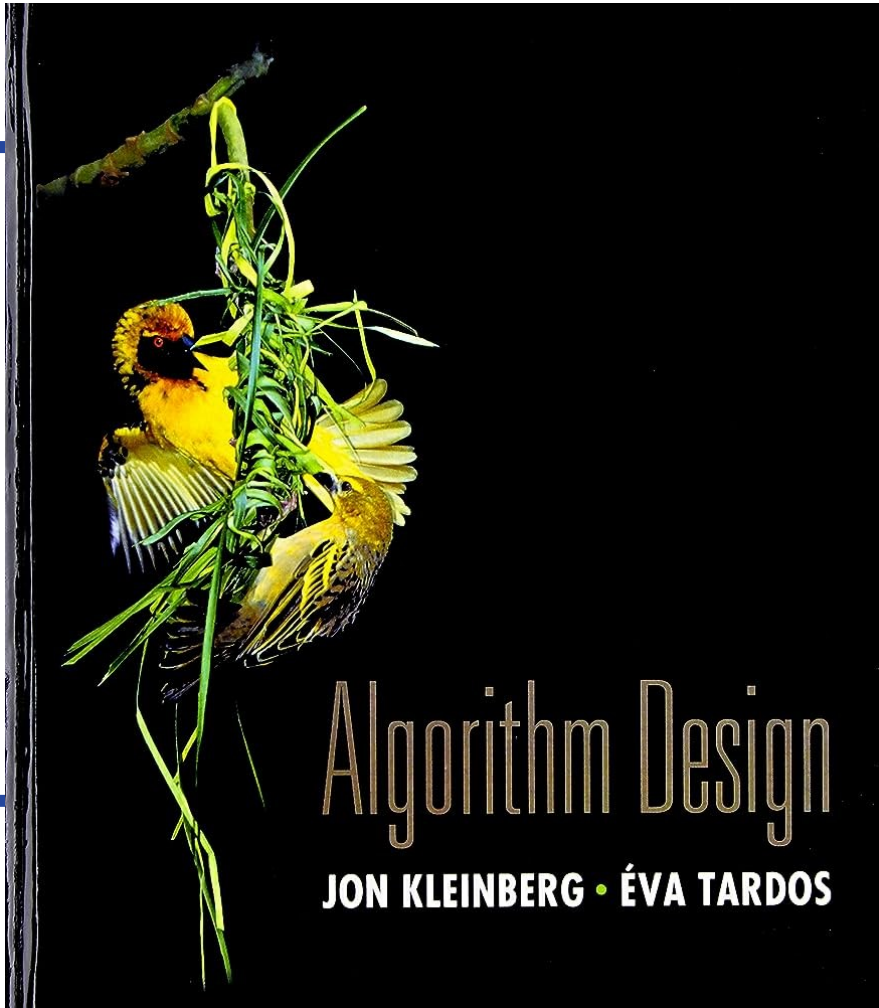


CS 310: Algorithms

# Lecture 11

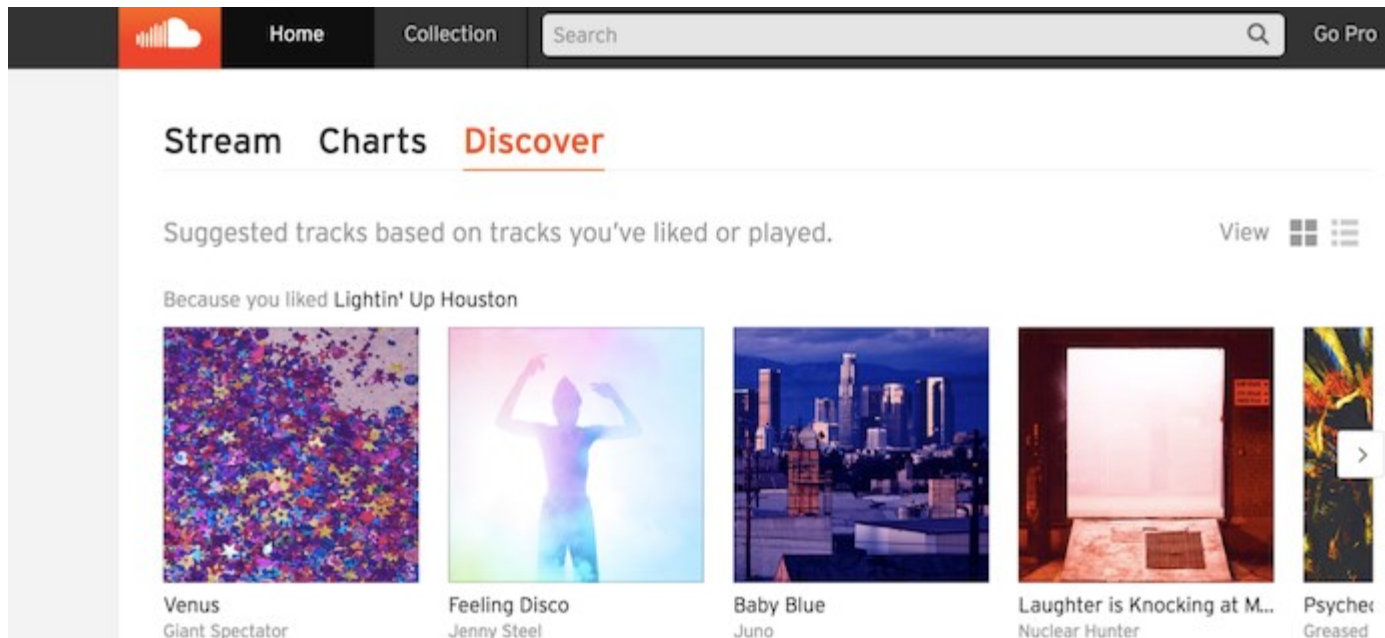
**Instructor:** Naveed Anwar Bhatti



# Chapter 5: Divide and Conquer

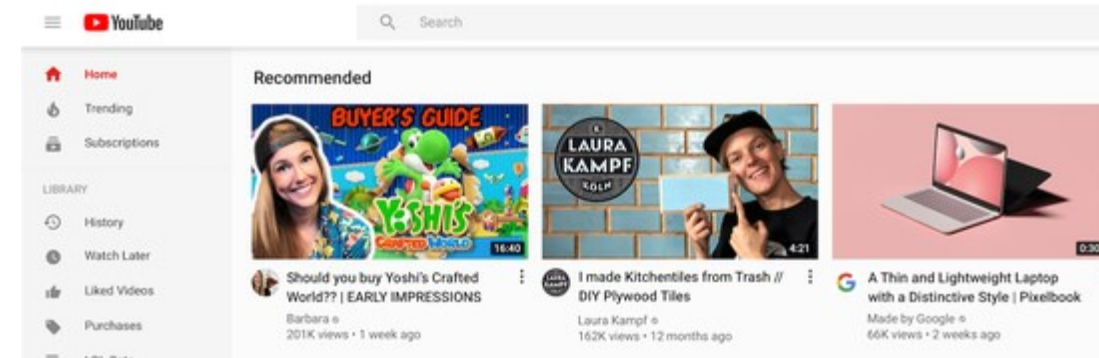
## Section 5.3: Counting Inversions

# Motivation: Recommendation Systems



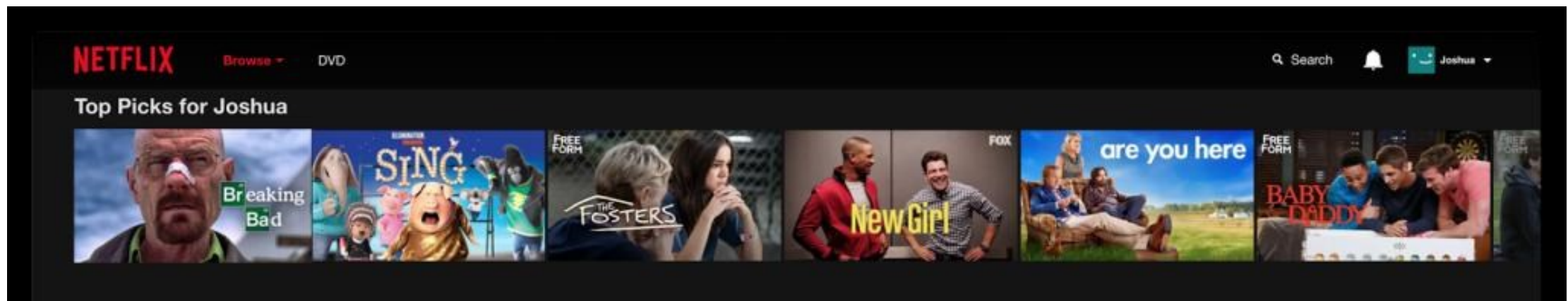
The Spotify Discover page features a navigation bar with 'Home', 'Collection', and a search bar. Below the navigation bar, there are tabs for 'Stream', 'Charts', and 'Discover'. The main content area is titled 'Suggested tracks based on tracks you've liked or played.' and includes a sub-header 'Because you liked Lightin' Up Houston'. Five track recommendations are displayed in a row, each with a cover image and title:

- Venus by Giant Spectator
- Feeling Disco by Jenny Steel
- Baby Blue by Juno
- Laughter is Knocking at M... by Nuclear Hunter
- Psyched by Greased



The YouTube Recommended page shows a navigation bar with 'Home', 'Trending', and 'Subscriptions'. Below the navigation bar, there are tabs for 'Home', 'Trending', and 'Subscriptions'. The main content area is titled 'Recommended' and displays three video recommendations:

- Should you buy Yoshi's Crafted World?? | EARLY IMPRESSIONS by Barbara (201K views • 1 week ago)
- I made Kitchentiles from Trash // DIY Plywood Tiles by Laura Kampf (162K views • 12 months ago)
- A Thin and Lightweight Laptop with a Distinctive Style | Pixelbook by Made by Google (66K views • 2 weeks ago)



The Netflix interface shows the 'NETFLIX' logo and a navigation bar with 'Browse' and 'DVD'. Below the navigation bar, there is a search bar and a user profile icon labeled 'Joshua'. The main content area is titled 'Top Picks for Joshua' and displays a row of recommended titles:

- Breaking Bad
- Sing
- The Fosters
- New Girl
- are you here
- Baby Daddy



# Motivation: Recommendation Systems



1



2

2



4

3



1

4



3

5



5



# Counting inversions



1

2

3

4

5



2

4

1

3

5

(2,1)



# Counting inversions



1



2

2



4

3



1

4



3

5



5

(2,1) (4,1)

