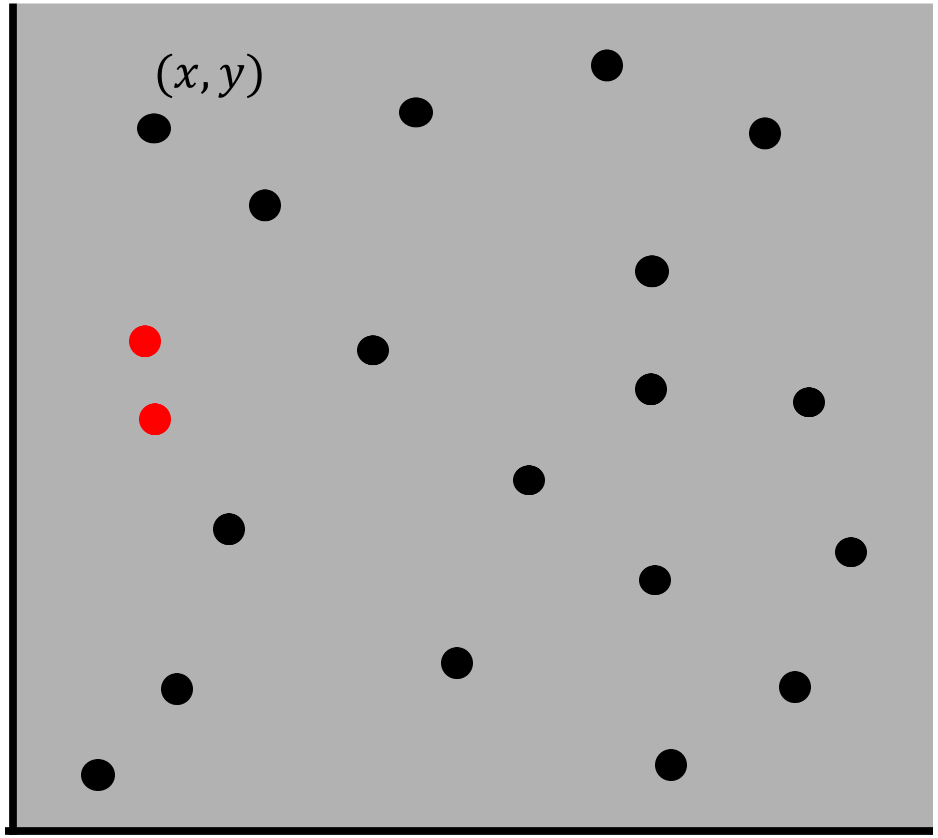
Randomized Algorithms

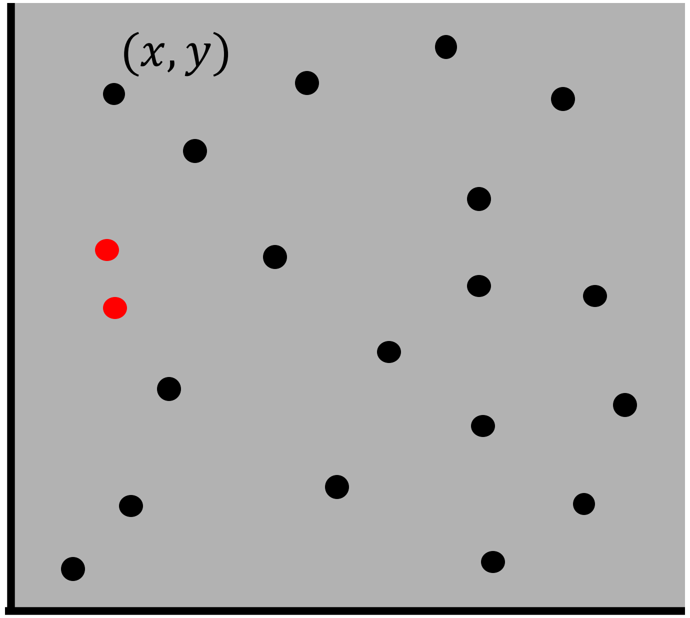
CSCI 432

Closest Pair Problem



Given n points, find a pair of points with the smallest distance between them.

Closest Pair Problem



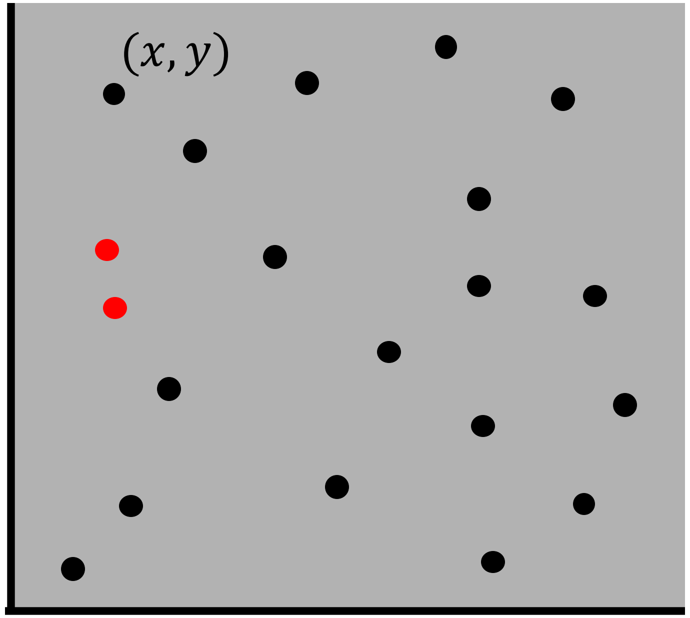
	P_1	P_2	…	P_n
P_1	/	$d_{1,2}$	…	$d_{1,n}$
P_2	$d_{2,1}$	/	…	$d_{2,n}$
…	…	…	…	…
P_n	$d_{n,1}$	$d_{n,2}$	…	/

Simple solution:

1. Compute distance for each pair.
2. Select smallest.

Running Time = ?

Closest Pair Problem



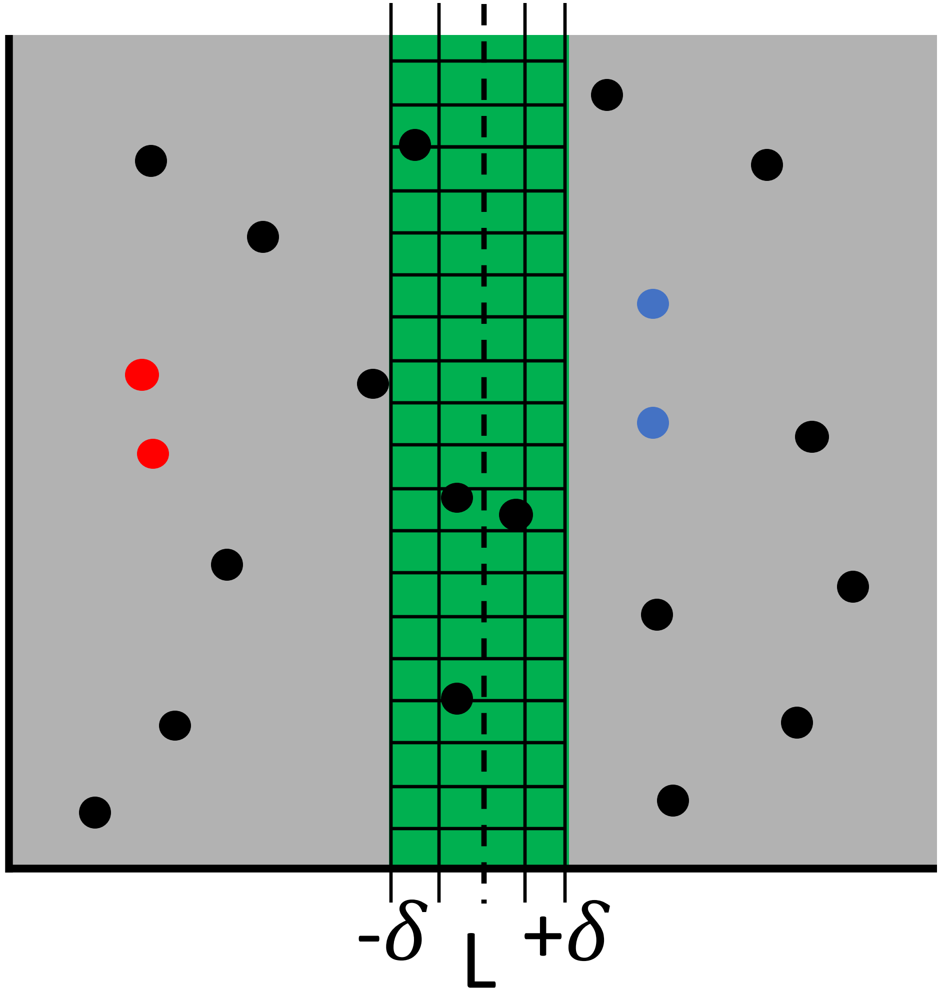
	P_1	P_2	...	P_n
P_1	/	$d_{1,2}$	...	$d_{1,n}$
P_2	$d_{2,1}$	/	...	$d_{2,n}$
...	...	...	...	...
P_n	$d_{n,1}$	$d_{n,2}$	...	/

Simple solution:

1. Compute distance for each pair.
2. Select smallest.

Running Time = $O(n^2)$

Closest Pair Problem



1. Sort points by x-coordinate and find median.
2. (Recursively) find closest left-hand and right-hand points.
3. Search for closer points that straddle median.

There is a constant limit on how many points we need to search through!

Running Time = $O(n \log n)$

Closest Pair Problem

Randomized Algorithm Basic Idea:

- Consider points in random order. $\{p_1, \ldots, p_n\}$

Closest Pair Problem

Randomized Algorithm Basic Idea:

- Consider points in random order. $\{p_1, \dots, p_n\}$
- Maintain value δ of closest pair encountered. $\delta = d(p_1, p_2)$ to start.

Closest Pair Problem

Randomized Algorithm Basic Idea:

- Consider points in random order. $\{p_1, \ldots, p_n\}$
- Maintain value δ of closest pair encountered. $\delta = d(p_1, p_2)$ to start.
- For new point p, check all "close" points for one $< \delta$.

Use hashing to keep track of "close" points.

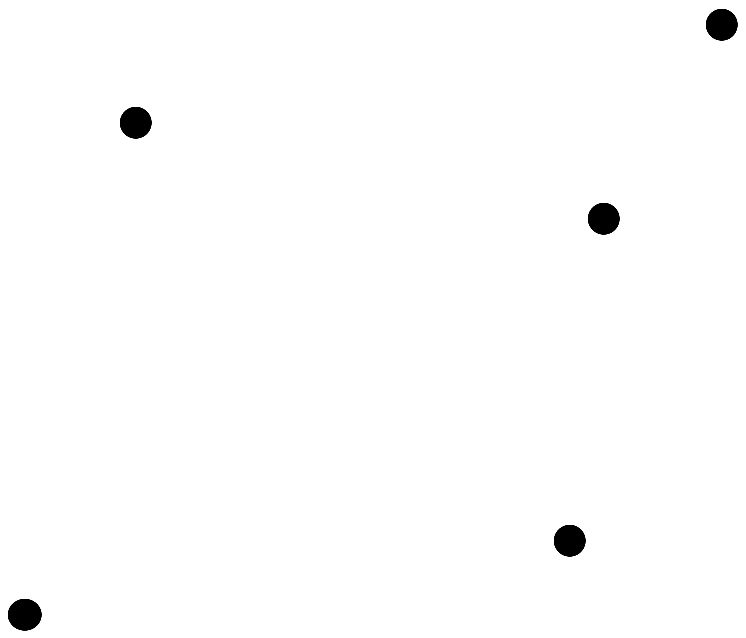
Closest Pair Problem

Randomized Algorithm Basic Idea:

- Consider points in random order. $\{p_1, \ldots, p_n\}$
- Maintain value δ of closest pair encountered. $\delta = d(p_1, p_2)$ to start.
- For new point p, check all "close" points for one $< \delta$.
- If found, stop, update δ and continue.

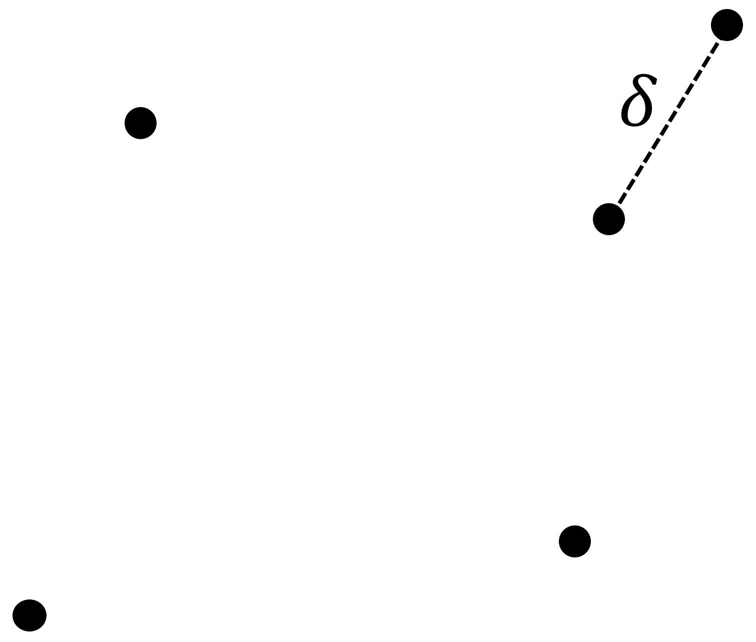
Closest Pair Problem

- Suppose we have already processed the first k points. (i.e. $\{p_1, \dots, p_k, \dots, p_n\}$).



