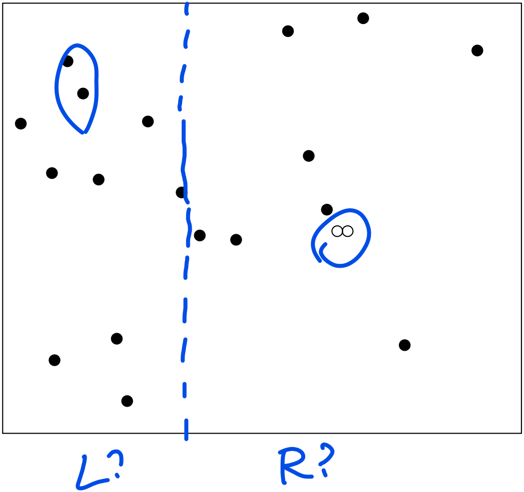
# CompSci 161
# Winter 2023 Lecture 23:
# Divide and Conquer V:
# Closest Pair of Points

# Closest Pair of Points

Given $n$ points, find the closest together.



# Closest Pair of Points: Brute Force

`Closest-Pair`
**Input**: $n$ points in $2D$-space
**Output**: The closest pair of points.

min $= \infty$
**for** $i = 2 \to n$ **do**
  **for** $j = 1 \to i - 1$ **do**
    **if** $(x_j - x_i)^2 + (y_j - y_i)^2 <$ min **then**
      min $= (x_j - x_i)^2 + (y_j - y_i)^2$
      closestPair $= ((x_i, y_i), (x_j, y_j))$
**return** closestPair

# Closest Pair of Points: Starting D&C

Incorrect but reasonable starting point:

Sort at beginning before call??

`Closest-Pair`

Sort by y initially

**Input**: $n$ points in $2D$-space

**Output**: The closest pair of points.

If $P$ is sufficiently small, use brute force. // $\mathcal{O}(1)$

$x_m \leftarrow$ median $x$-value from $P$ // $\Theta(n)$ via Select

$L \leftarrow$ any points from $P$ with $x$-coordinate $\leq x_m$

$R \leftarrow$ any points from $P$ with $x$-coordinate $> x_m$

keep sorted

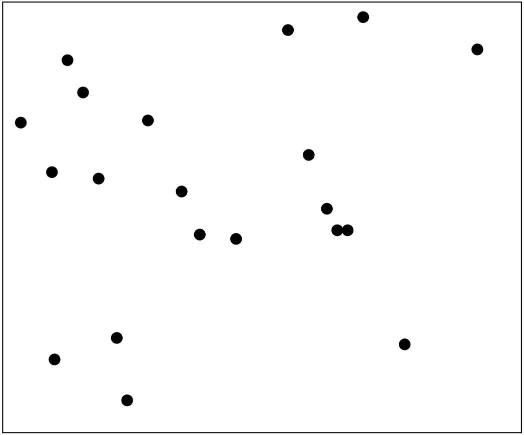
Let $l_1$ and $l_2$ be the closest pair of points in $L$

Let $r_1$ and $r_2$ be the closest pair of points in $R$

**return** whichever pair is closer together

What if middle "has" closest pair?

# Visualizing the D&C



# Closest Pair of Points: Idea One

`Closest-Pair`

**Input**: $n$ points in $2D$-space

**Output**: The closest pair of points.

If $P$ is sufficiently small, use brute force. // $\mathcal{O}(1)$

$x_m \leftarrow$ median $x$-value from $P$

$L \leftarrow$ any points from $P$ with $x$-coordinate $\leq x_m$

$R \leftarrow$ any points from $P$ with $x$-coordinate $> x_m$

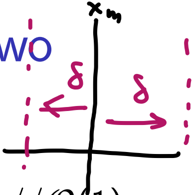
Let $l_1$ and $l_2$ be the closest pair of points in $L$

Let $r_1$ and $r_2$ be the closest pair of points in $R$

Check all pairs $(a, b)$ with $a \in L$ and $b \in R$

$T(n) = 2T\left(\frac{n}{2}\right) + n^2$ is $\Theta(n^2)$

# Closest Pair of Points: Idea Two




Idea two. Let $\delta =$ and partition the two...

If $P$ is sufficiently small, use brute force. $//\mathcal{O}(1)$

$x_m \leftarrow$ median $x$-value from $P$

$L \leftarrow$ any points from $P$ with $x$-coordinate $\leq x_m$

$R \leftarrow$ any points from $P$ with $x$-coordinate $> x_m$

Let $l_1$ and $l_2$ be the closest pair of points in $L$

Let $r_1$ and $r_2$ be the closest pair of points in $R$

Let $\delta = \min(d(l_1, l_2), d(r_1, r_2))$ $\leftarrow$

Let $M =$ set of points that could be closer than $\delta$

(points within $\delta$ of $x_m$)
along the $x$-axis.

# Finding closest pair in $M$

Now we have:

- $\delta =$ closest pair from $L$ or $R$
- a set $M$ of points, sorted by $y$-coordinate,

Determine one of two pieces of information:

- The closest pair of points in $M$
- Nothing, if closest pair in $M$ more than $\delta$ apart

# How close can points in $M$ be?

In order to finish the conquer step efficiently, I need to prove this claim:

**Claim:** if two points in $M$ are distance $< \delta$ apart, then their indices are within a constant $C \leq 11$

**Why would I care?** This is $O(n)$

$\min = \delta$

for $i = 1 \ldots |M|$

for $j = i+1 \ldots \cancel{|M|}$ $\min(i+C, |M|)$

if $d(M_i, M_j) < \min$

$\min = d(M_i, M_j)$

Time for alg:

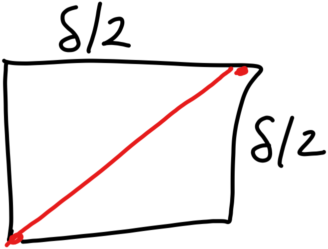
$T(n) = 2T\left(\frac{n}{2}\right) + n \longrightarrow \Theta(n \log n)$

# How close can points in $M$ be?

In order to finish the conquer step efficiently, I need to prove this claim:

**Claim:** if two points in $M$ are distance $< \delta$ apart, then their indices are within a constant

**Proof:** Look at points between $x_m + \delta$ and $x_m - \delta$




len diag:

$$\frac{\delta}{2}\sqrt{2} < \delta$$

# How close two points in $M$?