# Counting inversions



1

2

3

4

5



2

4

1

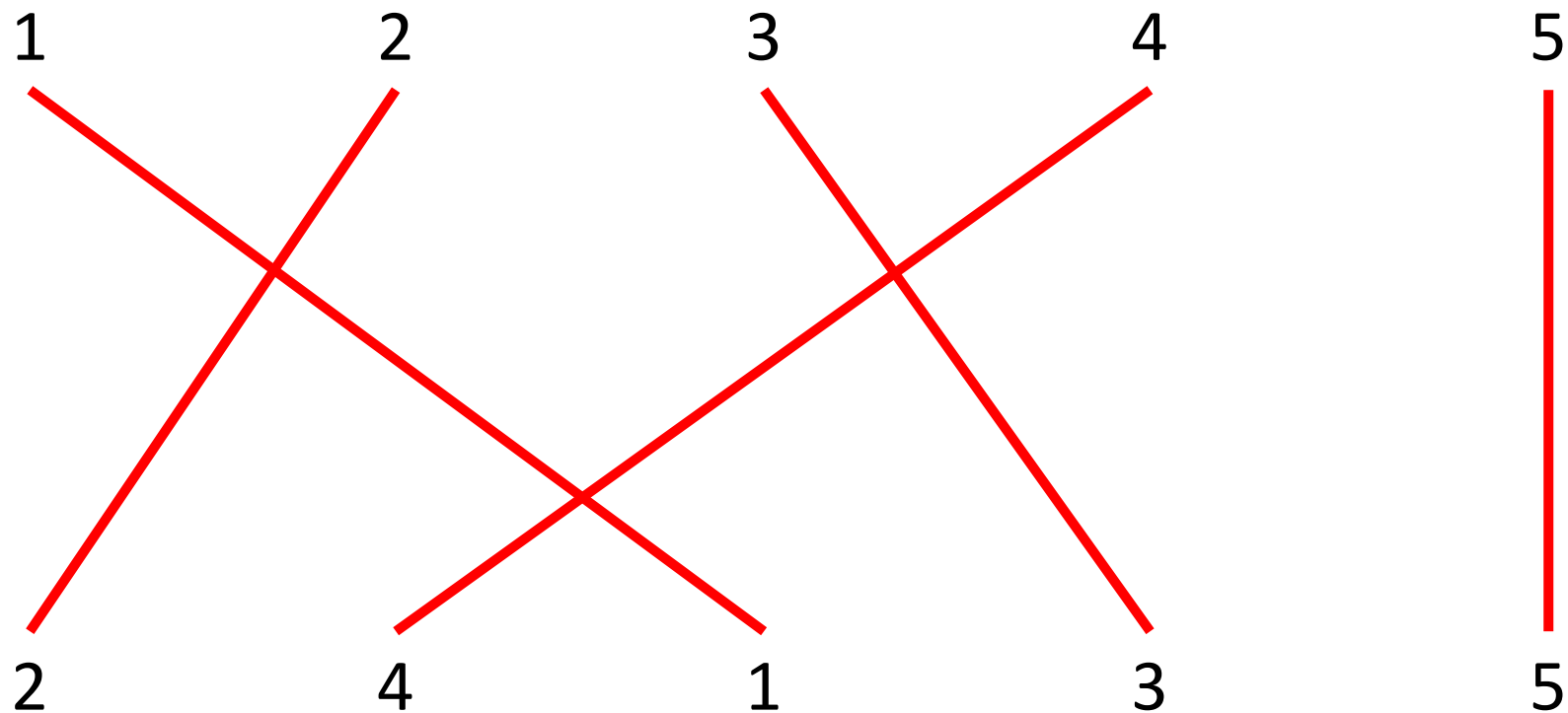
3

5

(2,1) (4,1) (4,3)

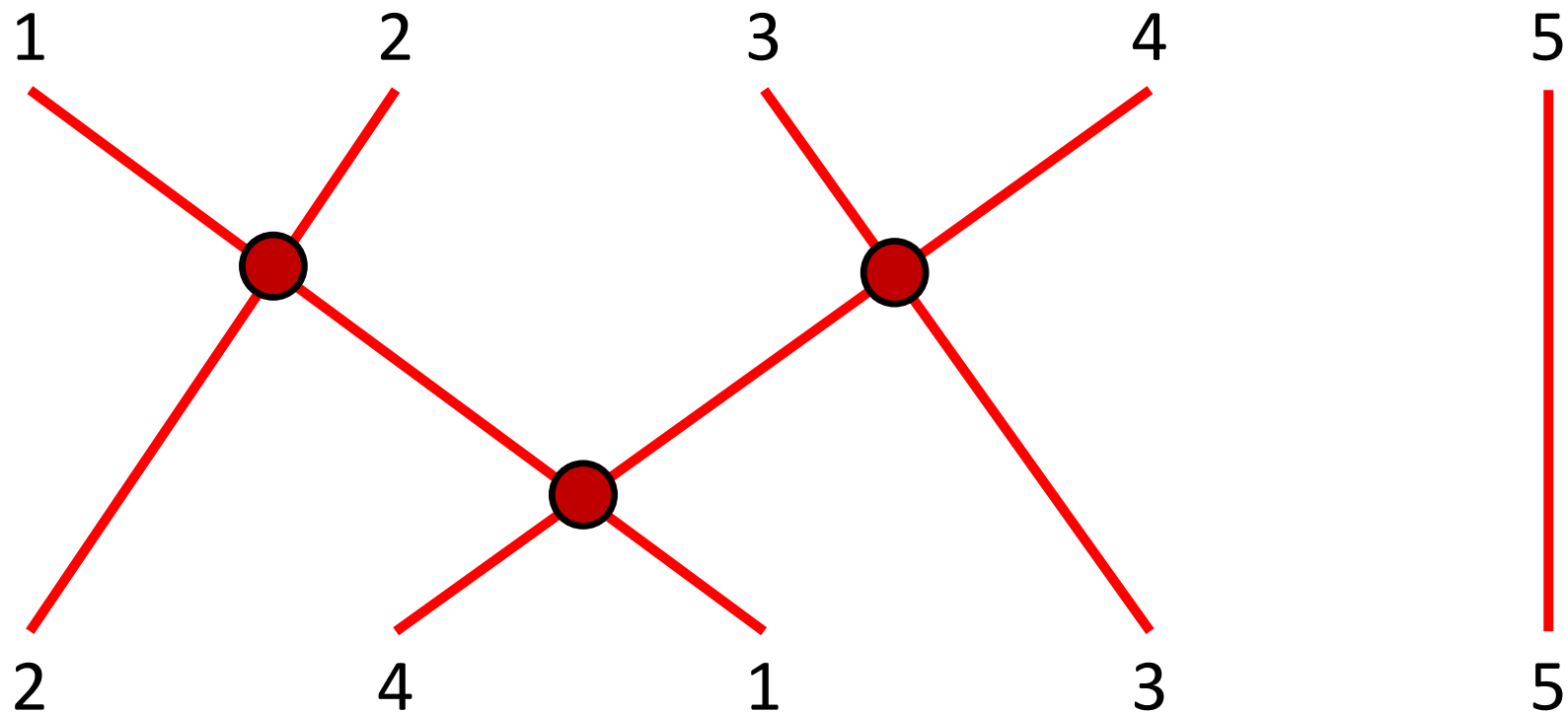


# Counting inversions





# Counting inversions



# Motivation: Recommendation Systems



1



2

2



4

3



1

4



3

5



5





# Counting inversions: Divide and Conquer (Warm Up!)

- Divide: separate list into two halves  $A$  and  $B$ .
- Conquer: recursively count inversions in each list.
- Combine: count inversions  $(a, b)$  with  $a \in A$  and  $b \in B$ .
- Return sum of three counts.

3	7	11	5	9	1	6	13	4	2	15	8	10	14	12	16
---	---	----	---	---	---	---	----	---	---	----	---	----	----	----	----

$(3,1)(7,5)(7,1)(7,6)(11,5)(11,9)(11,1)(11,6)(5,1)(9,1)(9,6) = 11$

3	7	11	5	9	1	6	13
---	---	----	---	---	---	---	----

$(4,2)(15,8)(15,10)(15,14)(15,12)(14,12) = 6$

4	2	15	8	10	14	12	16
---	---	----	---	----	----	----	----

$(3,2)(7,4)(7,2)(11,4)(11,2)(11,8)(11,10)(5,4)(5,2)(9,4)(9,2)(9,8)(6,4)(6,2)(13,4)(13,2)(13,8)(13,10)(13,12) = 19$

3	7	11	5	9	1	6	13
---	---	----	---	---	---	---	----

4	2	15	8	10	14	12	16
---	---	----	---	----	----	----	----

**Return = 11 + 6 + 19 = 36**

# Counting inversions: how to combine two subproblems?

Q. How to count inversions  $(a, b)$  with  $a \in A$  and  $b \in B$  *efficiently*?

A. Easy if A and B are sorted!

3	7	11	5	9	1	6	13
---	---	----	---	---	---	---	----

4	2	15	8	10	14	12	16
---	---	----	---	----	----	----	----



# Counting inversions: how to combine two subproblems?

Q. How to count inversions  $(a, b)$  with  $a \in A$  and  $b \in B$ ?

A. Easy if  $A$  and  $B$  are sorted!

## Algorithm:

- Scan  $A$  and  $B$  from left to right.
- Compare  $a_i$  and  $b_j$ .
- If  $a_i < b_j$ , then  $a_i$  is not inverted with any element left in  $B$ .
- If  $a_i > b_j$ , then  $b_j$  is inverted with every element left in  $A$ .
- Append smaller element to sorted list  $C$ .

3	7	11	5	9	1	6	13
---	---	----	---	---	---	---	----

1	3	5	6	7	9	11	13
---	---	---	---	---	---	----	----

$\uparrow$   
 $a_i$

4	2	15	8	10	14	12	16
---	---	----	---	----	----	----	----

2	4	8	10	12	14	15	16
---	---	---	----	----	----	----	----

$\uparrow$   
 $b_j$

# Counting inversions: how to combine two subproblems?

Q. How to count inversions (a, b) with  $a \in A$  and  $b \in B$ ?

A. Easy if A and B are sorted!

## Algorithm:

- Scan A and B from left to right.
- Compare  $a_i$  and  $b_j$ .
- If  $a_i < b_j$ , then  $a_i$  is not inverted with any element left in B.
- If  $a_i > b_j$ , then  $b_j$  is inverted with every element left in A.
- Append smaller element to sorted list C.

3	7	11	5	9	1	6	13
---	---	----	---	---	---	---	----

4	2	15	8	10	14	12	16
---	---	----	---	----	----	----	----

1	3	5	6	7	9	11	13
---	---	---	---	---	---	----	----

$a_i$

Inversions=0

2	4	8	10	12	14	15	16
---	---	---	----	----	----	----	----

$b_j$

--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--



# Counting inversions: how to combine two subproblems?

Q. How to count inversions (a, b) with  $a \in A$  and  $b \in B$ ?

A. Easy if A and B are sorted!

## Algorithm:

- Scan A and B from left to right.
- Compare  $a_i$  and  $b_j$ .
- If  $a_i < b_j$ , then  $a_i$  is not inverted with any element left in B.
- If  $a_i > b_j$ , then  $b_j$  is inverted with every element left in A.
- Append smaller element to sorted list C.