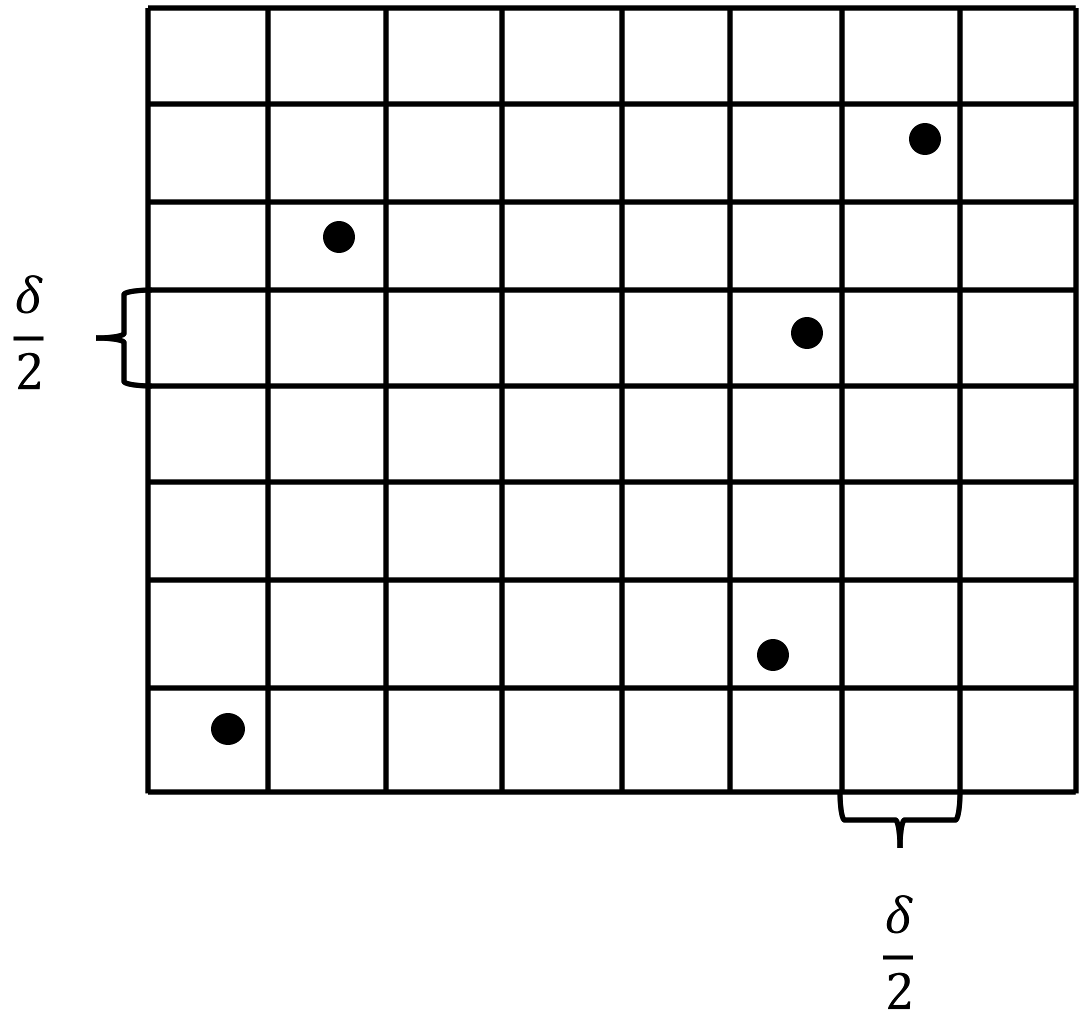
Closest Pair Problem

- Suppose we have already processed the first k points. (i.e. $\{p_1, \ldots, p_k, \ldots, p_n\}$).
- Let δ be the smallest distance in the first k points.



Closest Pair Problem

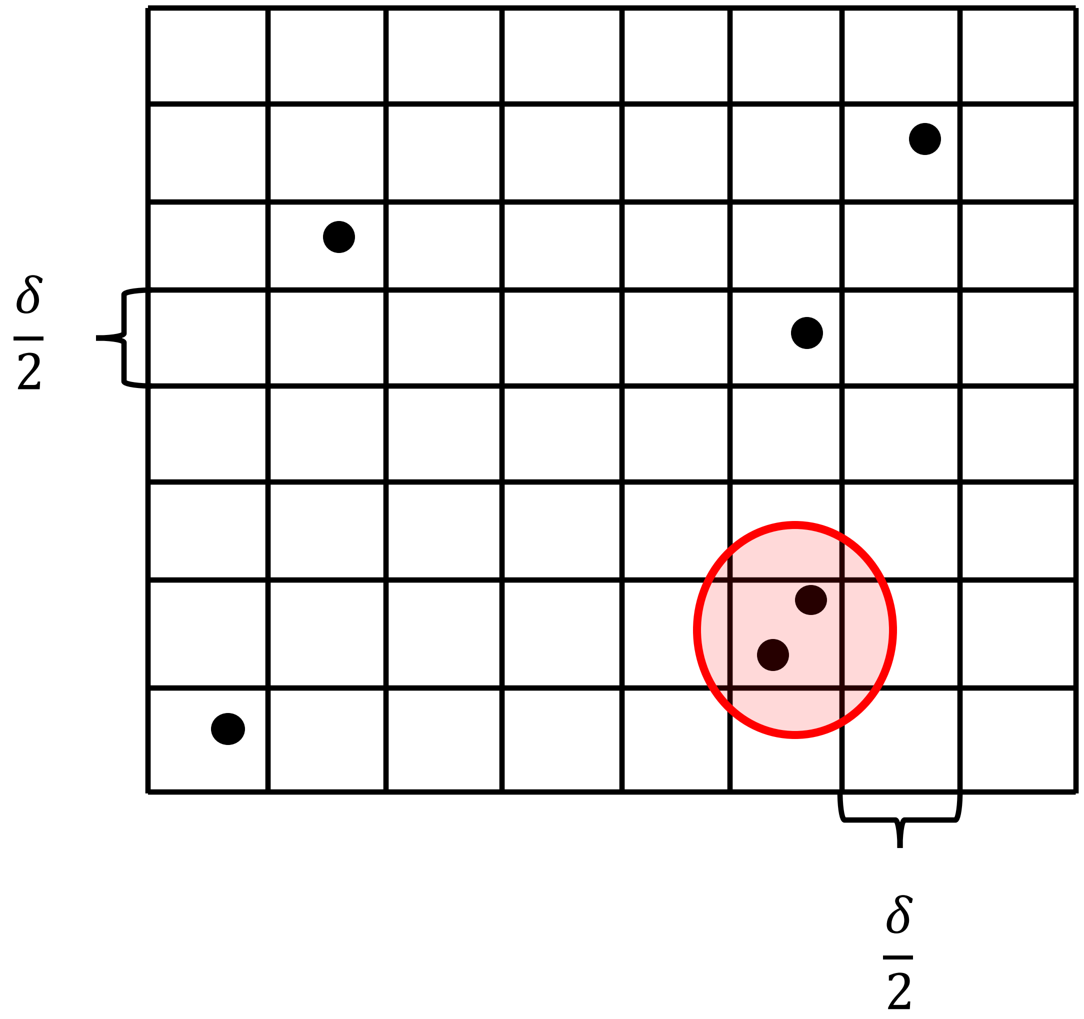
- Suppose we have already processed the first k points. (i.e. $\{p_1, \dots, p_k, \dots, p_n\}$).
- Let δ be the smallest distance in the first k points.
- Consider this grid overlaid on the first k points.



Closest Pair Problem

Claim: Each cell has at most 1 point in it.

Proof: ?



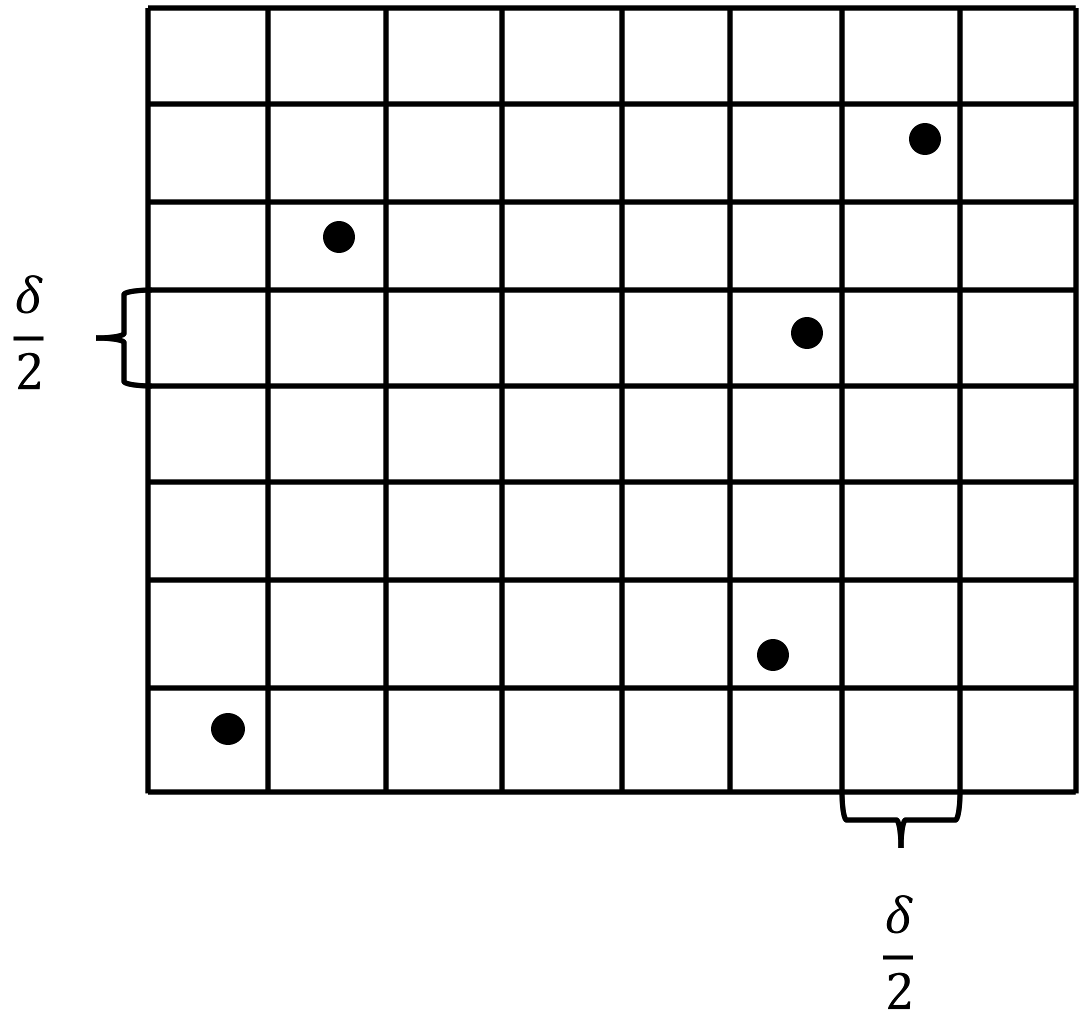
Closest Pair Problem

Claim: Each cell has at most 1 point in it.

Proof: The furthest points in a cell can be are in the corners at:

$$\sqrt{\left(\frac{\delta}{2}\right)^2 + \left(\frac{\delta}{2}\right)^2} = \frac{\delta}{\sqrt{2}} < \delta.$$

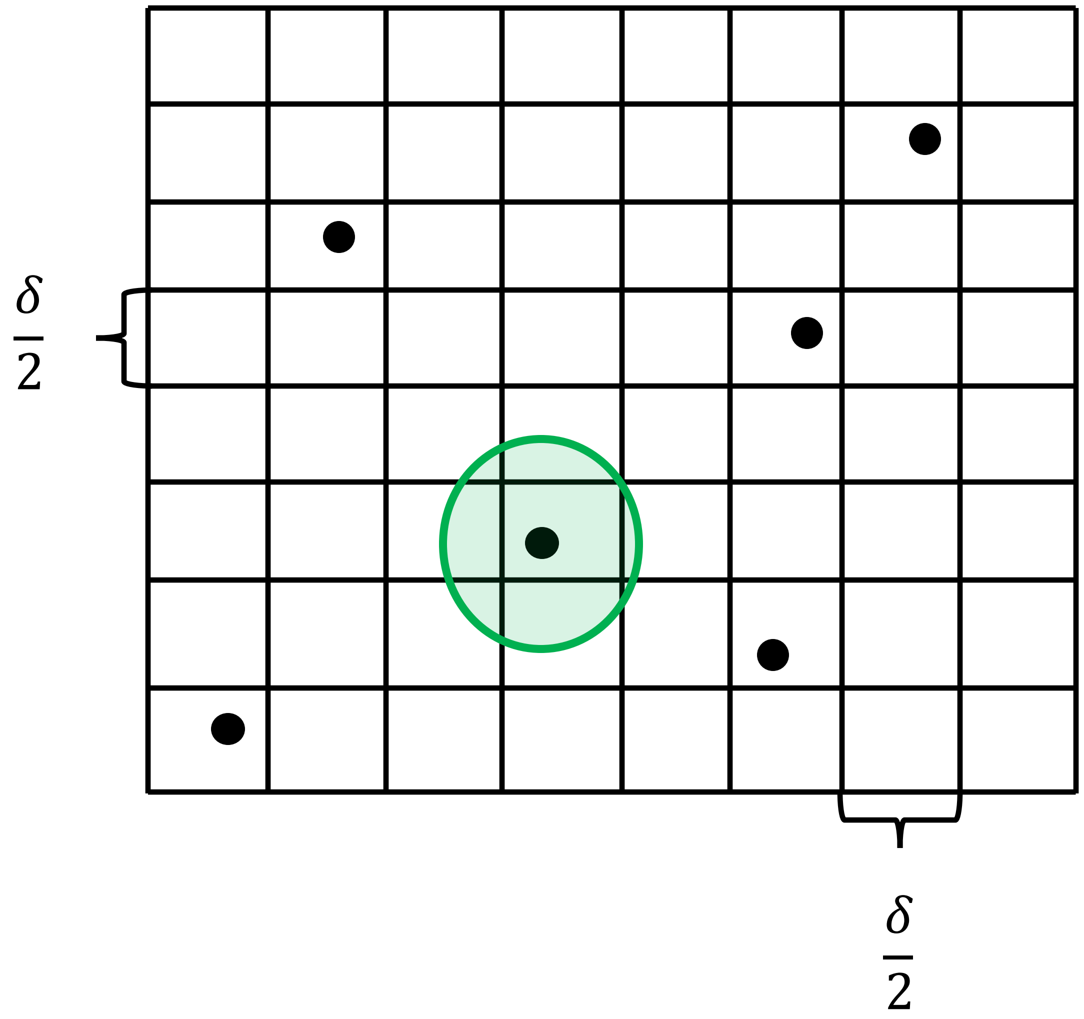
Since we stop as soon as we find some pair $< \delta$, we will never put two points in the same cell.



