_x \leq r_x - q_x \leq d(q,r) < \delta$

• $r_x - x^* \leq r_x - q_x \leq d(q,r) < \delta$



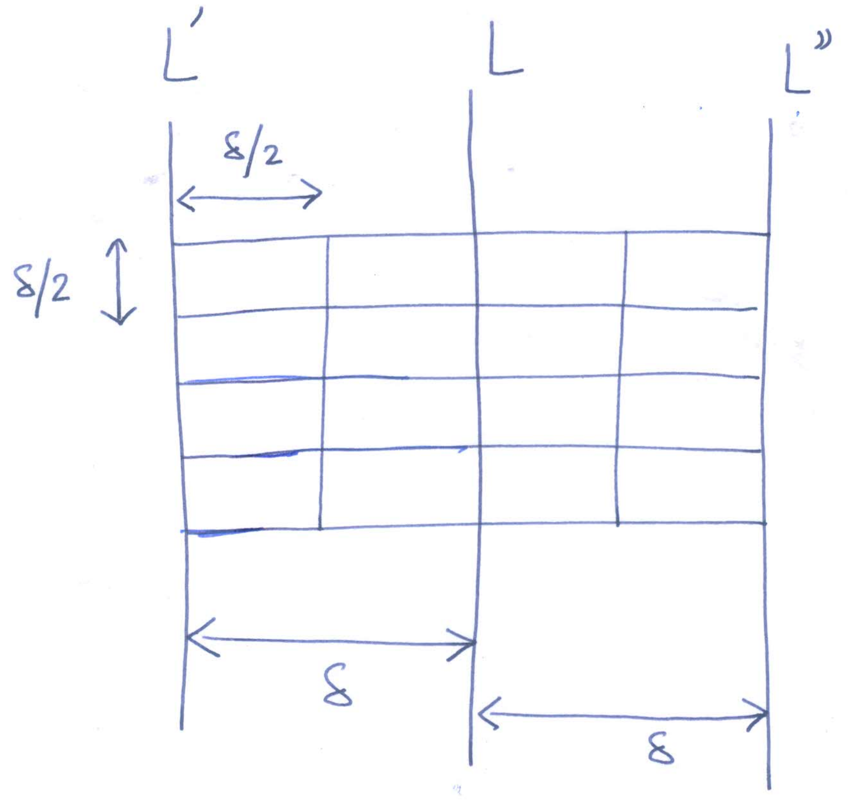
- Say $S \subseteq P$ is the area enclosed between L' and L''

- $\exists q \in Q \wedge r \in R$ s.t $d(q, r) < \delta$ iff $\exists s, s' \in S$ s.t $d(s, s') < \delta$

- Say S_y is the ~~set~~ list of points in S sorted by y coordinate.

Theorem: If $s, s' \in S$ have the property that $d(s, s') < \delta$, then s, s' are within 15 positions of each other in S_y.

Proof:



- Each $\delta/2$ by $\delta/2$ box has at most 1 point.
- If s and s' are more than 15 positions away, they must be separated by at least 3 rows in above grid.

- We want to check whether $\exists\, s, s' \in S$ s.t $d(s, s') < \delta$

- For each $s \in S_y$, we compute its distance to each of the next 15 pts in S_y. We consider the points with the shortest distance. [$O(n)$ time job]

- <u>Complexity</u>:
  - Say $T(n)$ is the total time.
  - Given $P = \{p_1, p_2, \ldots, p_n\}$, P_x and P_y can be built in $O(n \cdot \log n)$ time
  - Hence, Q and R can be defined in one pass over P_x: $O(n)$ time.
  - Recursion on each of Q & R costs $T\left(\frac{n}{2}\right)$
  - S_y can be obtained from P_y in $O(n)$ time.
  - $T(n) = 2 \cdot T\left(\frac{n}{2}\right) + O(n)$
  - $T(n) = O(n \cdot \log n)$.