3	7	11	5	9	1	6	13
---	---	----	---	---	---	---	----

4	2	15	8	10	14	12	16
---	---	----	---	----	----	----	----

1	3	5	6	7	9	11	13
---	---	---	---	---	---	----	----

Inversions=0

2	4	8	10	12	14	15	16
---	---	---	----	----	----	----	----

$a_i$

$b_j$

1															
---	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--

# Counting inversions: how to combine two subproblems?

Q. How to count inversions  $(a, b)$  with  $a \in A$  and  $b \in B$ ?

A. Easy if  $A$  and  $B$  are sorted!

## Algorithm:

- Scan  $A$  and  $B$  from left to right.
- Compare  $a_i$  and  $b_j$ .
- If  $a_i < b_j$ , then  $a_i$  is not inverted with any element left in  $B$ .
- If  $a_i > b_j$ , then  $b_j$  is inverted with every element left in  $A$ .
- Append smaller element to sorted list  $C$ .

3	7	11	5	9	1	6	13
---	---	----	---	---	---	---	----

4	2	15	8	10	14	12	16
---	---	----	---	----	----	----	----

1	3	5	6	7	9	11	13
---	---	---	---	---	---	----	----

$a_i$

Inversions=0

2	4	8	10	12	14	15	16
---	---	---	----	----	----	----	----

$b_j$

1															
---	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--



# Counting inversions: how to combine two subproblems?

Q. How to count inversions (a, b) with  $a \in A$  and  $b \in B$ ?

A. Easy if A and B are sorted!

## Algorithm:

- Scan A and B from left to right.
- Compare  $a_i$  and  $b_j$ .
- If  $a_i < b_j$ , then  $a_i$  is not inverted with any element left in B.
- If  $a_i > b_j$ , then  $b_j$  is inverted with every element left in A.
- Append smaller element to sorted list C.

3	7	11	5	9	1	6	13
---	---	----	---	---	---	---	----

1	3	5	6	7	9	11	13
---	---	---	---	---	---	----	----

$a_i$

Inversions=7

4	2	15	8	10	14	12	16
---	---	----	---	----	----	----	----

7

2	4	8	10	12	14	15	16
---	---	---	----	----	----	----	----

$b_j$

1	2														
---	---	--	--	--	--	--	--	--	--	--	--	--	--	--	--

# Counting inversions: how to combine two subproblems?

Q. How to count inversions (a, b) with  $a \in A$  and  $b \in B$ ?

A. Easy if A and B are sorted!

## Algorithm:

- Scan A and B from left to right.
- Compare  $a_i$  and  $b_j$ .
- If  $a_i < b_j$ , then  $a_i$  is not inverted with any element left in B.
- If  $a_i > b_j$ , then  $b_j$  is inverted with every element left in A.
- Append smaller element to sorted list C.

3	7	11	5	9	1	6	13
---	---	----	---	---	---	---	----

1	3	5	6	7	9	11	13
---	---	---	---	---	---	----	----

$a_i$

4	2	15	8	10	14	12	16
---	---	----	---	----	----	----	----

7

2	4	8	10	12	14	15	16
---	---	---	----	----	----	----	----

$b_j$

Inversions=7

1	2	3													
---	---	---	--	--	--	--	--	--	--	--	--	--	--	--	--

# Counting inversions: how to combine two subproblems?

Q. How to count inversions (a, b) with  $a \in A$  and  $b \in B$ ?

A. Easy if A and B are sorted!

## Algorithm:

- Scan A and B from left to right.
- Compare  $a_i$  and  $b_j$ .
- If  $a_i < b_j$ , then  $a_i$  is not inverted with any element left in B.
- If  $a_i > b_j$ , then  $b_j$  is inverted with every element left in A.
- Append smaller element to sorted list C.

3	7	11	5	9	1	6	13
---	---	----	---	---	---	---	----

1	3	5	6	7	9	11	13
---	---	---	---	---	---	----	----

$a_i$

Inversions=7

4	2	15	8	10	14	12	16
---	---	----	---	----	----	----	----

7

2	4	8	10	12	14	15	16
---	---	---	----	----	----	----	----

$b_j$

1	2	3													
---	---	---	--	--	--	--	--	--	--	--	--	--	--	--	--



# Counting inversions: how to combine two subproblems?

Q. How to count inversions (a, b) with  $a \in A$  and  $b \in B$ ?

A. Easy if A and B are sorted!

## Algorithm:

- Scan A and B from left to right.
- Compare  $a_i$  and  $b_j$ .
- If  $a_i < b_j$ , then  $a_i$  is not inverted with any element left in B.
- If  $a_i > b_j$ , then  $b_j$  is inverted with every element left in A.
- Append smaller element to sorted list C.

3	7	11	5	9	1	6	13
---	---	----	---	---	---	---	----

1	3	5	6	7	9	11	13
---	---	---	---	---	---	----	----

$a_i$

4	2	15	8	10	14	12	16
---	---	----	---	----	----	----	----

7 6

2	4	8	10	12	14	15	16
---	---	---	----	----	----	----	----

$b_j$

Inversions=13

1	2	3	4												
---	---	---	---	--	--	--	--	--	--	--	--	--	--	--	--

# Counting inversions: how to combine two subproblems?

Q. How to count inversions (a, b) with  $a \in A$  and  $b \in B$ ?

A. Easy if A and B are sorted!

## Algorithm:

- Scan A and B from left to right.
- Compare  $a_i$  and  $b_j$ .
- If  $a_i < b_j$ , then  $a_i$  is not inverted with any element left in B.
- If  $a_i > b_j$ , then  $b_j$  is inverted with every element left in A.
- Append smaller element to sorted list C.

3	7	11	5	9	1	6	13
---	---	----	---	---	---	---	----

1	3	5	6	7	9	11	13
---	---	---	---	---	---	----	----

$a_i$

4	2	15	8	10	14	12	16
---	---	----	---	----	----	----	----

7 6

2	4	8	10	12	14	15	16
---	---	---	----	----	----	----	----

$b_j$

Inversions=13

1	2	3	4	5										
---	---	---	---	---	--	--	--	--	--	--	--	--	--	--

# Counting inversions: how to combine two subproblems?

Q. How to count inversions (a, b) with  $a \in A$  and  $b \in B$ ?

A. Easy if A and B are sorted!

## Algorithm:

- Scan A and B from left to right.
- Compare  $a_i$  and  $b_j$ .
- If  $a_i < b_j$ , then  $a_i$  is not inverted with any element left in B.
- If  $a_i > b_j$ , then  $b_j$  is inverted with every element left in A.
- Append smaller element to sorted list C.

3	7	11	5	9	1	6	13
---	---	----	---	---	---	---	----

1	3	5	6	7	9	11	13
---	---	---	---	---	---	----	----

$a_i$

4	2	15	8	10	14	12	16
---	---	----	---	----	----	----	----

7 6

2	4	8	10	12	14	15	16
---	---	---	----	----	----	----	----

$b_j$

Inversions=13

1	2	3	4	5	6										
---	---	---	---	---	---	--	--	--	--	--	--	--	--	--	--



# Counting inversions: how to combine two subproblems?

Q. How to count inversions (a, b) with  $a \in A$  and  $b \in B$ ?

A. Easy if A and B are sorted!

## Algorithm:

- Scan A and B from left to right.
- Compare  $a_i$  and  $b_j$ .
- If  $a_i < b_j$ , then  $a_i$  is not inverted with any element left in B.
- If  $a_i > b_j$ , then  $b_j$  is inverted with every element left in A.
- Append smaller element to sorted list C.

3	7	11	5	9	1	6	13
---	---	----	---	---	---	---	----

1	3	5	6	7	9	11	13
---	---	---	---	---	---	----	----

$a_i$

Inversions=13

4	2	15	8	10	14	12	16
---	---	----	---	----	----	----	----

7 6

2	4	8	10	12	14	15	16
---	---	---	----	----	----	----	----

$b_j$

1	2	3	4	5	6	7								
---	---	---	---	---	---	---	--	--	--	--	--	--	--	--

# Counting inversions: how to combine two subproblems?

Q. How to count inversions (a, b) with  $a \in A$  and  $b \in B$ ?

A. Easy if A and B are sorted!

## Algorithm:

- Scan A and B from left to right.
- Compare  $a_i$  and  $b_j$ .
- If  $a_i < b_j$ , then  $a_i$  is not inverted with any element left in B.
- If  $a_i > b_j$ , then  $b_j$  is inverted with every element left in A.
- Append smaller element to sorted list C.

3	7	11	5	9	1	6	13
---	---	----	---	---	---	---	----

1	3	5	6	7	9	11	13
---	---	---	---	---	---	----	----

$a_i$

Inversions=13

4	2	15	8	10	14	12	16
---	---	----	---	----	----	----	----

7 6

2	4	8	10	12	14	15	16
---	---	---	----	----	----	----	----

$b_j$

1	2	3	4	5	6	7									
---	---	---	---	---	---	---	--	--	--	--	--	--	--	--	--

# Counting inversions: how to combine two subproblems?

Q. How to count inversions (a, b) with  $a \in A$  and  $b \in B$ ?

A. Easy if A and B are sorted!

## Algorithm:

- Scan A and B from left to right.
- Compare  $a_i$  and  $b_j$ .
- If  $a_i < b_j$ , then  $a_i$  is not inverted with any element left in B.
- If  $a_i > b_j$ , then  $b_j$  is inverted with every element left in A.
- Append smaller element to sorted list C.

3	7	11	5	9	1	6	13
---	---	----	---	---	---	---	----

1	3	5	6	7	9	11	13
---	---	---	---	---	---	----	----

↑  
 $a_i$

Inversions=16

4	2	15	8	10	14	12	16
---	---	----	---	----	----	----	----

7 6 3

2	4	8	10	12	14	15	16
---	---	---	----	----	----	----	----

↑  
 $b_j$

1	2	3	4	5	6	7	8								
---	---	---	---	---	---	---	---	--	--	--	--	--	--	--	--



# Counting inversions: how to combine two subproblems?

Q. How to count inversions (a, b) with  $a \in A$  and  $b \in B$ ?

A. Easy if A and B are sorted!

## Algorithm:

- Scan A and B from left to right.
- Compare  $a_i$  and  $b_j$ .
- If  $a_i < b_j$ , then  $a_i$  is not inverted with any element left in B.
- If  $a_i > b_j$ , then  $b_j$  is inverted with every element left in A.
- Append smaller element to sorted list C.