Closest Pair Problem

Claim: Each cell has at most 1 point in it.

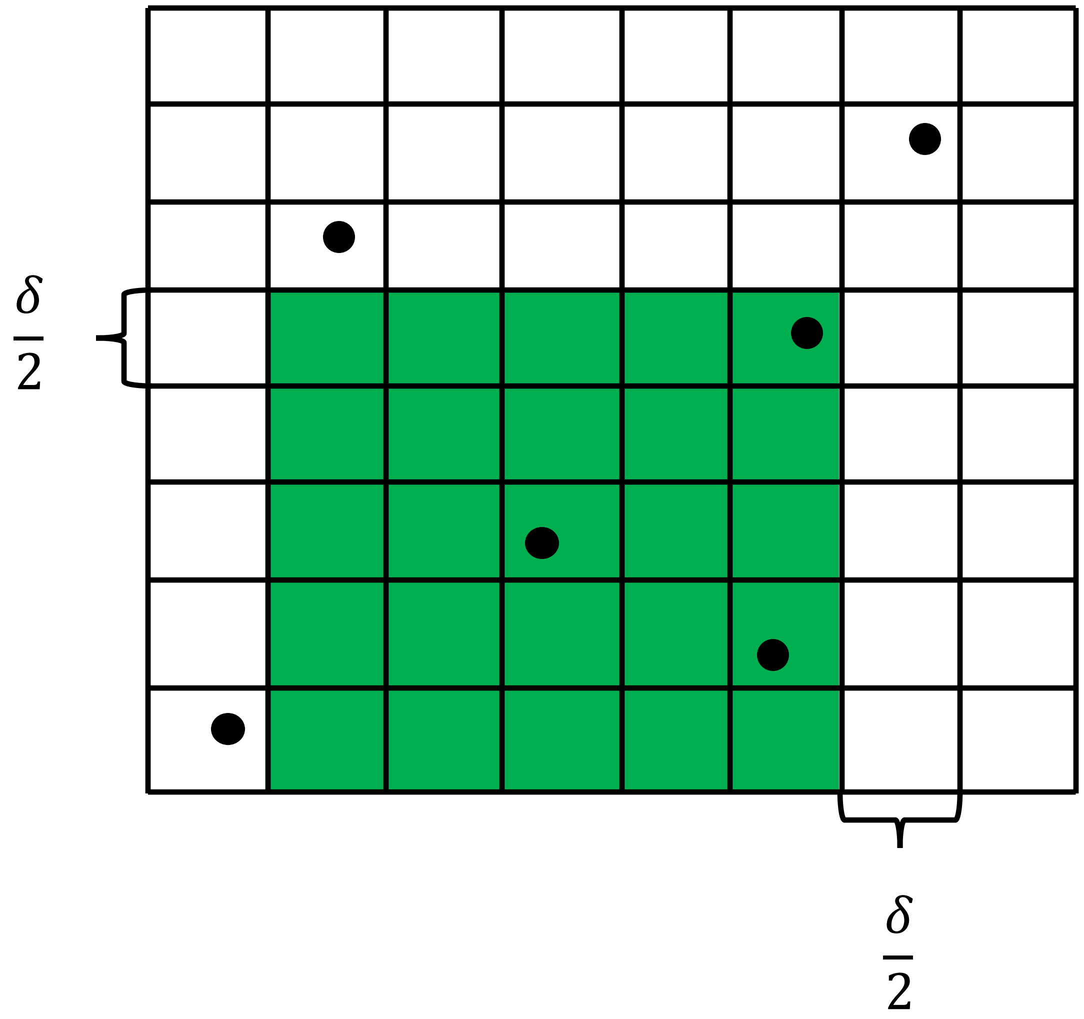
Given the location of a new point, which cells do we need to check for a point $< \delta$?



Closest Pair Problem

Claim: Each cell has at most 1 point in it.

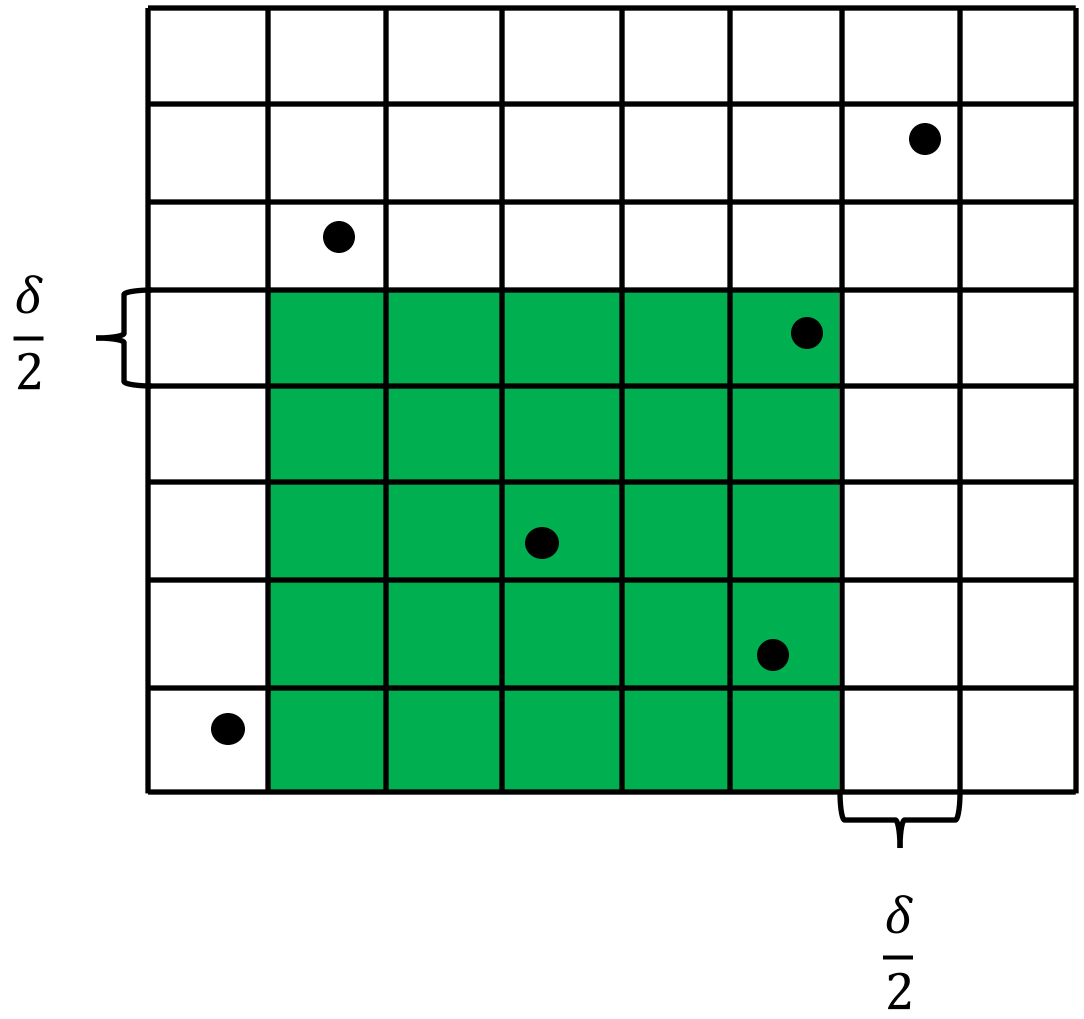
Given the location of a new point, which cells do we need to check for a point $< \delta$?



Closest Pair Problem

Claim: Each cell has at most 1 point in it.

Claim: We only need to check closest 25 cells to find point $< \delta$.

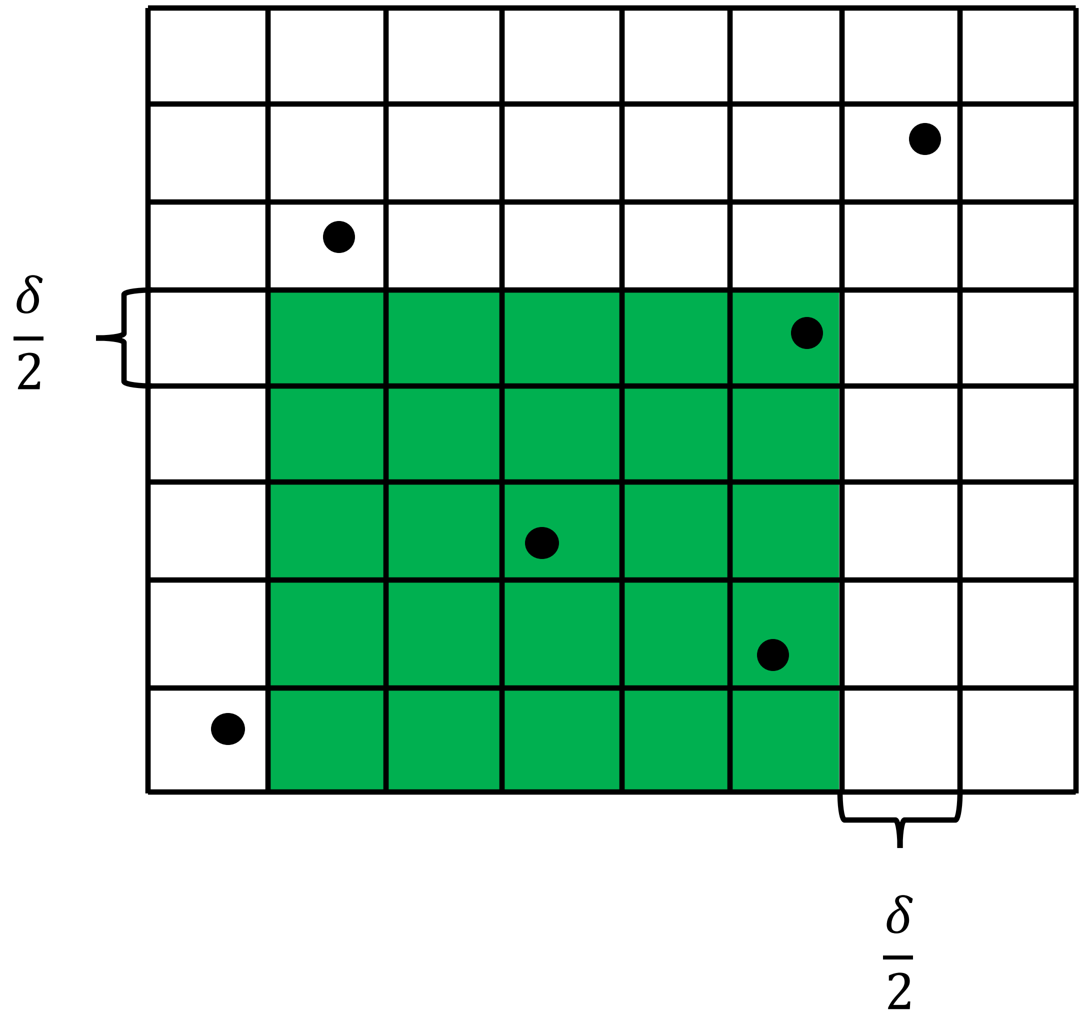


Closest Pair Problem

Claim: Each cell has at most 1 point in it.

Claim: We only need to check closest 25 cells to find point $< \delta$.

Conclusion: When we add a new point, we need to check its distance to $O(1)$ number of other points.



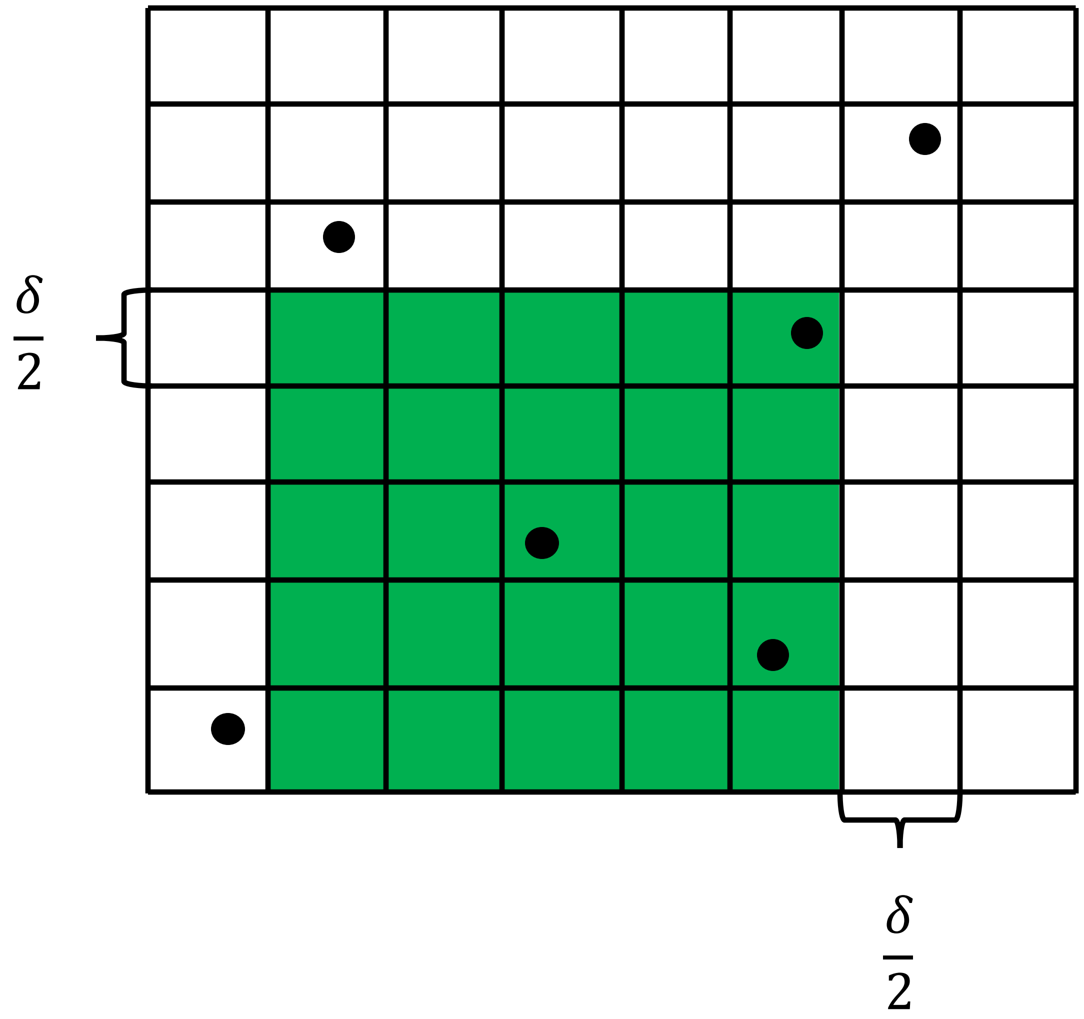
Closest Pair Problem

Claim: Each cell has at most 1 point in it.

Claim: We only need to check closest 25 cells to find point $< \delta$.

Conclusion: When we add a new point, we need to check its distance to $O(1)$ number of other points.

Can we find these points efficiently?



Closest Pair Problem

Claim: Each cell has at most 1 point in it.

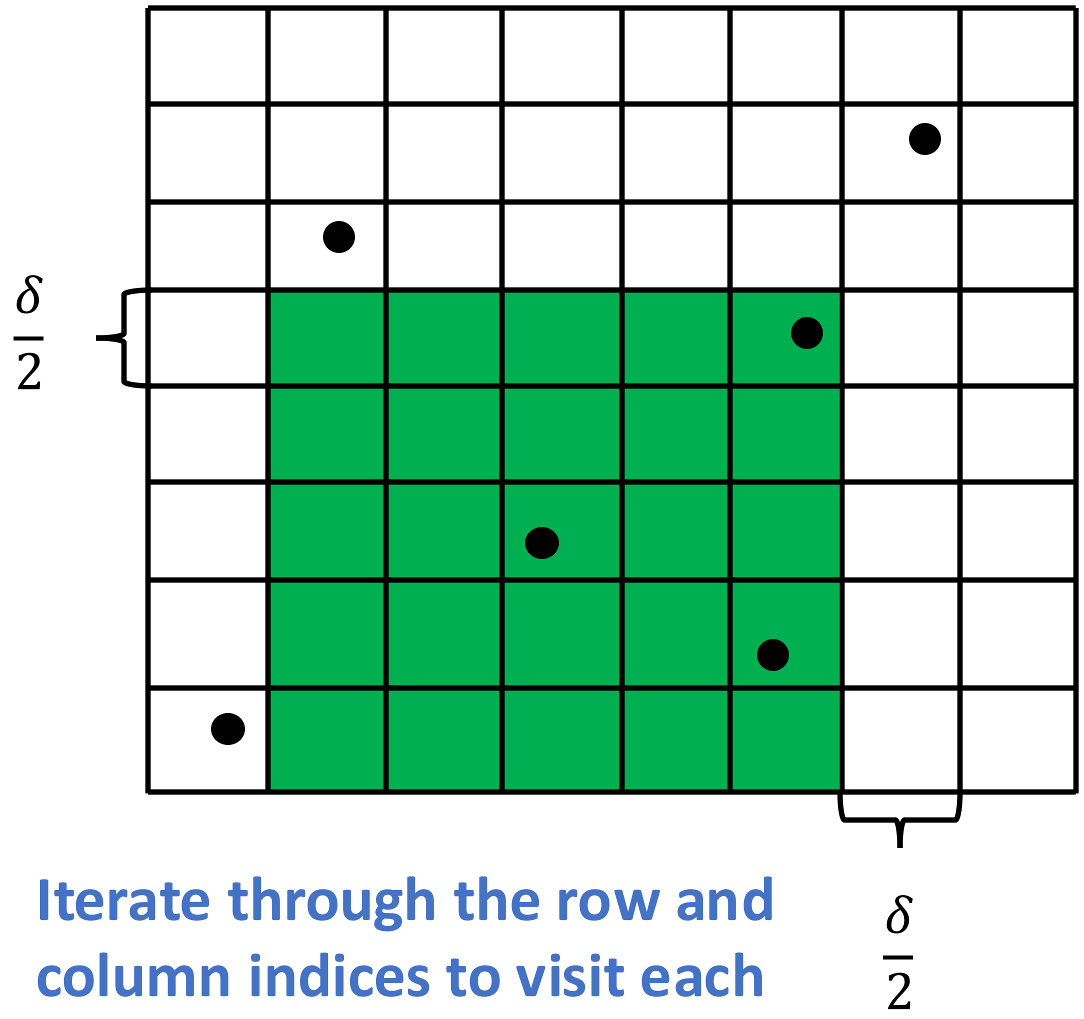
Claim: We only need to check closest 25 cells to find point $< \delta$.

Conclusion: When we add a new point, we need to check its distance to $O(1)$ number of other points.

Point (x, y) goes in cell $\left(\left\lfloor \frac{y}{\delta/2} \right\rfloor, \left\lfloor \frac{x}{\delta/2} \right\rfloor\right)$.

Row Index

Column Index



Iterate through the row and column indices to visit each of the 25 cells.

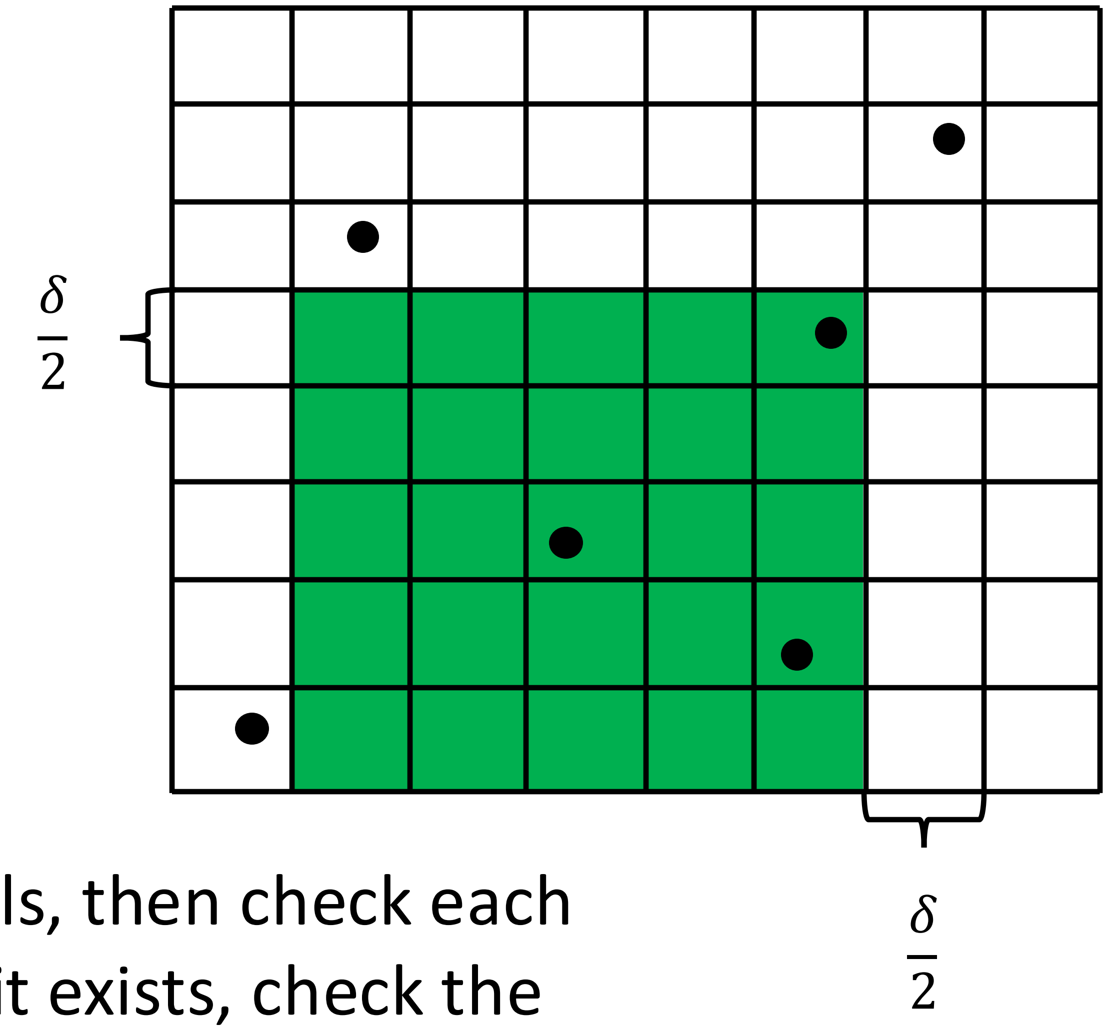
Closest Pair Problem

Claim: Each cell has at most 1 point in it.

Claim: We only need to check closest 25 cells to find point $< \delta$.

Conclusion: When we add a new point, we need to check its distance to $O(1)$ number of other points.

Point (x, y) goes in cell $\left(\left\lfloor\frac{y}{\delta/_2}\right\rfloor, \left\lfloor\frac{x}{\delta/_2}\right\rfloor\right)$.



Make a hash table that maps points to cells, then check each of the 25 cells for an entry in the table. If it exists, check the distance between the existing and new points.

Closest Pair Problem

```
findClosestPair(on sequence P)
```

Closest Pair Problem

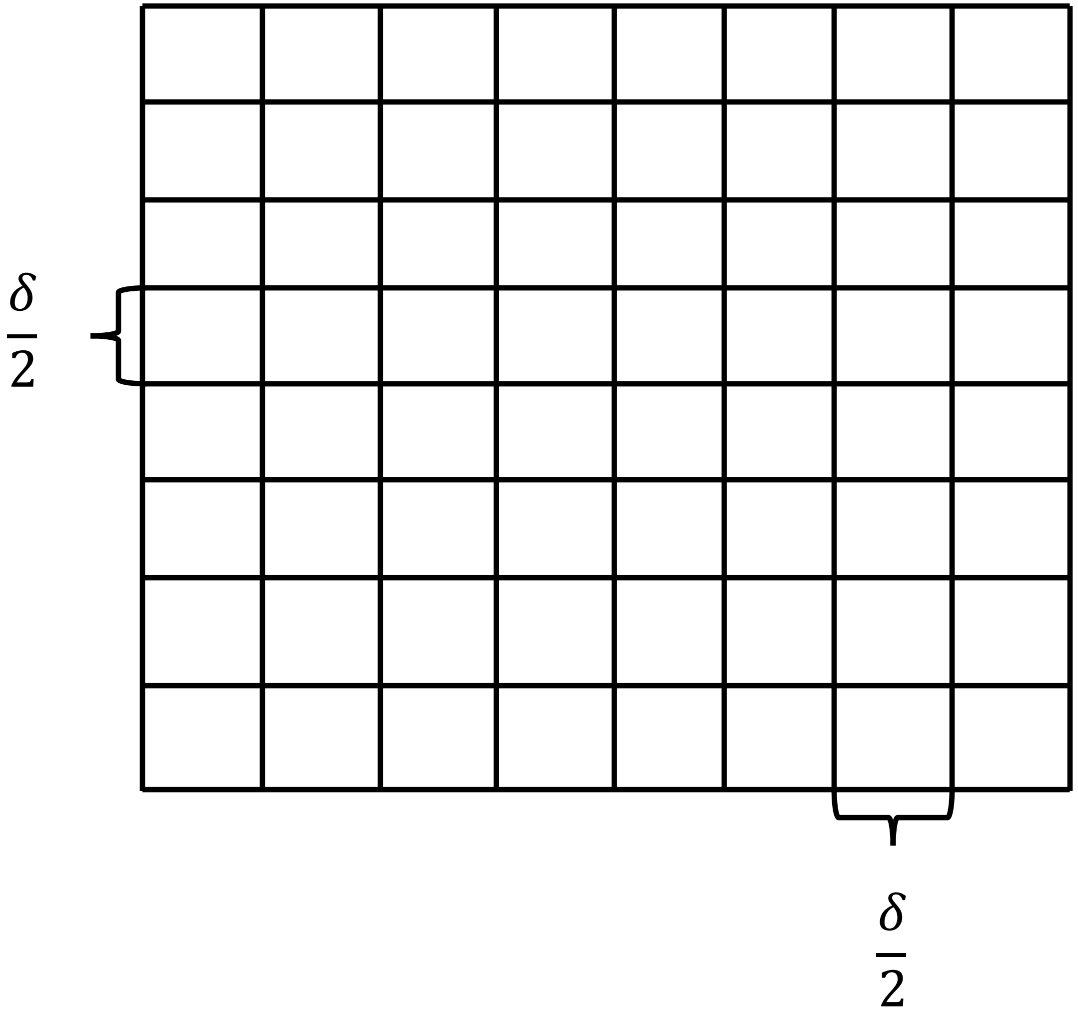
```
findClosestPair(on sequence P)
   Order points in random sequence
```

Closest Pair Problem

```
findClosestPair(on sequence P)
   Order points in random sequence
   δ = d(p₁, p₂)
```