3	7	11	5	9	1	6	13
---	---	----	---	---	---	---	----

1	3	5	6	7	9	11	13
---	---	---	---	---	---	----	----

$a_i$

Inversions=16

4	2	15	8	10	14	12	16
---	---	----	---	----	----	----	----

7 6 3

2	4	8	10	12	14	15	16
---	---	---	----	----	----	----	----

$b_j$

1	2	3	4	5	6	7	8	9							
---	---	---	---	---	---	---	---	---	--	--	--	--	--	--	--

# Counting inversions: how to combine two subproblems?

Q. How to count inversions (a, b) with  $a \in A$  and  $b \in B$ ?

A. Easy if A and B are sorted!

## Algorithm:

- Scan A and B from left to right.
- Compare  $a_i$  and  $b_j$ .
- If  $a_i < b_j$ , then  $a_i$  is not inverted with any element left in B.
- If  $a_i > b_j$ , then  $b_j$  is inverted with every element left in A.
- Append smaller element to sorted list C.

3	7	11	5	9	1	6	13
---	---	----	---	---	---	---	----

1	3	5	6	7	9	11	13
---	---	---	---	---	---	----	----

$a_i$

Inversions=16

4	2	15	8	10	14	12	16
---	---	----	---	----	----	----	----

7 6 3

2	4	8	10	12	14	15	16
---	---	---	----	----	----	----	----

$b_j$

1	2	3	4	5	6	7	8	9							
---	---	---	---	---	---	---	---	---	--	--	--	--	--	--	--

# Counting inversions: how to combine two subproblems?

Q. How to count inversions (a, b) with  $a \in A$  and  $b \in B$ ?

A. Easy if A and B are sorted!

## Algorithm:

- Scan A and B from left to right.
- Compare  $a_i$  and  $b_j$ .
- If  $a_i < b_j$ , then  $a_i$  is not inverted with any element left in B.
- If  $a_i > b_j$ , then  $b_j$  is inverted with every element left in A.
- Append smaller element to sorted list C.

3	7	11	5	9	1	6	13
---	---	----	---	---	---	---	----

1	3	5	6	7	9	11	13
---	---	---	---	---	---	----	----

$a_i$

Inversions=18

4	2	15	8	10	14	12	16
---	---	----	---	----	----	----	----

7 6 3 2

2	4	8	10	12	14	15	16
---	---	---	----	----	----	----	----

$b_j$

1	2	3	4	5	6	7	8	9	10						
---	---	---	---	---	---	---	---	---	----	--	--	--	--	--	--

# Counting inversions: how to combine two subproblems?

Q. How to count inversions (a, b) with  $a \in A$  and  $b \in B$ ?

A. Easy if A and B are sorted!

## Algorithm:

- Scan A and B from left to right.
- Compare  $a_i$  and  $b_j$ .
- If  $a_i < b_j$ , then  $a_i$  is not inverted with any element left in B.
- If  $a_i > b_j$ , then  $b_j$  is inverted with every element left in A.
- Append smaller element to sorted list C.

3	7	11	5	9	1	6	13
---	---	----	---	---	---	---	----

1	3	5	6	7	9	11	13
---	---	---	---	---	---	----	----

$a_i$

Inversions=18

4	2	15	8	10	14	12	16
---	---	----	---	----	----	----	----

7 6 3 2

2	4	8	10	12	14	15	16
---	---	---	----	----	----	----	----

$b_j$

1	2	3	4	5	6	7	8	9	10	11					
---	---	---	---	---	---	---	---	---	----	----	--	--	--	--	--



# Counting inversions: how to combine two subproblems?

Q. How to count inversions (a, b) with  $a \in A$  and  $b \in B$ ?

A. Easy if A and B are sorted!

## Algorithm:

- Scan A and B from left to right.
- Compare  $a_i$  and  $b_j$ .
- If  $a_i < b_j$ , then  $a_i$  is not inverted with any element left in B.
- If  $a_i > b_j$ , then  $b_j$  is inverted with every element left in A.
- Append smaller element to sorted list C.

3	7	11	5	9	1	6	13
---	---	----	---	---	---	---	----

1	3	5	6	7	9	11	13
---	---	---	---	---	---	----	----

$\uparrow$   
 $a_i$

Inversions=19

4	2	15	8	10	14	12	16
---	---	----	---	----	----	----	----

7 6 3 2 1

2	4	8	10	12	14	15	16
---	---	---	----	----	----	----	----

$\uparrow$   
 $b_j$

1	2	3	4	5	6	7	8	9	10	11	12				
---	---	---	---	---	---	---	---	---	----	----	----	--	--	--	--

# Counting inversions: how to combine two subproblems?

Q. How to count inversions  $(a, b)$  with  $a \in A$  and  $b \in B$ ?

A. Easy if  $A$  and  $B$  are sorted!

## Algorithm:

- Scan  $A$  and  $B$  from left to right.
- Compare  $a_i$  and  $b_j$ .
- If  $a_i < b_j$ , then  $a_i$  is not inverted with any element left in  $B$ .
- If  $a_i > b_j$ , then  $b_j$  is inverted with every element left in  $A$ .
- Append smaller element to sorted list  $C$ .

3	7	11	5	9	1	6	13
---	---	----	---	---	---	---	----

1	3	5	6	7	9	11	13
---	---	---	---	---	---	----	----

$\uparrow$   
 $a_i$

Inversions=19

4	2	15	8	10	14	12	16
---	---	----	---	----	----	----	----

7 6 3 2 1

2	4	8	10	12	14	15	16
---	---	---	----	----	----	----	----

$\uparrow$   
 $b_j$

1	2	3	4	5	6	7	8	9	10	11	12	13			
---	---	---	---	---	---	---	---	---	----	----	----	----	--	--	--

# Counting inversions: how to combine two subproblems?

Q. How to count inversions  $(a, b)$  with  $a \in A$  and  $b \in B$ ?

A. Easy if  $A$  and  $B$  are sorted!

## Algorithm:

- Scan  $A$  and  $B$  from left to right.
- Compare  $a_i$  and  $b_j$ .
- If  $a_i < b_j$ , then  $a_i$  is not inverted with any element left in  $B$ .
- If  $a_i > b_j$ , then  $b_j$  is inverted with every element left in  $A$ .
- Append smaller element to sorted list  $C$ .

3	7	11	5	9	1	6	13
---	---	----	---	---	---	---	----

1	3	5	6	7	9	11	13
---	---	---	---	---	---	----	----

$\uparrow$   
 $a_i$

Inversions=19

4	2	15	8	10	14	12	16
---	---	----	---	----	----	----	----

7 6 3 2 1 0

2	4	8	10	12	14	15	16
---	---	---	----	----	----	----	----

$\uparrow$   
 $b_j$

1	2	3	4	5	6	7	8	9	10	11	12	13	14		
---	---	---	---	---	---	---	---	---	----	----	----	----	----	--	--

# Counting inversions: how to combine two subproblems?

Q. How to count inversions (a, b) with  $a \in A$  and  $b \in B$ ?

A. Easy if A and B are sorted!

## Algorithm:

- Scan A and B from left to right.
- Compare  $a_i$  and  $b_j$ .
- If  $a_i < b_j$ , then  $a_i$  is not inverted with any element left in B.
- If  $a_i > b_j$ , then  $b_j$  is inverted with every element left in A.
- Append smaller element to sorted list C.

3	7	11	5	9	1	6	13
---	---	----	---	---	---	---	----

1	3	5	6	7	9	11	13
---	---	---	---	---	---	----	----

$\uparrow$   
 $a_i$

Inversions=19

4	2	15	8	10	14	12	16
---	---	----	---	----	----	----	----

7 6 3 2 1 0 0

2	4	8	10	12	14	15	16
---	---	---	----	----	----	----	----

$\uparrow$   
 $b_j$

1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	
---	---	---	---	---	---	---	---	---	----	----	----	----	----	----	--

# Counting inversions: how to combine two subproblems?

Q. How to count inversions (a, b) with  $a \in A$  and  $b \in B$ ?

A. Easy if A and B are sorted!

## Algorithm:

- Scan A and B from left to right.
- Compare  $a_i$  and  $b_j$ .
- If  $a_i < b_j$ , then  $a_i$  is not inverted with any element left in B.
- If  $a_i > b_j$ , then  $b_j$  is inverted with every element left in A.
- Append smaller element to sorted list C.

3	7	11	5	9	1	6	13
---	---	----	---	---	---	---	----

1	3	5	6	7	9	11	13
---	---	---	---	---	---	----	----

↑  
 $a_i$

Inversions=19

4	2	15	8	10	14	12	16
---	---	----	---	----	----	----	----

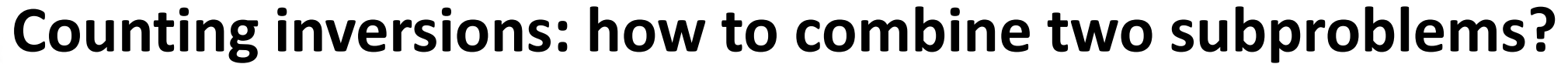
7 6 3 2 1 0 0 0

2	4	8	10	12	14	15	16
---	---	---	----	----	----	----	----

↑  
 $b_j$

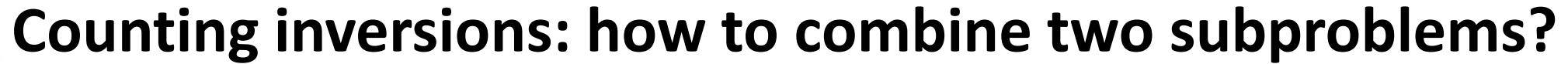
1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
---	---	---	---	---	---	---	---	---	----	----	----	----	----	----	----





```
13 return (A, inversions)
```

 $x=8$ 



```
13 return (A, inversions)
```

 $x=7$ 



```
13 return (A, inversions)
```

 $x=7$ 


$$1 \quad m \leftarrow \text{length}(A_1) + \text{length}(A_2)$$

*inversions=7*

 $x=6$ 

3	7	11	5	9	1	6	13
---	---	----	---	---	---	---	----

4	2	15	8	10	14	12	16
---	---	----	---	----	----	----	----

1	3	5	6	7	9	11	13
---	---	---	---	---	---	----	----

2	4	8	10	12	14	15	16
---	---	---	----	----	----	----	----

$$\uparrow$$
  

$$a_i$$
$$\uparrow$$
  

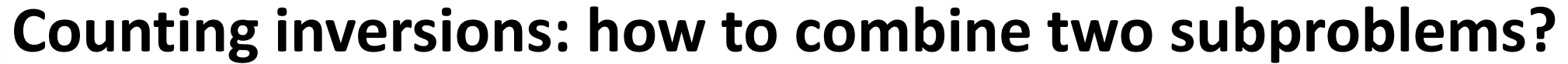
$$b_j$$
[illegible]

```

12      inversions ← inversions + x