Closest Pair Problem

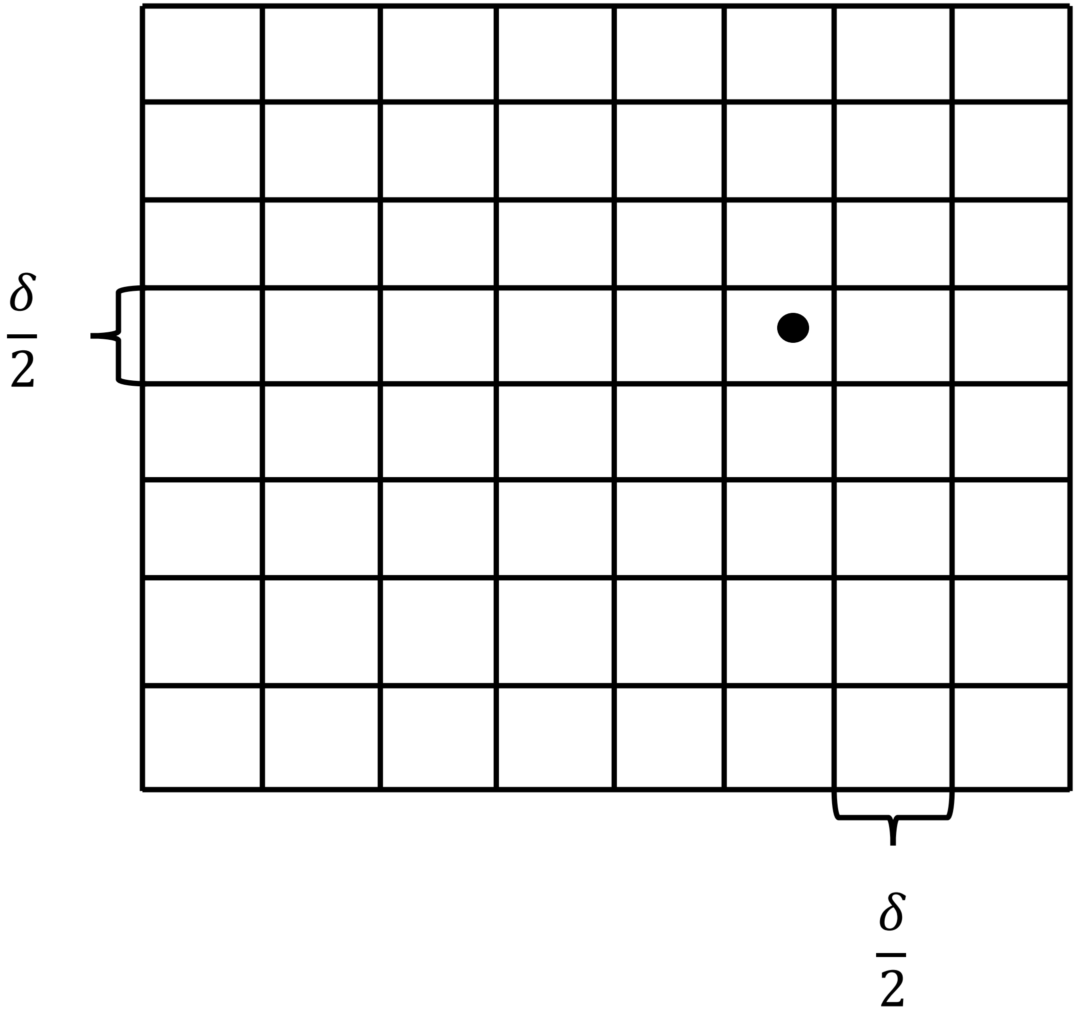
```
findClosestPair(on sequence P)
   Order points in random sequence
   δ = d(p₁, p₂)
   Make map of cell size δ/2
```



$$(x, y) \rightarrow \left(\left\lfloor \frac{y}{\delta/2} \right\rfloor, \left\lfloor \frac{x}{\delta/2} \right\rfloor \right)$$

Closest Pair Problem

```
findClosestPair(on sequence P)
   Order points in random sequence
   δ = d(p₁, p₂)
   Make map of cell size δ/2
   for i = 1, 2,..., n
      Determine cell containing pᵢ
```

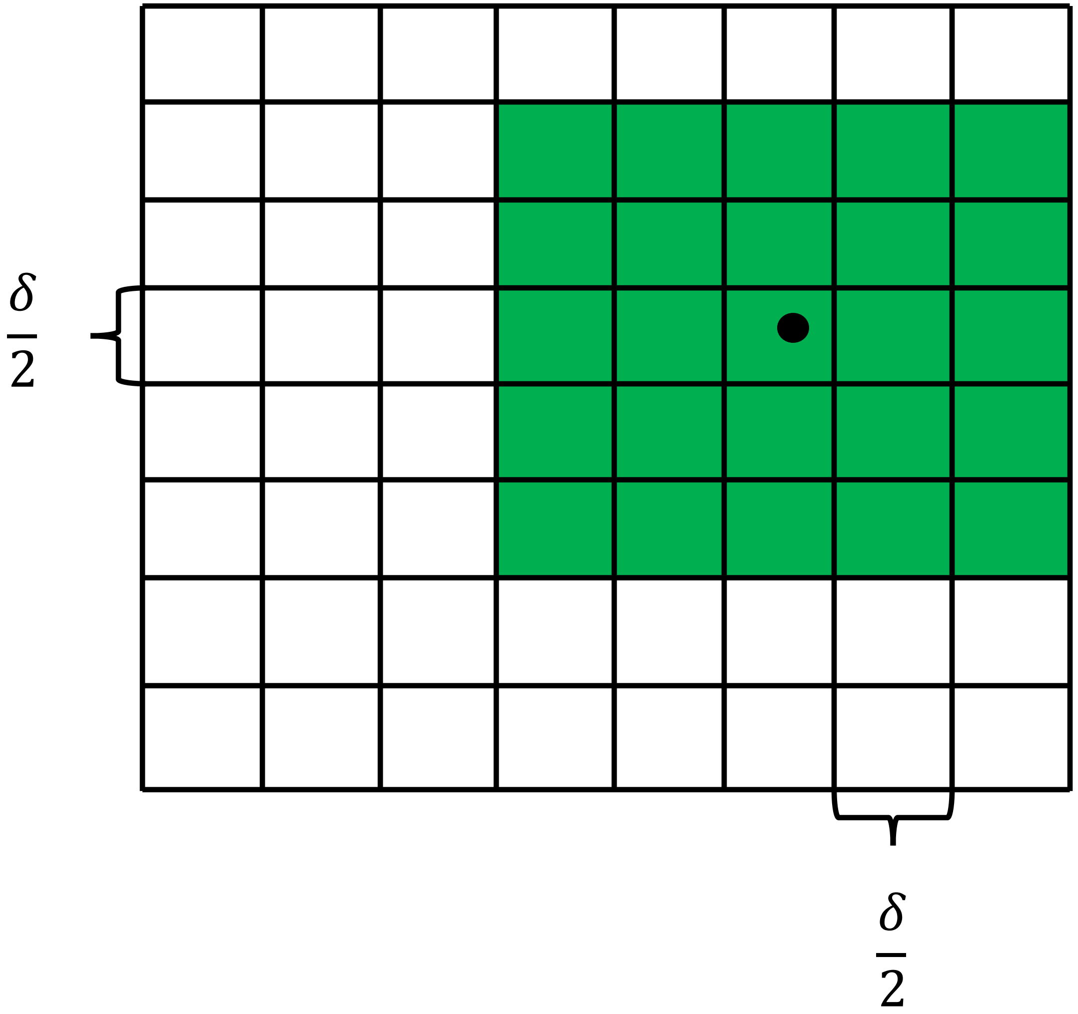


$$(x, y) \rightarrow \left(\left\lfloor \frac{y}{\delta/2} \right\rfloor, \left\lfloor \frac{x}{\delta/2} \right\rfloor \right)$$

row **col**

Closest Pair Problem

```
findClosestPair(on sequence P)
   Order points in random sequence
   δ = d(p₁, p₂)
   Make map of cell size δ/2
   for i = 1, 2,..., n
      Determine cell containing pᵢ
      Lookup 25 cells surrounding pᵢ
```



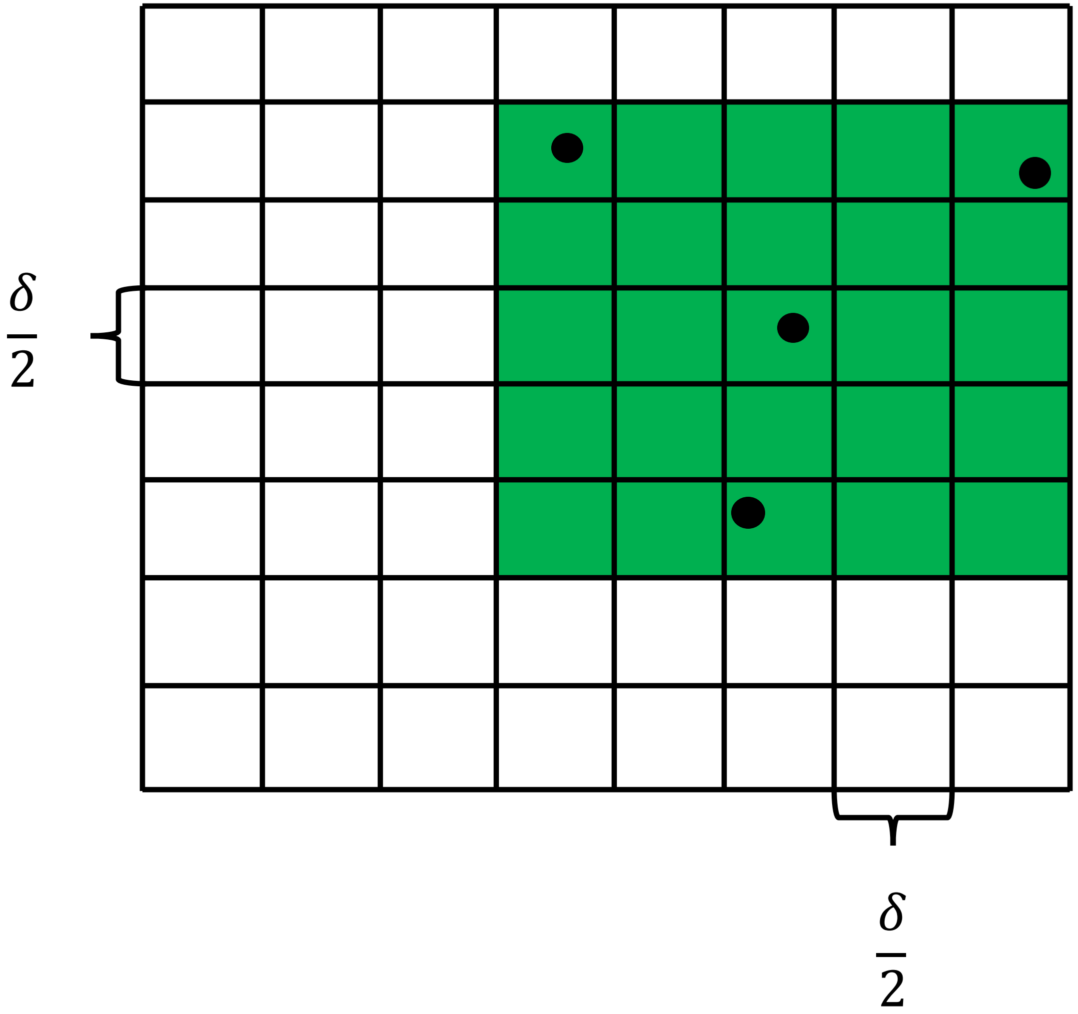
$$(x, y) \rightarrow \left(\left\lfloor \frac{y}{\delta/2} \right\rfloor, \left\lfloor \frac{x}{\delta/2} \right\rfloor \right)$$

row **col**

```
for r = row - 2 to row + 2
  for c = col - 2 to col + 2
    //check for entry (r,c)
```

Closest Pair Problem

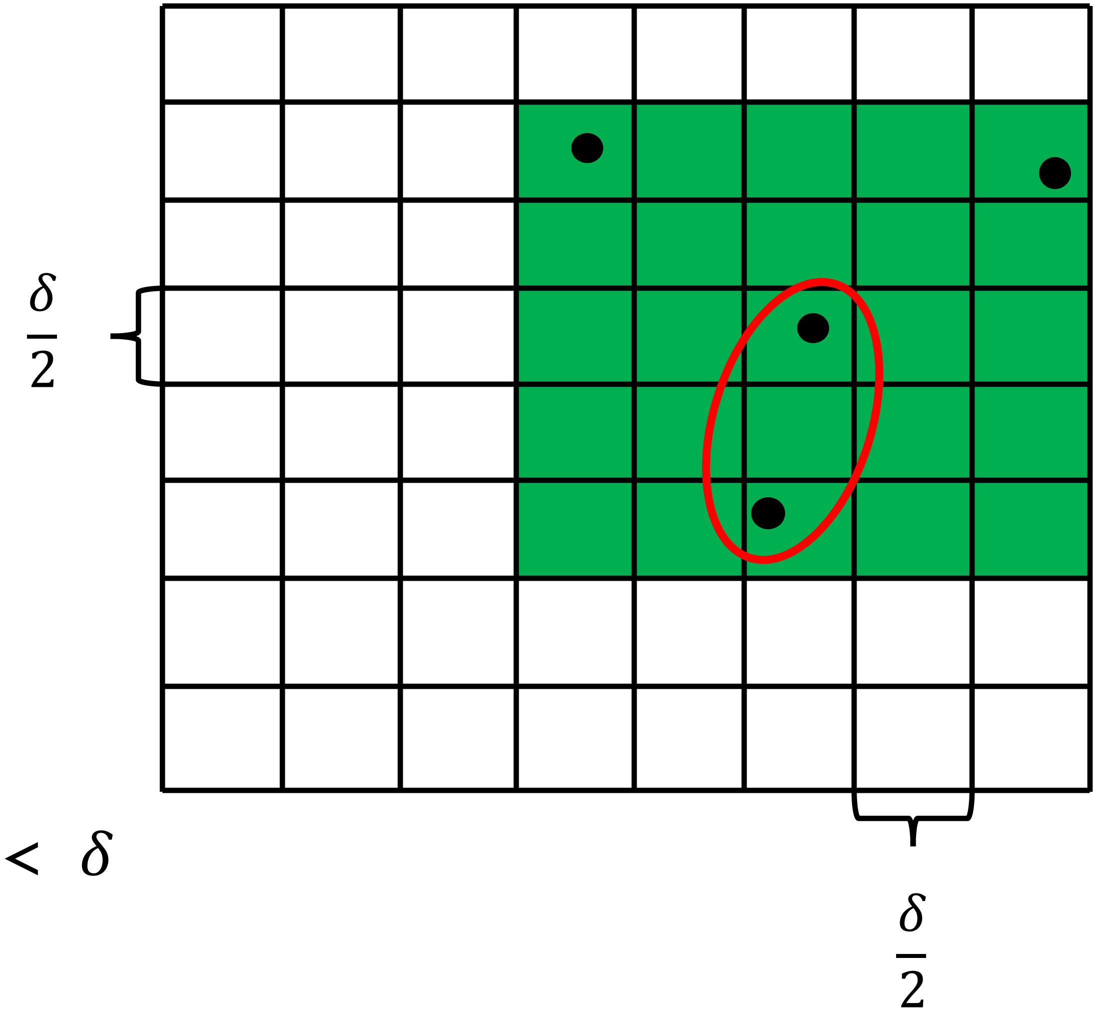
```
findClosestPair(on sequence P)
   Order points in random sequence
   δ = d(p₁, p₂)
   Make map of cell size δ/2
   for i = 1, 2,..., n
      Determine cell containing pᵢ
      Lookup 25 cells surrounding pᵢ
      Find dist to surrounding points
```



Closest Pair Problem

```
findClosestPair(on sequence P)
   Order points in random sequence
   δ = d(p₁, p₂)
   Make map of cell size δ/2
   for i = 1, 2,..., n
      Determine cell containing pᵢ
      Lookup 25 cells surrounding pᵢ
      Find dist to surrounding points
      if ∃ pⱼ such that δ′ = d(pᵢ, pⱼ) < δ
```

Use p_j that make δ' smallest

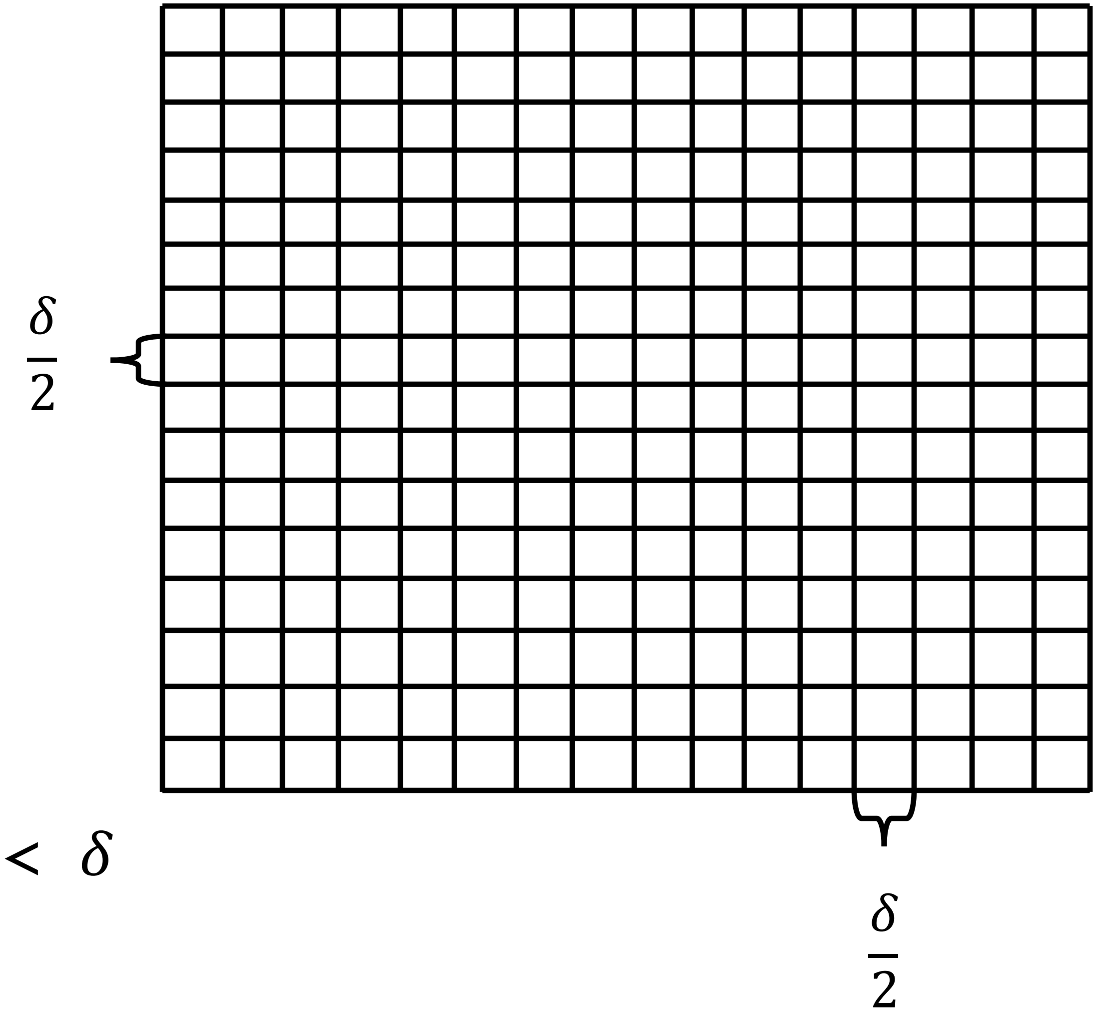


Closest Pair Problem

findClosestPair(on sequence P)
   Order points in random sequence
   δ = d(p_1, p_2)
   Make map of cell size $\delta/2$
   for i = 1, 2,..., n
      Determine cell containing p_i
      Lookup 25 cells surrounding p_i
      Find dist to surrounding points
      if $\exists$ p_j such that $\delta' = d(p_i, p_j) < \delta$
         Delete map

Closest Pair Problem

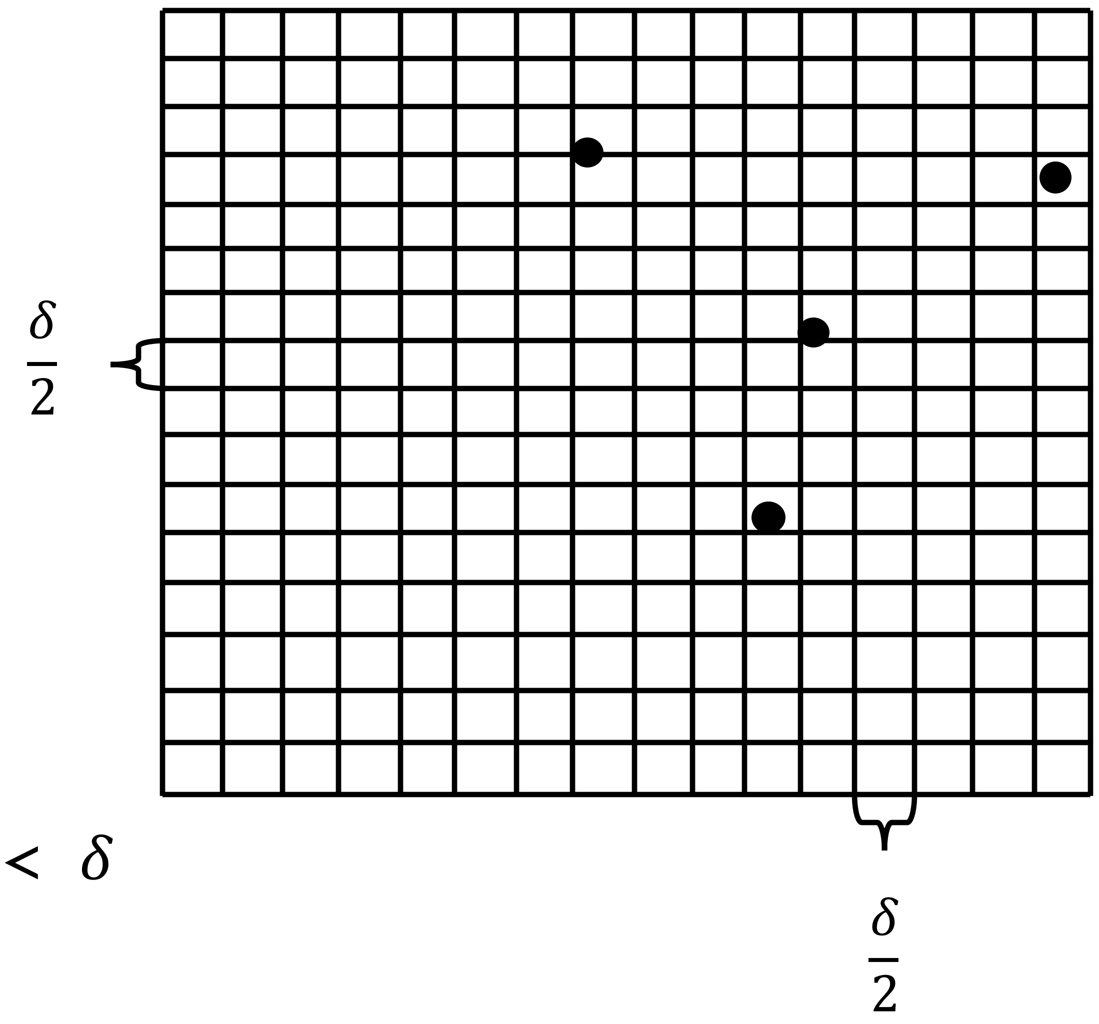
```
findClosestPair(on sequence P)
   Order points in random sequence
   δ = d(p1, p2)
   Make map of cell size δ/2
   for i = 1, 2,..., n
      Determine cell containing pi
      Lookup 25 cells surrounding pi
      Find dist to surrounding points
      if ∃ pj such that δ′ = d(pi, pj) < δ
         Delete map
         Make map of cell size δ′/2
```



$$(\boldsymbol{x}, \boldsymbol{y}) \rightarrow \left(\left\lfloor \frac{\boldsymbol{y}}{\boldsymbol{\delta}/2} \right\rfloor, \left\lfloor \frac{\boldsymbol{x}}{\boldsymbol{\delta}/2} \right\rfloor \right)$$

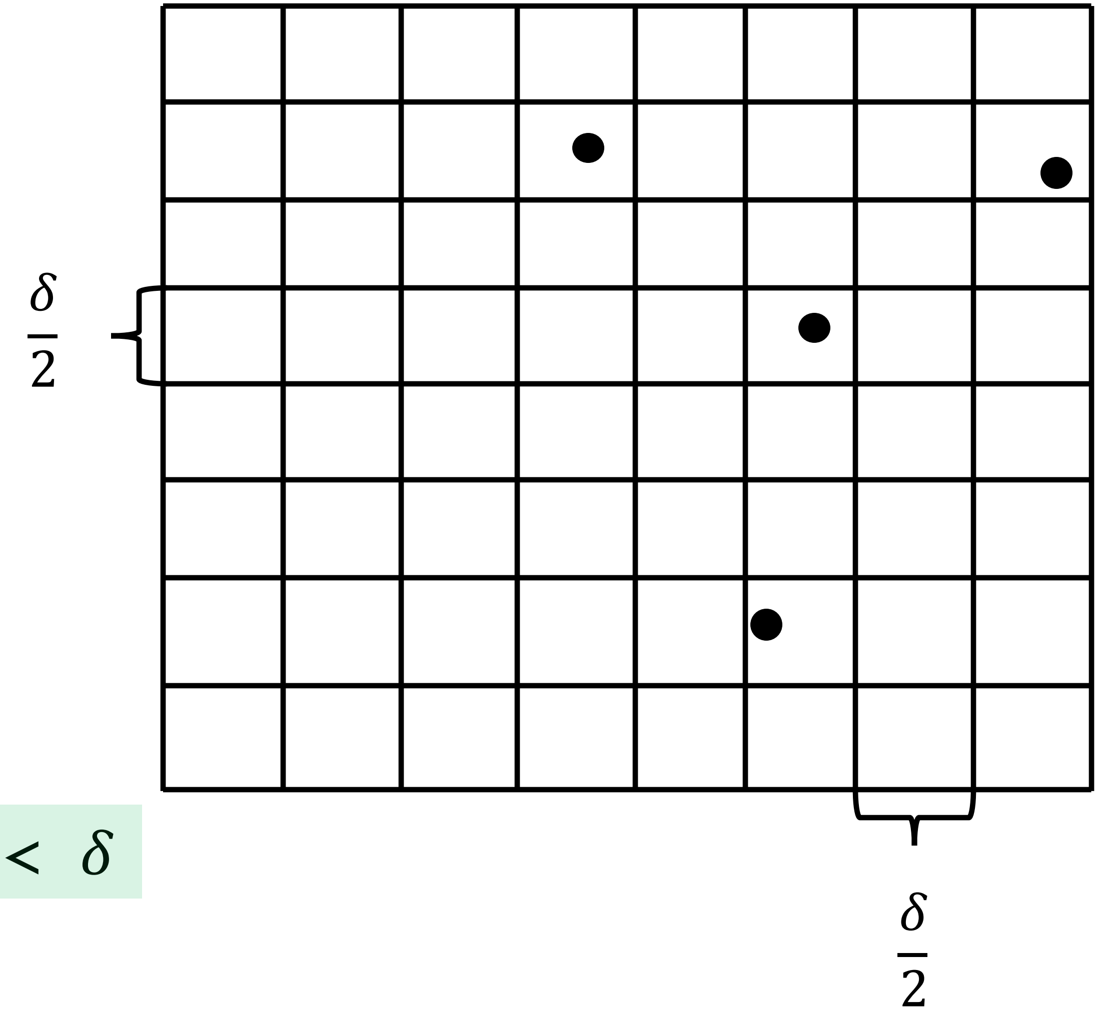
Closest Pair Problem

findClosestPair(on sequence P)
   Order points in random sequence
   $\delta = d(p_1, p_2)$
   Make map of cell size $\delta/2$
   for $i = 1, 2, \ldots, n$
      Determine cell containing p_i
      Lookup 25 cells surrounding p_i
      Find dist to surrounding points
      if $\exists\, p_j$ such that $\delta' = d(p_i, p_j) < \delta$
         Delete map
         Make map of cell size $\delta'/2$
         for $p_k = p_1, p_2, \ldots, p_i$
            Insert cell containing p_k into map



Closest Pair Problem

```
findClosestPair(on sequence P)
   Order points in random sequence
   δ = d(p₁, p₂)
   Make map of cell size δ/2
   for i = 1, 2,..., n
      Determine cell containing pᵢ
      Lookup 25 cells surrounding pᵢ
      Find dist to surrounding points
      if ∃ pⱼ such that δ′ = d(pᵢ, pⱼ) < δ
         Delete map
         Make map of cell size δ′/2
         for pₖ = p₁, p₂,...,pᵢ
            Insert cell containing pₖ into map
      else
         Insert cell containing pᵢ into map
```



Closest Pair Problem

findClosestPair(on sequence P)

  Order points in random sequence Optimal?