```

```
13 return (A, inversions)
```


$$1 \quad m \leftarrow \text{length}(A_1) + \text{length}(A_2)$$

*inversions=13*

 $x=6$ 

3	7	11	5	9	1	6	13
---	---	----	---	---	---	---	----

4	2	15	8	10	14	12	16
---	---	----	---	----	----	----	----

1	3	5	6	7	9	11	13
---	---	---	---	---	---	----	----

2	4	8	10	12	14	15	16
---	---	---	----	----	----	----	----

$$\uparrow$$
  

$$a_i$$
$$\overline{b_j}$$
[illegible]

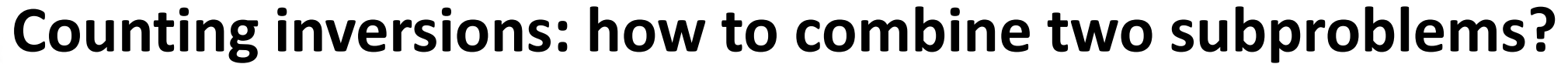
```

12      inversions ← inversions + x

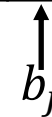
```

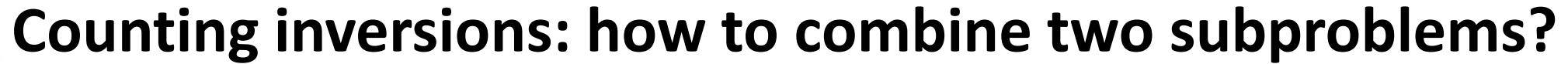
```
13 return (A, inversions)
```



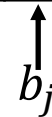


```
13 return (A, inversions)
```

 $x=5$ 



```
13 return (A, inversions)
```

 $x=4$ 



# Counting inversions: how to combine two subproblems?

*Merge – And – Count*( $A_1, A_2$ )

1  $m \leftarrow \text{length}(A_1) + \text{length}(A_2)$

2  $x \leftarrow \text{length}(A_1)$

3  $\text{inversions} \leftarrow 0$

4  $i \leftarrow 1; j \leftarrow 1$

5 **for**  $k = 1$  **to**  $m$

6     **if**  $A_1[i] \leq A_2[j]$

7          $A[k] \leftarrow A_1[i]$

8          $i \leftarrow i + 1$

9          $x \leftarrow x - 1$

10     **else**  $A[k] \leftarrow A_2[j]$

11          $j \leftarrow j + 1$

12          $\text{inversions} \leftarrow \text{inversions} + x$

13 **return** ( $A, \text{inversions}$ )

*inversions*=16

*x*=3

3	7	11	5	9	1	6	13
---	---	----	---	---	---	---	----

4	2	15	8	10	14	12	16
---	---	----	---	----	----	----	----

1	3	5	6	7	9	11	13
---	---	---	---	---	---	----	----

2	4	8	10	12	14	15	16
---	---	---	----	----	----	----	----

$\uparrow$   
 $a_i$

$\uparrow$   
 $b_j$

1	2	3	4	5	6	7	8								
---	---	---	---	---	---	---	---	--	--	--	--	--	--	--	--

# Counting inversions: how to combine two subproblems?

*Merge – And – Count*( $A_1, A_2$ )

1  $m \leftarrow \text{length}(A_1) + \text{length}(A_2)$

2  $x \leftarrow \text{length}(A_1)$

3  $\text{inversions} \leftarrow 0$

4  $i \leftarrow 1; j \leftarrow 1$

5 **for**  $k = 1$  **to**  $m$

6     **if**  $A_1[i] \leq A_2[j]$

7          $A[k] \leftarrow A_1[i]$

8          $i \leftarrow i + 1$

9          $x \leftarrow x - 1$

10     **else**  $A[k] \leftarrow A_2[j]$

11          $j \leftarrow j + 1$

12          $\text{inversions} \leftarrow \text{inversions} + x$

13 **return** ( $A, \text{inversions}$ )

$\text{inversions} = 16$

$x = 2$

3	7	11	5	9	1	6	13
---	---	----	---	---	---	---	----

4	2	15	8	10	14	12	16
---	---	----	---	----	----	----	----

1	3	5	6	7	9	11	13
---	---	---	---	---	---	----	----

2	4	8	10	12	14	15	16
---	---	---	----	----	----	----	----

$\uparrow$   
 $a_i$

$\uparrow$   
 $b_j$

1	2	3	4	5	6	7	8	9							
---	---	---	---	---	---	---	---	---	--	--	--	--	--	--	--



# Counting inversions: how to combine two subproblems?

*Merge – And – Count*( $A_1, A_2$ )

1  $m \leftarrow \text{length}(A_1) + \text{length}(A_2)$

2  $x \leftarrow \text{length}(A_1)$

3  $\text{inversions} \leftarrow 0$

4  $i \leftarrow 1; j \leftarrow 1$

5 **for**  $k = 1$  **to**  $m$

6     **if**  $A_1[i] \leq A_2[j]$

7          $A[k] \leftarrow A_1[i]$

8          $i \leftarrow i + 1$

9          $x \leftarrow x - 1$

10     **else**  $A[k] \leftarrow A_2[j]$

11          $j \leftarrow j + 1$

12          $\text{inversions} \leftarrow \text{inversions} + x$

13 **return** ( $A, \text{inversions}$ )

*inversions*=18

*x*=2

3	7	11	5	9	1	6	13
---	---	----	---	---	---	---	----

4	2	15	8	10	14	12	16
---	---	----	---	----	----	----	----

1	3	5	6	7	9	11	13
---	---	---	---	---	---	----	----

2	4	8	10	12	14	15	16
---	---	---	----	----	----	----	----

$\uparrow$   
 $a_i$

$\uparrow$   
 $b_j$

1	2	3	4	5	6	7	8	9	10						
---	---	---	---	---	---	---	---	---	----	--	--	--	--	--	--

# Counting inversions: how to combine two subproblems?

*Merge – And – Count*( $A_1, A_2$ )

1  $m \leftarrow \text{length}(A_1) + \text{length}(A_2)$

2  $x \leftarrow \text{length}(A_1)$

3  $\text{inversions} \leftarrow 0$

4  $i \leftarrow 1; j \leftarrow 1$

5 **for**  $k = 1$  **to**  $m$

6     **if**  $A_1[i] \leq A_2[j]$

7          $A[k] \leftarrow A_1[i]$

8          $i \leftarrow i + 1$

9          $x \leftarrow x - 1$

10     **else**  $A[k] \leftarrow A_2[j]$

11          $j \leftarrow j + 1$

12          $\text{inversions} \leftarrow \text{inversions} + x$

13 **return** ( $A, \text{inversions}$ )

$\text{inversions} = 18$

$x = 1$

3	7	11	5	9	1	6	13
---	---	----	---	---	---	---	----

4	2	15	8	10	14	12	16
---	---	----	---	----	----	----	----

1	3	5	6	7	9	11	13
---	---	---	---	---	---	----	----

2	4	8	10	12	14	15	16
---	---	---	----	----	----	----	----

$\uparrow$   
 $a_i$

$\uparrow$   
 $b_j$

1	2	3	4	5	6	7	8	9	10	11					
---	---	---	---	---	---	---	---	---	----	----	--	--	--	--	--

# Counting inversions: how to combine two subproblems?

*Merge – And – Count*( $A_1, A_2$ )

1  $m \leftarrow \text{length}(A_1) + \text{length}(A_2)$

2  $x \leftarrow \text{length}(A_1)$

3  $\text{inversions} \leftarrow 0$

4  $i \leftarrow 1; j \leftarrow 1$

5 **for**  $k = 1$  **to**  $m$

6     **if**  $A_1[i] \leq A_2[j]$

7          $A[k] \leftarrow A_1[i]$

8          $i \leftarrow i + 1$

9          $x \leftarrow x - 1$

10     **else**  $A[k] \leftarrow A_2[j]$

11          $j \leftarrow j + 1$

12          $\text{inversions} \leftarrow \text{inversions} + x$

13 **return** ( $A, \text{inversions}$ )

$\text{inversions} = 19$

$x = 1$

3	7	11	5	9	1	6	13
---	---	----	---	---	---	---	----

4	2	15	8	10	14	12	16
---	---	----	---	----	----	----	----

1	3	5	6	7	9	11	13
---	---	---	---	---	---	----	----

2	4	8	10	12	14	15	16
---	---	---	----	----	----	----	----

$\uparrow$   
 $a_i$

$\uparrow$   
 $b_j$

1	2	3	4	5	6	7	8	9	10	11	12				
---	---	---	---	---	---	---	---	---	----	----	----	--	--	--	--

# Counting inversions: how to combine two subproblems?

*Merge – And – Count*( $A_1, A_2$ )

1  $m \leftarrow \text{length}(A_1) + \text{length}(A_2)$

2  $x \leftarrow \text{length}(A_1)$

3  $\text{inversions} \leftarrow 0$

4  $i \leftarrow 1; j \leftarrow 1$

5 **for**  $k = 1$  **to**  $m$

6     **if**  $A_1[i] \leq A_2[j]$

7          $A[k] \leftarrow A_1[i]$

8          $i \leftarrow i + 1$

9          $x \leftarrow x - 1$

10     **else**  $A[k] \leftarrow A_2[j]$

11          $j \leftarrow j + 1$

12          $\text{inversions} \leftarrow \text{inversions} + x$

13 **return** ( $A, \text{inversions}$ )

*inversions*=19

*x*=0

3	7	11	5	9	1	6	13
---	---	----	---	---	---	---	----

4	2	15	8	10	14	12	16
---	---	----	---	----	----	----	----

1	3	5	6	7	9	11	13
---	---	---	---	---	---	----	----

2	4	8	10	12	14	15	16
---	---	---	----	----	----	----	----

$\uparrow$   
 $a_i$

$\uparrow$   
 $b_j$

1	2	3	4	5	6	7	8	9	10	11	12	13			
---	---	---	---	---	---	---	---	---	----	----	----	----	--	--	--

# Counting inversions: how to combine two subproblems?

*Merge – And – Count*( $A_1, A_2$ )

1  $m \leftarrow \text{length}(A_1) + \text{length}(A_2)$

2  $x \leftarrow \text{length}(A_1)$

3  $\text{inversions} \leftarrow 0$

4  $i \leftarrow 1; j \leftarrow 1$

5 **for**  $k = 1$  **to**  $m$

6     **if**  $A_1[i] \leq A_2[j]$

7          $A[k] \leftarrow A_1[i]$

8          $i \leftarrow i + 1$

9          $x \leftarrow x - 1$

10     **else**  $A[k] \leftarrow A_2[j]$

11          $j \leftarrow j + 1$

12          $\text{inversions} \leftarrow \text{inversions} + x$

13 **return** ( $A, \text{inversions}$ )

*inversions*=19

*x*=0

3	7	11	5	9	1	6	13
---	---	----	---	---	---	---	----

4	2	15	8	10	14	12	16
---	---	----	---	----	----	----	----

1	3	5	6	7	9	11	13
---	---	---	---	---	---	----	----

2	4	8	10	12	14	15	16
---	---	---	----	----	----	----	----

$\uparrow$   
 $a_i$

$\uparrow$   
 $b_j$

1	2	3	4	5	6	7	8	9	10	11	12	13	14		
---	---	---	---	---	---	---	---	---	----	----	----	----	----	--	--