  $\delta = d(p_1, p_2)$

  Make map of cell size $\delta/2$

  for i = 1, 2,..., n

    Determine cell containing p_i

    Lookup 25 cells surrounding p_i

    Find dist to surrounding points

    if $\exists\ p_j$ such that $\delta' = d(p_i, p_j) < \delta$

      Delete map

      Make map of cell size $\delta'/2$

      for $p_k = p_1, p_2, \ldots, p_i$

        Insert cell containing p_k into map

    else

      Insert cell containing p_i into map

Closest Pair Problem

```
findClosestPair(on sequence P)
   Order points in random sequence
   δ = d(p₁, p₂)
   Make map of cell size δ/2
   for i = 1, 2,..., n
      Determine cell containing pᵢ
      Lookup 25 cells surrounding pᵢ
      Find dist to surrounding points
      if ∃ pⱼ such that δ′ = d(pᵢ, pⱼ) < δ
         Delete map
         Make map of cell size δ′/2
         for pₖ = p₁, p₂,...,pᵢ
            Insert cell containing pₖ into map
      else
         Insert cell containing pᵢ into map
```

Optimal?

Yes – It considers (though not necessarily computes) every pair of points.

Closest Pair Problem

```
findClosestPair(on sequence P)
   Order points in random sequence
   δ = d(p₁, p₂)
   Make map of cell size δ/2
   for i = 1, 2,..., n
      Determine cell containing pᵢ
      Lookup 25 cells surrounding pᵢ
      Find dist to surrounding points
      if ∃ pⱼ such that δ′ = d(pᵢ, pⱼ) < δ
         Delete map
         Make map of cell size δ′/2
         for pₖ = p₁, p₂,...,pᵢ
            Insert cell containing pₖ into map
      else
         Insert cell containing pᵢ into map
```

Running Time.

Closest Pair Problem

```
findClosestPair(on sequence P)
   Order points in random sequence
   δ = d(p₁, p₂)
   Make map of cell size δ/2
   for i = 1, 2,..., n
      Determine cell containing pᵢ
      Lookup 25 cells surrounding pᵢ
      Find dist to surrounding points
      if ∃ pⱼ such that δ′ = d(pᵢ, pⱼ) < δ
         Delete map
         Make map of cell size δ′/2
         for pₖ = p₁, p₂,...,pᵢ
            Insert cell containing pₖ into map
      else
         Insert cell containing pᵢ into map
```

Running Time.

For a single value of `i`, what is the running time of this block?

Closest Pair Problem

```
findClosestPair(on sequence P)
   Order points in random sequence
   δ = d(p_1, p_2)
   Make map of cell size δ/2
   for i = 1, 2,..., n
      Determine cell containing p_i
      Lookup 25 cells surrounding p_i
      Find dist to surrounding points
      if ∃ p_j such that δ' = d(p_i, p_j) < δ
         Delete map
         Make map of cell size δ'/2
         for p_k = p_1, p_2,...,p_i
            Insert cell containing p_k into map
      else
         Insert cell containing p_i into map
```

Running Time.

For a single value of **i**, what is the running time of this block?

$$O(1)$$

$$(x, y) \rightarrow \left(\left\lfloor \frac{y}{\delta/2} \right\rfloor, \left\lfloor \frac{x}{\delta/2} \right\rfloor \right)$$

```
for r = row - 2 to row + 2
  for c = col - 2 to col + 2
    //check for entry (r,c)
```

Closest Pair Problem

```
findClosestPair(on sequence P)
   Order points in random sequence
   δ = d(p₁, p₂)
   Make map of cell size δ/2
   for i = 1, 2,..., n
      Determine cell containing pᵢ
      Lookup 25 cells surrounding pᵢ
      Find dist to surrounding points
      if ∃ pⱼ such that δ′ = d(pᵢ, pⱼ) < δ
         Delete map
         Make map of cell size δ′/2
         for pₖ = p₁, p₂,...,pᵢ
            Insert cell containing pₖ into map
      else
         Insert cell containing pᵢ into map
```

Running Time.

What is the worst-case number of insertion operations?

Closest Pair Problem

```
findClosestPair(on sequence P)
   Order points in random sequence
   δ = d(p₁, p₂)
   Make map of cell size δ/2
   for i = 1, 2,..., n
      Determine cell containing pᵢ
      Lookup 25 cells surrounding pᵢ
      Find dist to surrounding points
      if ∃ pⱼ such that δ′ = d(pᵢ, pⱼ) < δ
         Delete map
         Make map of cell size δ′/2
         for pₖ = p₁, p₂,...,pᵢ
            Insert cell containing pₖ into map
      else
         Insert cell containing pᵢ into map
```

Running Time.

What is the worst-case number of insertion operations?

$$O(n^2)$$

If adding every new point decreases δ, every previous point needs to be reinserted.

Closest Pair Problem

```
findClosestPair(on sequence P)
   Order points in random sequence
   δ = d(p₁, p₂)
   Make map of cell size δ/2
   for i = 1, 2,..., n
      Determine cell containing pᵢ
      Lookup 25 cells surrounding pᵢ
      Find dist to surrounding points
      if ∃ pⱼ such that δ′ = d(pᵢ, pⱼ) < δ
         Delete map
         Make map of cell size δ′/2
         for pₖ = p₁, p₂,...,pᵢ
            Insert cell containing pₖ into map
      else
         Insert cell
```

Worst-case Running Time: $O(n^2)$

Running Time.

What is the worst-case number of insertion operations?

$$O(n^2)$$

If adding every new point decreases δ, every previous point needs to be reinserted.

Closest Pair Problem

```
findClosestPair(on sequence P)
   Order points in random sequence
   δ = d(p₁, p₂)
   Make map of cell size δ/2
   for i = 1, 2,..., n
      Determine cell containing pᵢ
      Lookup 25 cells surrounding pᵢ
      Find dist to surrounding points
      if ∃ pⱼ such that δ′ = d(pᵢ, pⱼ) < δ
         Delete map
         Make map of cell size δ′/2
         for pₖ = p₁, p₂,...,pᵢ
            Insert cell containing pₖ into map
      else
         Insert cell
```

Expected Running Time: ??

Running Time.

What is the worst-case number of insertion operations?

$$O(n^2)$$

If adding every new point decreases δ, every previous point needs to be reinserted.

Closest Pair Problem

We want to determine the expected number of insertion operations.

- I.e., How many times are we likely to insert values into a map?
- This is the running time of the algorithm.

Closest Pair Problem

We want to determine the expected number of insertion operations.

Let $X_i = \begin{cases} 1, \text{ if } p_i \text{ changed } \delta \\ 0, \text{ otherwise} \end{cases}$

Iteration i

```
for i = 1, 2,..., n
   Determine cell containing pi
   Lookup 25 cells surrounding pi
   Find dist to surrounding points
   if ∃ pj such that δ′ = d(pi, pj) < δ
      Delete map
      Make map of cell size δ′/2
      for pk = p1, p2,...,pi
         Insert cell containing pk into map
   else
      Insert cell containing pi into map
```

Closest Pair Problem

We want to determine the expected number of insertion operations.

Let $X_i = \begin{cases} 1, \text{ if } p_i \text{ changed } \delta \\ 0, \text{ otherwise} \end{cases}$ $\Rightarrow$ # Insertions = ?

Iteration i

Total # of insertions for the full algorithm?

```
for i = 1, 2,..., n
   Determine cell containing pi
   Lookup 25 cells surrounding pi
   Find dist to surrounding points
   if ∃ pj such that δ′ = d(pi, pj) < δ
      Delete map
      Make map of cell size δ′/2
      for pk = p1, p2,...,pi
         Insert cell containing pk into map
   else
      Insert cell containing pi into map
```

Closest Pair Problem

We want to determine the expected number of insertion operations.

Let $X_i = \begin{cases} 1, \text{if } p_i \text{ changed } \delta \\ 0, \text{otherwise} \end{cases}$ $\Rightarrow$ # Insertions $= n$

Iteration i

Each point is inserted once when it is first encountered.

```
for i = 1, 2,..., n
   Determine cell containing pᵢ
   Lookup 25 cells surrounding pᵢ
   Find dist to surrounding points
   if ∃ pⱼ such that δ′ = d(pᵢ, pⱼ) < δ
      Delete map
      Make map of cell size δ′/2
      for pₖ = p₁, p₂,...,pᵢ
         Insert cell containing pₖ into map
   else
      Insert cell containing pᵢ into map
```

Closest Pair Problem

We want to determine the expected number of insertion operations.

Let $X_i = \begin{cases} 1, \text{ if } p_i \text{ changed } \delta \\ 0, \text{ otherwise} \end{cases}$ $\Rightarrow$ # Insertions $= n + \sum_{i=1}^{n}(i-1)X_i$

Iteration i

$i-1$ points are reinserted if δ changes in iteration i.

```
for i = 1, 2,..., n
    Determine cell containing p_i
    Lookup 25 cells surrounding p_i
    Find dist to surrounding points
    if ∃ p_j such that δ′ = d(p_i, p_j) < δ
        Delete map
        Make map of cell size δ′/2
        for p_k = p_1, p_2,...,p_i
            Insert cell containing p_k into map
    else
        Insert cell containing p_i into map
```

Closest Pair Problem

We want to determine the expected number of insertion operations.

Let $X_i = \begin{cases} 1, \text{ if } p_i \text{ changed } \delta \\ 0, \text{ otherwise} \end{cases}$ $\Rightarrow$ # Insertions $= n + \sum_{i=1}^{n} (i-1) X_i$

Consider the random sequence of points $p_1, \ldots, p_i$.

Closest Pair Problem

We want to determine the expected number of insertion operations.

Let $X_i = \begin{cases} 1, \text{ if } p_i \text{ changed } \delta \\ 0, \text{ otherwise} \end{cases}$ $\Rightarrow$ # Insertions $= n + \sum_{i=1}^{n} (i-1) X_i$

Consider the random sequence of points $p_1, \ldots, p_i$. Suppose $\delta = d(p, q)$.

Closest Pair Problem

We want to determine the expected number of insertion operations.

Let $X_i = \begin{cases} 1, \text{ if } p_i \text{ changed } \delta \\ 0, \text{ otherwise} \end{cases}$ $\Rightarrow$ # Insertions $= n + \sum_{i=1}^{n} (i-1) X_i$

Consider the random sequence of points $p_1, \ldots, p_i$. Suppose $\delta = d(p, q)$.
If δ decreased in iteration i, where must p or q be in the sequence?

Closest Pair Problem

We want to determine the expected number of insertion operations.

Let $X_i = \begin{cases} 1, \text{ if } p_i \text{ changed } \delta \\ 0, \text{ otherwise} \end{cases}$ $\Rightarrow$ # Insertions $= n + \sum_{i=1}^{n}(i-1)X_i$

Consider the random sequence of points $p_1, \dots, p_i$. Suppose $\delta = d(p, q)$.
If δ decreased in iteration i, where must p or q be in the sequence?

Last.

If p and q at the closest pair, and iteration i triggers that discovery, then p or q must have arrived at iteration i.

Closest Pair Problem

We want to determine the expected number of insertion operations.

Let $X_i = \begin{cases} 1, \text{ if } p_i \text{ changed } \delta \\ 0, \text{ otherwise} \end{cases}$ $\Rightarrow$ # Insertions $= n + \sum_{i=1}^{n}(i-1)X_i$

Consider the random sequence of points $p_1, \dots, p_i$. Suppose $\delta = d(p, q)$.
For δ to decrease in iteration i, p or q must be last in the sequence.

Closest Pair Problem

We want to determine the expected number of insertion operations.

Let $X_i = \begin{cases} 1, \text{ if } p_i \text{ changed } \delta \\ 0, \text{ otherwise} \end{cases}$ $\Rightarrow$ # Insertions $= n + \sum_{i=1}^{n}(i-1)X_i$

Consider the random sequence of points $p_1, \ldots, p_i$. Suppose $\delta = d(p, q)$.
For δ to decrease in iteration i, p or q must be last in the sequence.

What is the probability that p or q are the last point in the sequence?

I.e., What is the probability that either of two specific points are last in a random list of i points?

Closest Pair Problem

We want to determine the expected number of insertion operations.

Let $X_i = \begin{cases} 1, \text{ if } p_i \text{ changed } \delta \\ 0, \text{ otherwise} \end{cases}$ $\Rightarrow$ # Insertions $= n + \sum_{i=1}^{n}(i-1)X_i$

Consider the random sequence of points $p_1, \dots, p_i$. Suppose $\delta = d(p, q)$.
For δ to decrease in iteration i, p or q must be last in the sequence.

What is the probability that p or q are the last point in the sequence?

Suppose I have a random permutation of {1,2,3}, what is the likelihood that 2 is last?

{1,2,3}, {1,3,2}, {2,1,3}, {2,3,1}, {3,1,2}, {3,2,1}

Closest Pair Problem

We want to determine the expected number of insertions operations.

Let $X_i = \begin{cases} 1, \text{ if } p_i \text{ changed } \delta \\ 0, \text{ otherwise} \end{cases}$ $\Rightarrow$ # Insertions $= n + \sum_{i=1}^{n}(i-1)X_i$

Consider the random sequence of points $p_1, \dots, p_i$. Suppose $\delta = d(p, q)$.
For δ to decrease in iteration i, p or q must be last in the sequence.

What is the probability that p or q are the last point in the sequence?

Suppose I have a random permutation of {1,2,3}, what is the likelihood that 2 is last?

{1,2,3}, {1,3,2}, {2,1,3}, {2,3,1}, {3,1,2}, {3,2,1}

$$= \frac{1}{3}$$

Closest Pair Problem

We want to determine the expected number of insertion operations.

Let $X_i = \begin{cases} 1, \text{ if } p_i \text{ changed } \delta \\ 0, \text{ otherwise} \end{cases}$ $\Rightarrow$ # Insertions $= n + \sum_{i=1}^{n} (i-1) X_i$

Consider the random sequence of points $p_1, \dots, p_i$. Suppose $\delta = d(p,q)$. For δ to decrease in iteration i, p or q must be last in the sequence.

What is the probability that p or q are the last point in the sequence?

Suppose I have a random permutation of {1,2,3}, what is the likelihood that 2 or 3 is last?

{1,2,3}, {1,3,2}, {2,1,3}, {2,3,1}, {3,1,2}, {3,2,1}

Closest Pair Problem

We want to determine the expected number of insertion operations.

Let $X_i = \begin{cases} 1, \text{ if } p_i \text{ changed } \delta \\ 0, \text{ otherwise} \end{cases}$ $\Rightarrow$ # Insertions $= n + \sum_{i=1}^{n}(i-1)X_i$

Consider the random sequence of points $p_1, \dots, p_i$. Suppose $\delta = d(p, q)$.
For δ to decrease in iteration i, p or q must be last in the sequence.

What is the probability that p or q are the last point in the sequence?

Suppose I have a random permutation of {1,2,3}, what is the likelihood that 2 or 3 is last?

{1,2,3}, {1,3,2}, {2,1,3}, {2,3,1}, {3,1,2}, {3,2,1}

$$= \frac{1}{3} + \frac{1}{3} = \frac{2}{3}$$

Closest Pair Problem

We want to determine the expected number of insertions operations.

Let $X_i = \begin{cases} 1, \text{ if } p_i \text{ changed } \delta \\ 0, \text{ otherwise} \end{cases}$ $\Rightarrow$ # Insertions $= n + \sum_{i=1}^{n} (i-1) X_i$

Consider the random sequence of points $p_1, \dots, p_i$. Suppose $\delta = d(p, q)$.
For δ to decrease in iteration i, p or q must be last in the sequence.

What is the probability that p or q are the last point in the sequence?

Suppose I have a random permutation of i elements, what is the likelihood that two specific ones are last?

Closest Pair Problem

We want to determine the expected number of insertion operations.

Let $X_i = \begin{cases} 1, \text{ if } p_i \text{ changed } \delta \\ 0, \text{ otherwise} \end{cases}$ $\Rightarrow$ # Insertions $= n + \sum_{i=1}^{n}(i-1)X_i$

Consider the random sequence of points $p_1, \dots, p_i$. Suppose $\delta = d(p, q)$.
For δ to decrease in iteration i, p or q must be last in the sequence.

What is the probability that p or q are the last point in the sequence?

Suppose I have a random permutation of i elements, what is the likelihood that two specific ones are last? $= \frac{2}{i}$

Closest Pair Problem

We want to determine the expected number of insertion operations.

Let $X_i = \begin{cases} 1, \text{ if } p_i \text{ changed } \delta \\ 0, \text{ otherwise} \end{cases}$ $\Rightarrow$ # Insertions $= n + \sum_{i=1}^{n} (i-1) X_i$

Consider the random sequence of points $p_1, \ldots, p_i$. Suppose $\delta = d(p, q)$.
For δ to decrease in iteration i, p or q must be last in the sequence.

The probability that p or q are the last point in the sequence is $\frac{2}{i}$.

Closest Pair Problem

We want to determine the expected number of insertion operations.

Let $X_i = \begin{cases} 1, \text{ if } p_i \text{ changed } \delta \\ 0, \text{ otherwise} \end{cases}$ $\Rightarrow$ # Insertions $= n + \sum_{i=1}^{n}(i-1)X_i$

Consider the random sequence of points $p_1, \ldots, p_i$. Suppose $\delta = d(p, q)$. For δ to decrease in iteration i, p or q must be last in the sequence.

The probability that p or q are the last point in the sequence is $\frac{2}{i}$.

Thus, the probability that δ decreased in iteration $i \leq \frac{2}{i}$

It could be $< \frac{2}{i}$ since there may be multiple pairs of points in the first i values that define δ, and those pairs would have been found earlier than iteration i.

Closest Pair Problem

We want to determine the expected number of insertions operations.

Let $X_i = \begin{cases} 1, \text{ if } p_i \text{ changed } \delta \\ 0, \text{ otherwise} \end{cases}$ $\Rightarrow$ # Insertions $= n + \sum_{i=1}^{n} (i-1) X_i$

Consider the random sequence of points $p_1, \dots, p_i$. Suppose $\delta = d(p, q)$.
For δ to decrease in iteration i, p or q must be last in the sequence.

The probability that p or q are the last point in the sequence is $\frac{2}{i}$.
Thus, the probability that δ decreased in iteration $i \leq \frac{2}{i}$

Then, E[# Insertions]

Closest Pair Problem

We want to determine the expected number of insertions operations.

Let $X_i = \begin{cases} 1, \text{ if } p_i \text{ changed } \delta \\ 0, \text{ otherwise} \end{cases}$ $\Rightarrow$ # Insertions $= n + \sum_{i=1}^{n}(i-1)X_i$

Consider the random sequence of points $p_1, \ldots, p_i$. Suppose $\delta = d(p, q)$.
For δ to decrease in iteration i, p or q must be last in the sequence.

The probability that p or q are the last point in the sequence is $\frac{2}{i}$.
Thus, the probability that δ decreased in iteration $i \leq \frac{2}{i}$

Then, $E[\text{\# Insertions}] = E[n + \sum_{i=1}^{n}(i-1)X_i]$

Closest Pair Problem

We want to determine the expected number of insertions operations.

Let $X_i = \begin{cases} 1, \text{ if } p_i \text{ changed } \delta \\ 0, \text{ otherwise} \end{cases}$ $\Rightarrow$ # Insertions $= n + \sum_{i=1}^{n}(i-1)X_i$

Consider the random sequence of points $p_1, \dots, p_i$. Suppose $\delta = d(p, q)$.
For δ to decrease in iteration i, p or q must be last in the sequence.

The probability that p or q are the last point in the sequence is $\frac{2}{i}$.
Thus, the probability that δ decreased in iteration $i \leq \frac{2}{i}$

Then, $E[\text{\# Insertions}] = E[n + \sum_{i=1}^{n}(i-1)X_i] = n + \sum_{i=1}^{n}(i-1)E[X_i]$

"Linearity of Expectation"

Closest Pair Problem

We want to determine the expected number of insertions operations.

Let $X_i = \begin{cases} 1, \text{ if } p_i \text{ changed } \delta \\ 0, \text{ otherwise} \end{cases}$ $\Rightarrow$ # Insertions $= n + \sum_{i=1}^{n}(i-1)X_i$

Consider the random sequence of points $p_1, \ldots, p_i$. Suppose $\delta = d(p, q)$.
For δ to decrease in iteration i, p or q must be last in the sequence.

The probability that p or q are the last point in the sequence is $\frac{2}{i}$.
Thus, the probability that δ decreased in iteration $i \leq \frac{2}{i}$

Then, $E[\text{\# Insertions}] = E[n + \sum_{i=1}^{n}(i-1)X_i] = n + \sum_{i=1}^{n}(i-1)E[X_i]$

Closest Pair Problem

We want to determine the expected number of insertion operations.

Let $X_i = \begin{cases} 1, \text{ if } p_i \text{ changed } \delta \\ 0, \text{ otherwise} \end{cases}$ $\Rightarrow$ # Insertions $= n + \sum_{i=1}^{n}(i-1)X_i$

Consider the random sequence of points $p_1, \ldots, p_i$. Suppose $\delta = d(p, q)$.
For δ to decrease in iteration i, p or q must be last in the sequence.

The probability that p or q are the last point in the sequence is $\frac{2}{i}$.

Thus, the probability that δ decreased in iteration $i \leq \frac{2}{i}$

Then, $E[\text{\# Insertions}] = E[n + \sum_{i=1}^{n}(i-1)X_i] = n + \sum_{i=1}^{n}(i-1)E[X_i]$

Since X_i is a 0-1 variable, its excepted value is the probability it equals 1.

Closest Pair Problem

We want to determine the expected number of insertion operations.

Let $X_i = \begin{cases} 1, \text{ if } p_i \text{ changed } \delta \\ 0, \text{ otherwise} \end{cases}$ $\Rightarrow$ # Insertions $= n + \sum_{i=1}^{n}(i-1)X_i$

Consider the random sequence of points $p_1, \ldots, p_i$. Suppose $\delta = d(p, q)$.
For δ to decrease in iteration i, p or q must be last in the sequence.

The probability that p or q are the last point in the sequence is $\frac{2}{i}$.

$\mathbf{Pr}[X_i = 1] =$ probability that δ decreased in iteration $i \leq \frac{2}{i}$

Then, $E[\text{\# Insertions}] = E[n + \sum_{i=1}^{n}(i-1)X_i] = n + \sum_{i=1}^{n}(i-1)E[X_i]$

Since X_i is a 0-1 variable, its excepted value is the probability it equals 1.

Closest Pair Problem

We want to determine the expected number of insertion operations.

Let $X_i = \begin{cases} 1, \text{ if } p_i \text{ changed } \delta \\ 0, \text{ otherwise} \end{cases}$ $\Rightarrow$ # Insertions $= n + \sum_{i=1}^{n} (i-1) X_i$

Consider the random sequence of points $p_1, \dots, p_i$. Suppose $\delta = d(p, q)$.
For δ to decrease in iteration i, p or q must be last in the sequence.

The probability that p or q are the last point in the sequence is $\frac{2}{i}$.

$\mathbf{Pr}[X_i = 1] =$ probability that δ decreased in iteration $i \leq \frac{2}{i}$

Then, $E[\text{\# Insertions}] = E[n + \sum_{i=1}^{n} (i-1) X_i] = n + \sum_{i=1}^{n} (i-1) E[X_i]$

$\leq n + \sum_{i=1}^{n} (i-1) \frac{2}{i}$

Since X_i is a 0-1 variable, its excepted value is the probability it equals 1.

Closest Pair Problem

We want to determine the expected number of insertion operations.

Let $X_i = \begin{cases} 1, \text{ if } p_i \text{ changed } \delta \\ 0, \text{ otherwise} \end{cases}$ $\Rightarrow$ # Insertions $= n + \sum_{i=1}^{n}(i-1)X_i$

Consider the random sequence of points $p_1, \dots, p_i$. Suppose $\delta = d(p, q)$.
For δ to decrease in iteration i, p or q must be last in the sequence.

The probability that p or q are the last point in the sequence is $\frac{2}{i}$.
Thus, the probability that δ decreased in iteration $i \leq \frac{2}{i}$

Then, E[# Insertions] $= E[n + \sum_{i=1}^{n}(i-1)X_i] = n + \sum_{i=1}^{n}(i-1)E[X_i]$

$$\leq n + \sum_{i=1}^{n}(i-1)\frac{2}{i} \leq n + \sum_{i=1}^{n} i\frac{2}{i}$$

Closest Pair Problem

We want to determine the expected number of insertion operations.

Let $X_i = \begin{cases} 1, \text{ if } p_i \text{ changed } \delta \\ 0, \text{ otherwise} \end{cases}$ $\Rightarrow$ # Insertions $= n + \sum_{i=1}^{n}(i-1)X_i$

Consider the random sequence of points $p_1, \ldots, p_i$. Suppose $\delta = d(p, q)$.
For δ to decrease in iteration i, p or q must be last in the sequence.

The probability that p or q are the last point in the sequence is $\frac{2}{i}$.
Thus, the probability that δ decreased in iteration $i \leq \frac{2}{i}$

Then, $E[\text{\# Insertions}] = E[n + \sum_{i=1}^{n}(i-1)X_i] = n + \sum_{i=1}^{n}(i-1)E[X_i]$

$$\leq n + \sum_{i=1}^{n}(i-1)\frac{2}{i} \leq n + \sum_{i=1}^{n} i\frac{2}{i} = n + 2\sum_{i=1}^{n} 1$$

Closest Pair Problem

We want to determine the expected number of insertions operations.

Let $X_i = \begin{cases} 1, \text{ if } p_i \text{ changed } \delta \\ 0, \text{ otherwise} \end{cases}$ $\Rightarrow$ # Insertions $= n + \sum_{i=1}^{n}(i-1)X_i$

Consider the random sequence of points $p_1, \ldots, p_i$. Suppose $\delta = d(p,q)$.
For δ to decrease in iteration i, p or q must be last in the sequence.

The probability that p or q are the last point in the sequence is $\frac{2}{i}$.
Thus, the probability that δ decreased in iteration $i \leq \frac{2}{i}$

Then, E[# Insertions] $= E[n + \sum_{i=1}^{n}(i-1)X_i] = n + \sum_{i=1}^{n}(i-1)E[X_i]$

$$\leq n + \sum_{i=1}^{n}(i-1)\frac{2}{i} \leq n + \sum_{i=1}^{n} i\frac{2}{i} = n + 2\sum_{i=1}^{n} 1$$

$$= n + 2n = 3n$$

Closest Pair Problem

```
findClosestPair(on sequence P)
   Order points in random sequence
   δ = d(p_1, p_2)
   Make map of cell size δ/2
   for i = 1, 2,..., n
      Determine cell containing p_i
      Lookup 25 cells surrounding p_i
      Find dist to surrounding points
      if ∃ p_j such that δ′ = d(p_i, p_j) < δ
         Delete map
         Make map of cell size δ′/2
         for p_k = p_1, p_2,...,p_i
            Insert cell containing p_k into map
      else
         Insert cell containing p_i into map
```

Running Time.

Worst Case:

$O(n^2)$

Expected Case:

$O(n)$

Closest Pair Problem

```
findClosestPair(on sequence P)
   Order points in random sequence
   δ = d(p₁, p₂)
   Make map of cell size δ/2
   for i = 1, 2,..., n
      Determine cell containing pᵢ
      Lookup 25 cells surrounding pᵢ
      Find dist to surrounding points
      if ∃ pⱼ such that δ′ = d(pᵢ, pⱼ) < δ
         Delete map
         Make map of cell size δ′/2
         for pₖ = p₁, p₂,...,pᵢ
            Insert cell containing pₖ into map
      else
         Insert cell containing pᵢ into map
```

Running Time.

Worst Case:

$O(n^2)$

Expected Case:

$O(n)$

Would need to argue this can be done in $O(1)$ time (i.e., not many collisions).