# Counting inversions: how to combine two subproblems?

*Merge – And – Count*( $A_1, A_2$ )

1  $m \leftarrow \text{length}(A_1) + \text{length}(A_2)$

2  $x \leftarrow \text{length}(A_1)$

3  $\text{inversions} \leftarrow 0$

4  $i \leftarrow 1; j \leftarrow 1$

5 **for**  $k = 1$  **to**  $m$

6     **if**  $A_1[i] \leq A_2[j]$

7          $A[k] \leftarrow A_1[i]$

8          $i \leftarrow i + 1$

9          $x \leftarrow x - 1$

10     **else**  $A[k] \leftarrow A_2[j]$

11          $j \leftarrow j + 1$

12          $\text{inversions} \leftarrow \text{inversions} + x$

13 **return** ( $A, \text{inversions}$ )

$\text{inversions} = 19$

$x = 0$

3	7	11	5	9	1	6	13
---	---	----	---	---	---	---	----

4	2	15	8	10	14	12	16
---	---	----	---	----	----	----	----

1	3	5	6	7	9	11	13
---	---	---	---	---	---	----	----

2	4	8	10	12	14	15	16
---	---	---	----	----	----	----	----

$\uparrow$   
 $a_i$

$\uparrow$   
 $b_j$

1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	
---	---	---	---	---	---	---	---	---	----	----	----	----	----	----	--



# Counting inversions: how to combine two subproblems?

*Merge – And – Count*( $A_1, A_2$ )

1  $m \leftarrow \text{length}(A_1) + \text{length}(A_2)$

2  $x \leftarrow \text{length}(A_1)$

3  $\text{inversions} \leftarrow 0$

4  $i \leftarrow 1; j \leftarrow 1$

5 **for**  $k = 1$  **to**  $m$

6     **if**  $A_1[i] \leq A_2[j]$

7          $A[k] \leftarrow A_1[i]$

8          $i \leftarrow i + 1$

9          $x \leftarrow x - 1$

10     **else**  $A[k] \leftarrow A_2[j]$

11          $j \leftarrow j + 1$

12          $\text{inversions} \leftarrow \text{inversions} + x$

13 **return** ( $A, \text{inversions}$ )

$\text{inversions} = 19$

$x = 0$

3	7	11	5	9	1	6	13
---	---	----	---	---	---	---	----

4	2	15	8	10	14	12	16
---	---	----	---	----	----	----	----

1	3	5	6	7	9	11	13
---	---	---	---	---	---	----	----

2	4	8	10	12	14	15	16
---	---	---	----	----	----	----	----

1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
---	---	---	---	---	---	---	---	---	----	----	----	----	----	----	----

$\uparrow$   
 $a_i$

$\uparrow$   
 $b_j$

# Counting inversions: how to combine two subproblems?

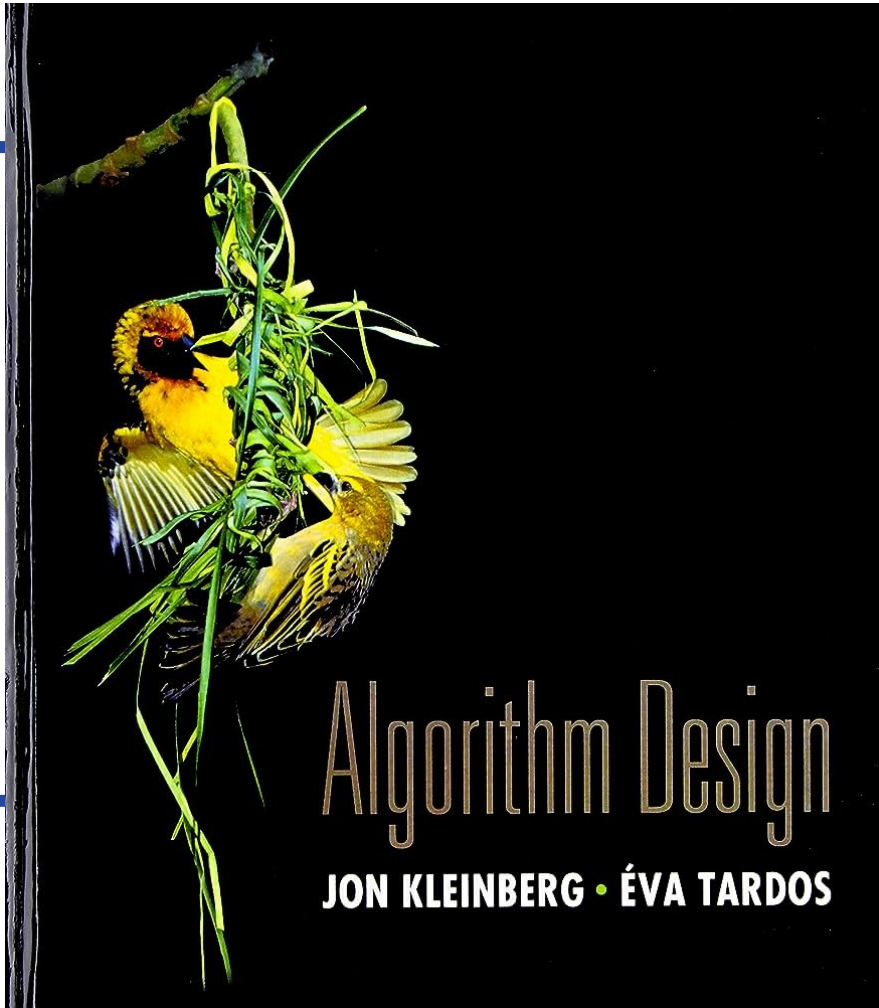
*Sort – And – Count*( $A$ )

```
1    if ( $length(A) > 1$ )
2         $A_1 \leftarrow A[1 \cdots \lfloor n/2 \rfloor] \leftarrow \Theta(1)$ 
3         $A_2 \leftarrow A[\lfloor n/2 \rfloor + 1 \cdots n] \leftarrow \Theta(1)$ 
4         $(A_1, A_{1\_inversions}) \leftarrow \text{Sort – And – Count}(A_1) \leftarrow T(n/2)$ 
5         $(A_2, A_{2\_inversions}) \leftarrow \text{Sort – And – Count}(A_2) \leftarrow T(n/2)$ 
6         $(A, A_{inversions}) \leftarrow \text{Merge – And – Count}(A_1, A_2) \leftarrow \Theta(n)$ 
7    return  $A$ 
```

# Counting inversions: divide-and-conquer algorithm analysis

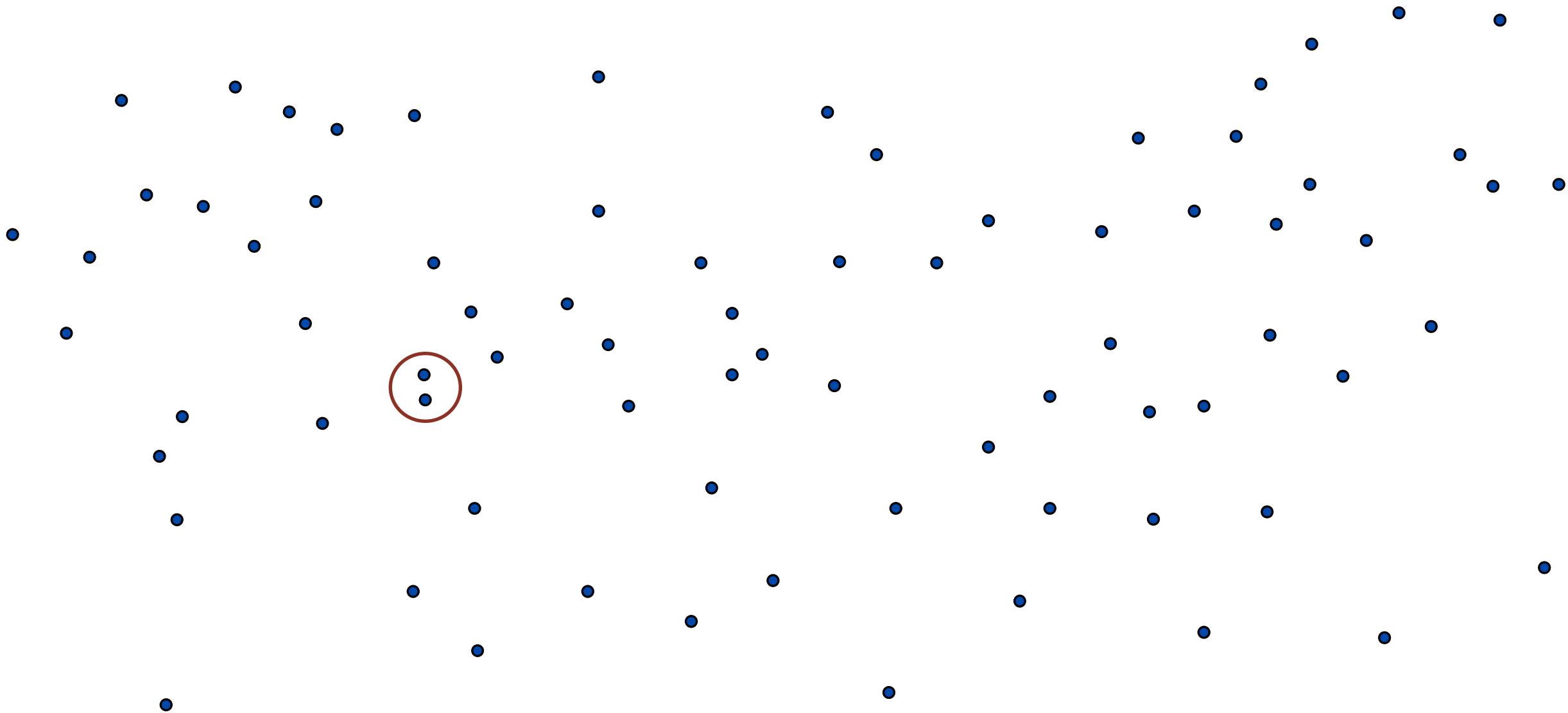
- Proposition. The sort-and-count algorithm counts the number of inversions in a permutation of size  $n$  in  $O(n \log n)$  time.
- **Proof:** The worst-case running time  $T(n)$  satisfies the recurrence:

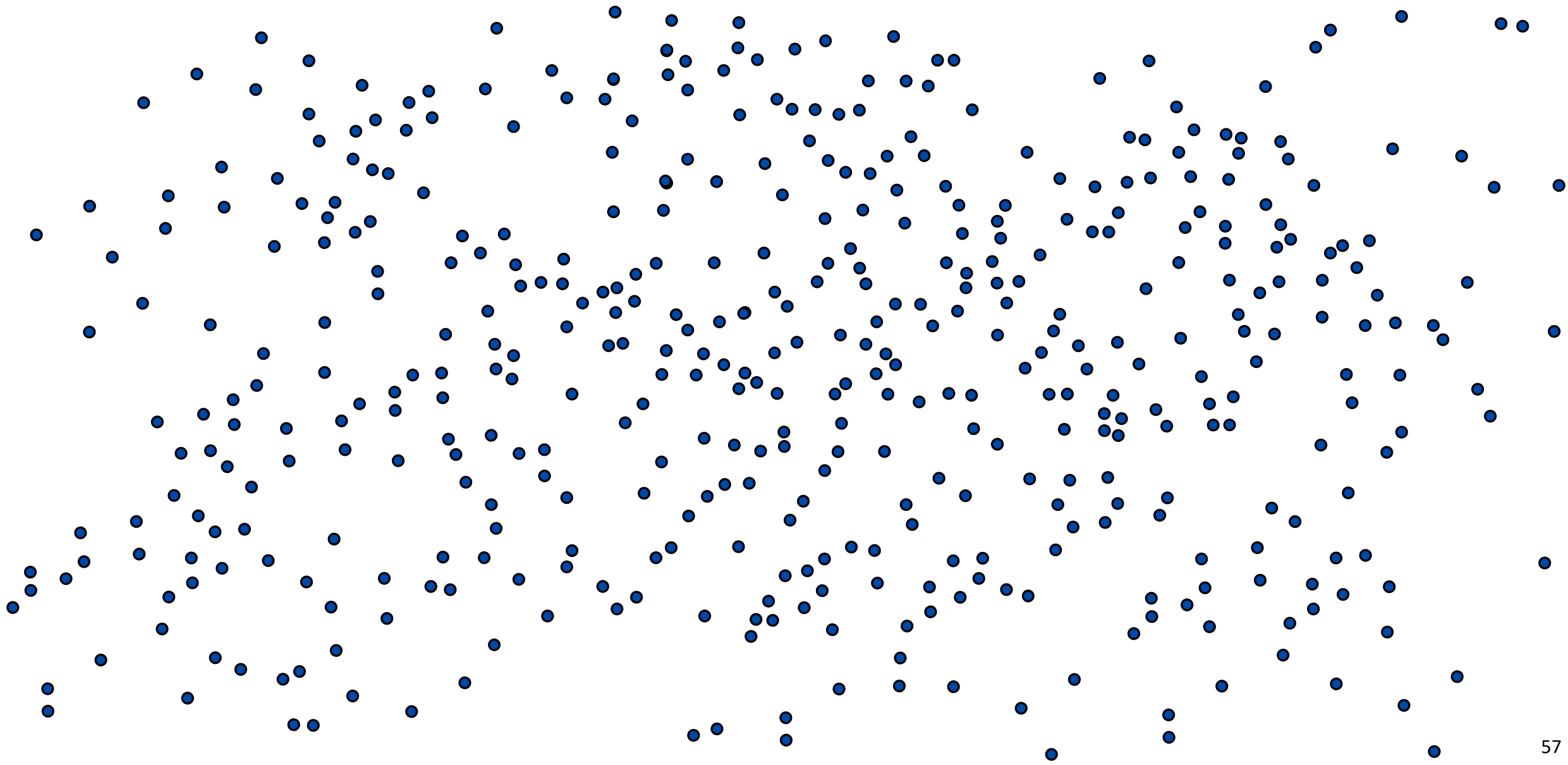
$$T(n) = \begin{cases} \Theta(1) & \text{if } n = 1 \\ 2T\left(\frac{n}{2}\right) + \Theta(n) & \text{if } n > 1 \end{cases}$$



Chapter 5:  
Divide and Conquer

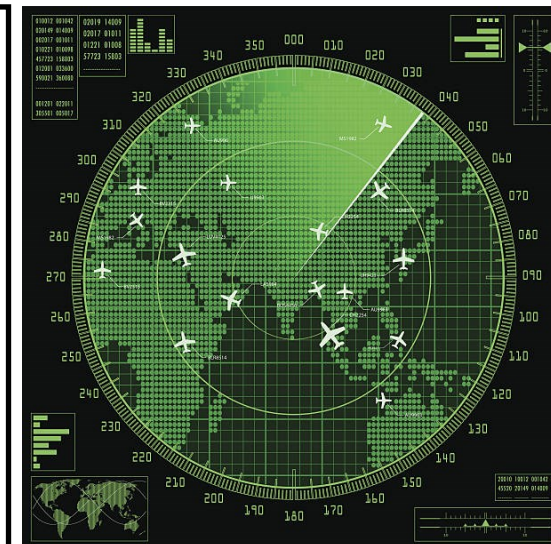
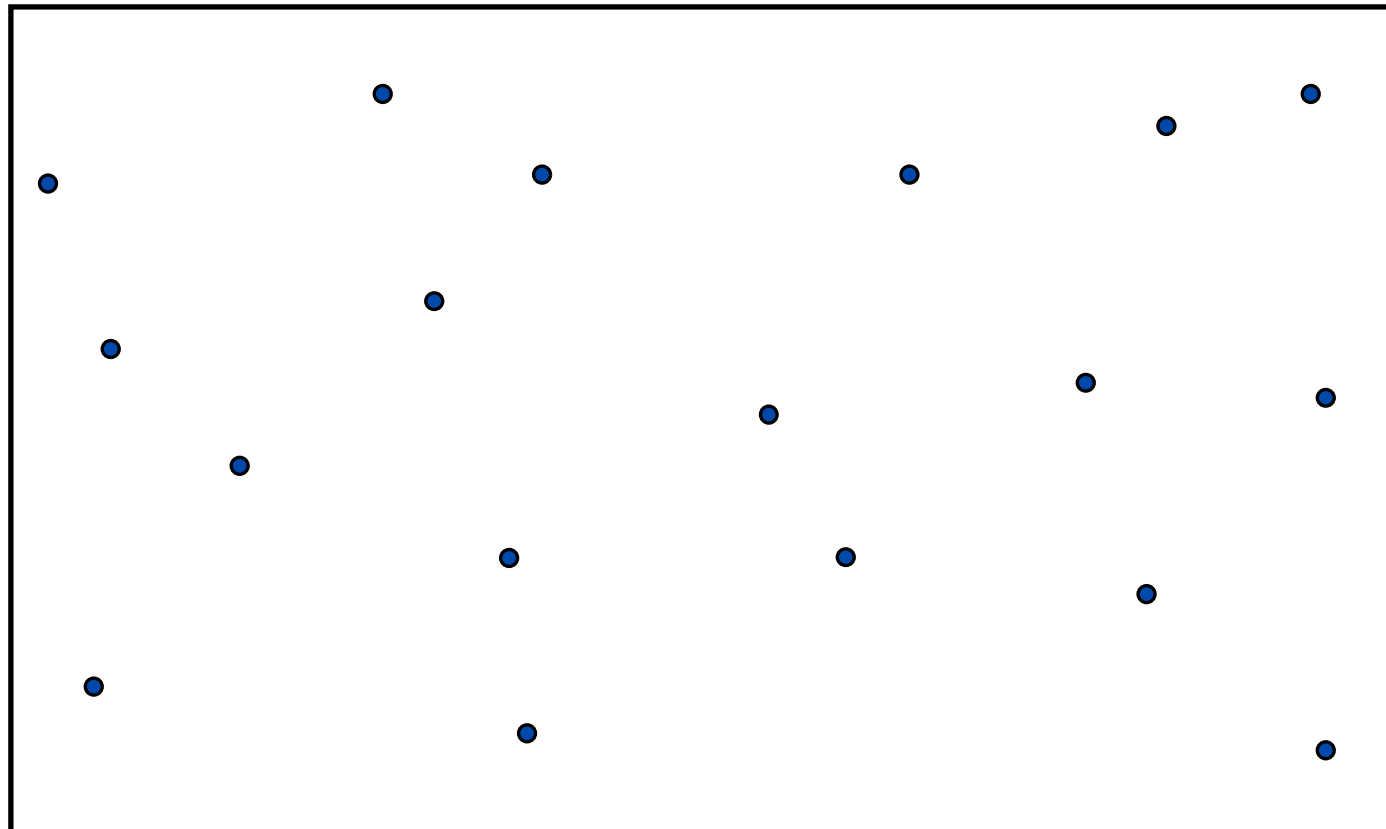
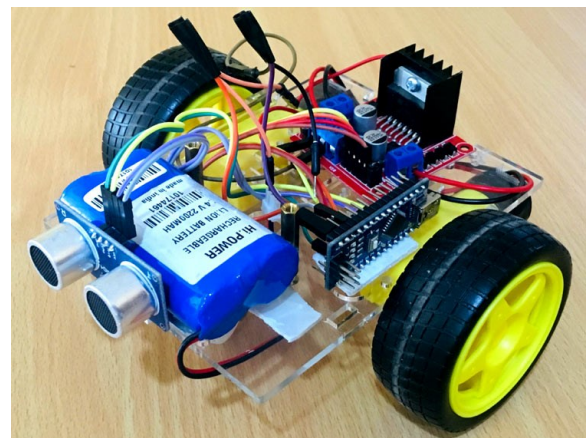
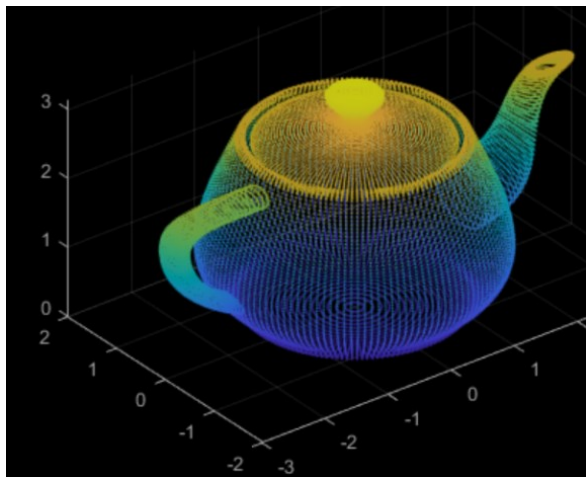
Section 5.4:  
Closed Pair of Points





# Closest pair of points

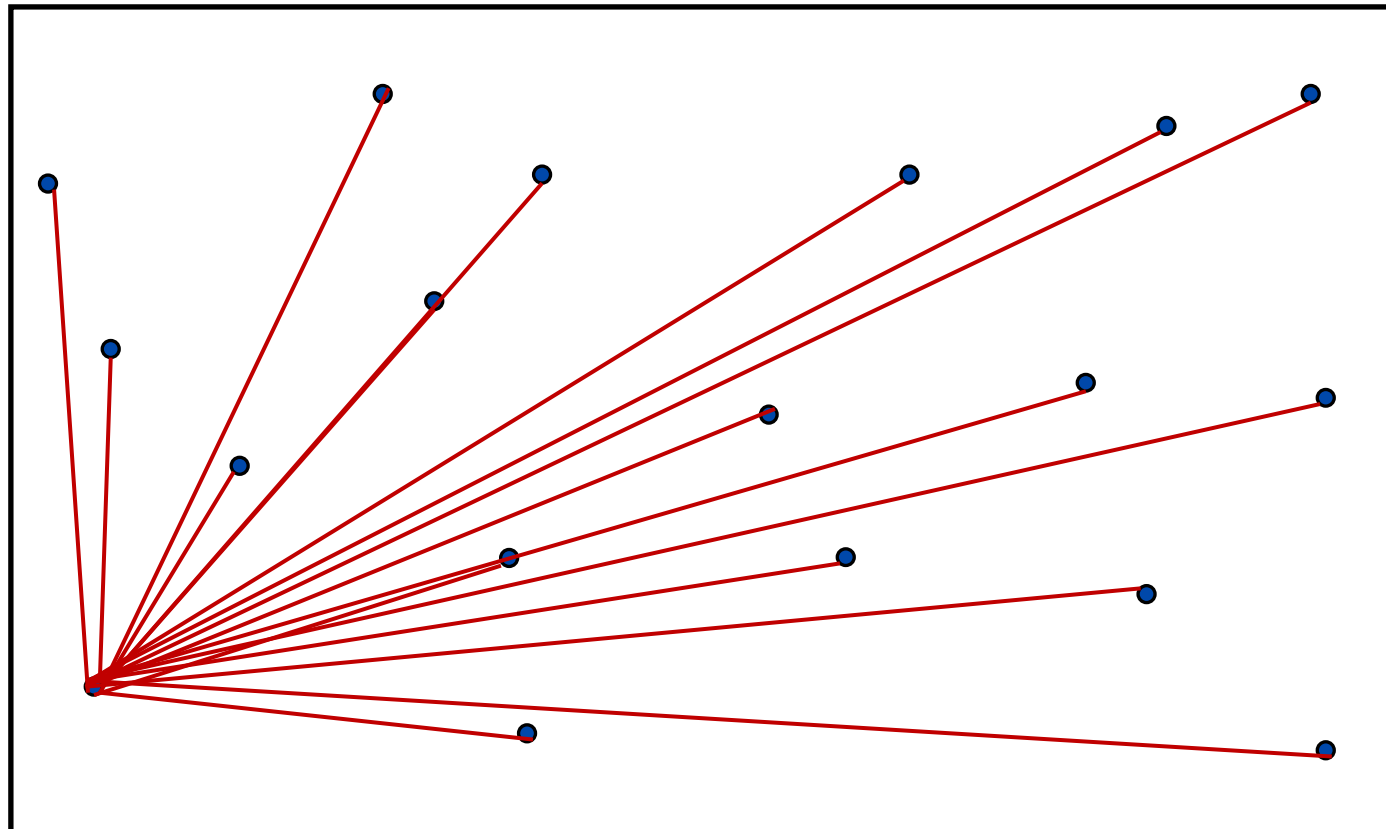
- **Closest pair problem:** Given  $n$  points in the plane, find a pair of points with the smallest Euclidean distance between them.





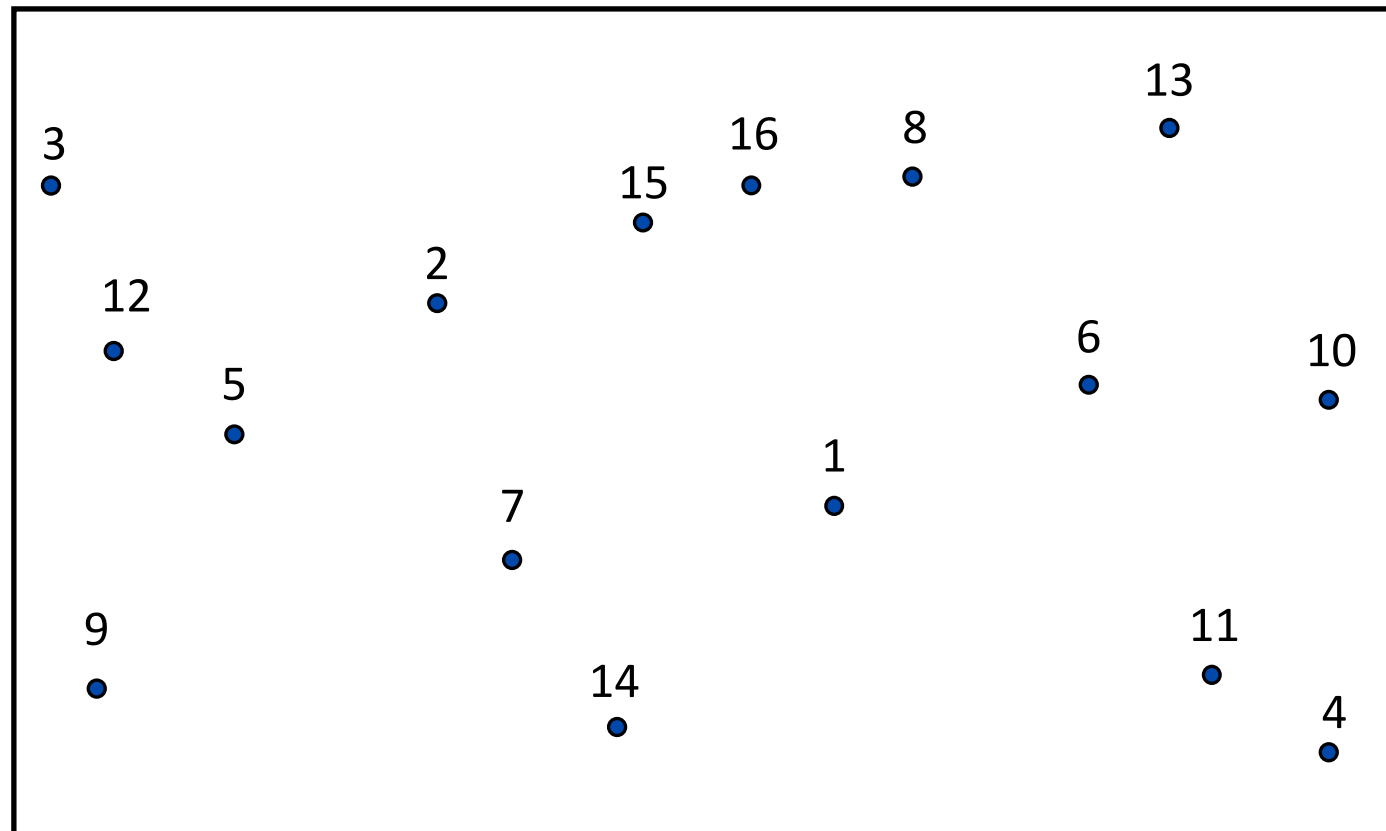
# Closest pair of points

- Closest pair problem. Given  $n$  points in the plane, find a pair of points with the smallest Euclidean distance between them.



# Closest pair of points

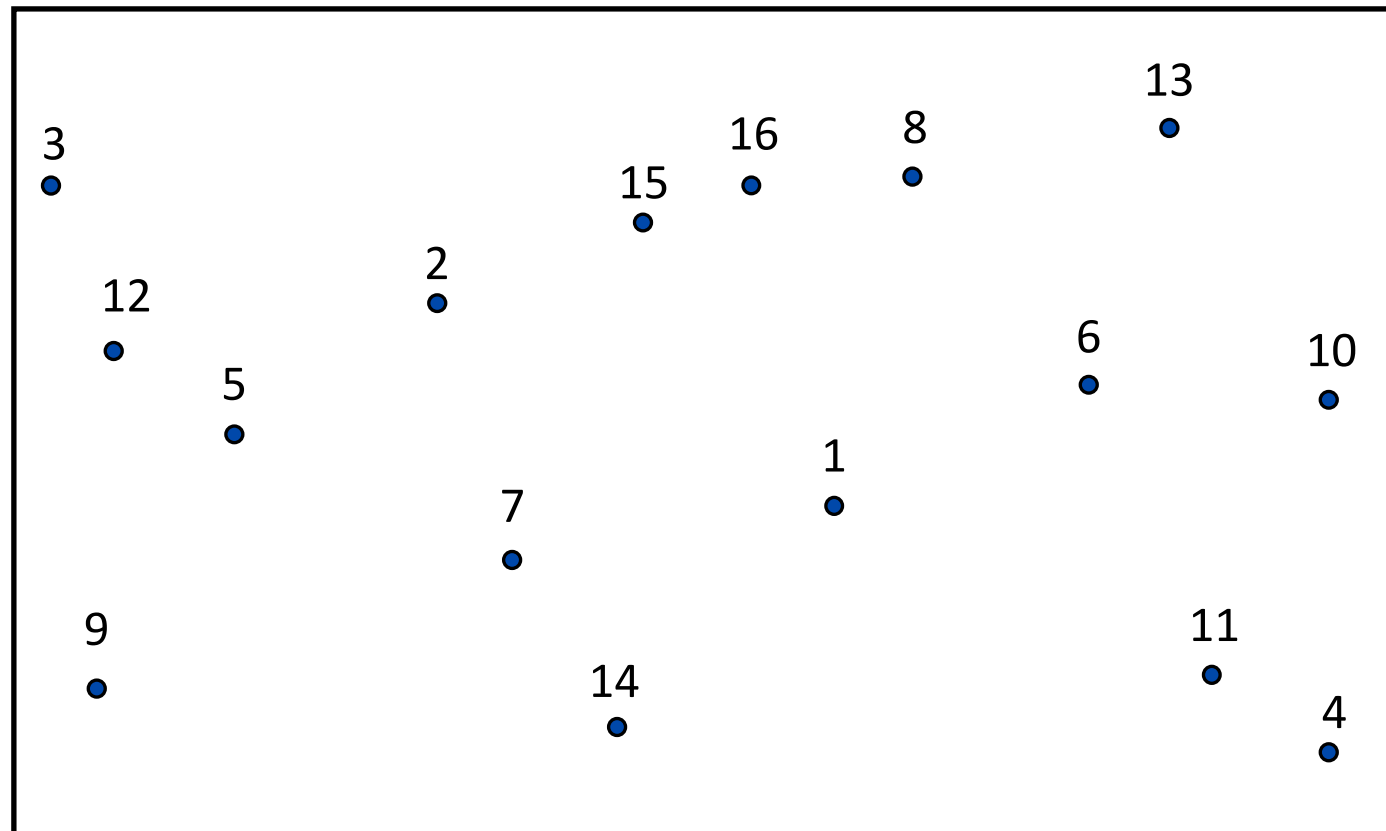
- Closest pair problem. Given  $n$  points in the plane, find a pair of points with the smallest Euclidean distance between them.



(1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17)

# Closest pair of points: Sorting Solution

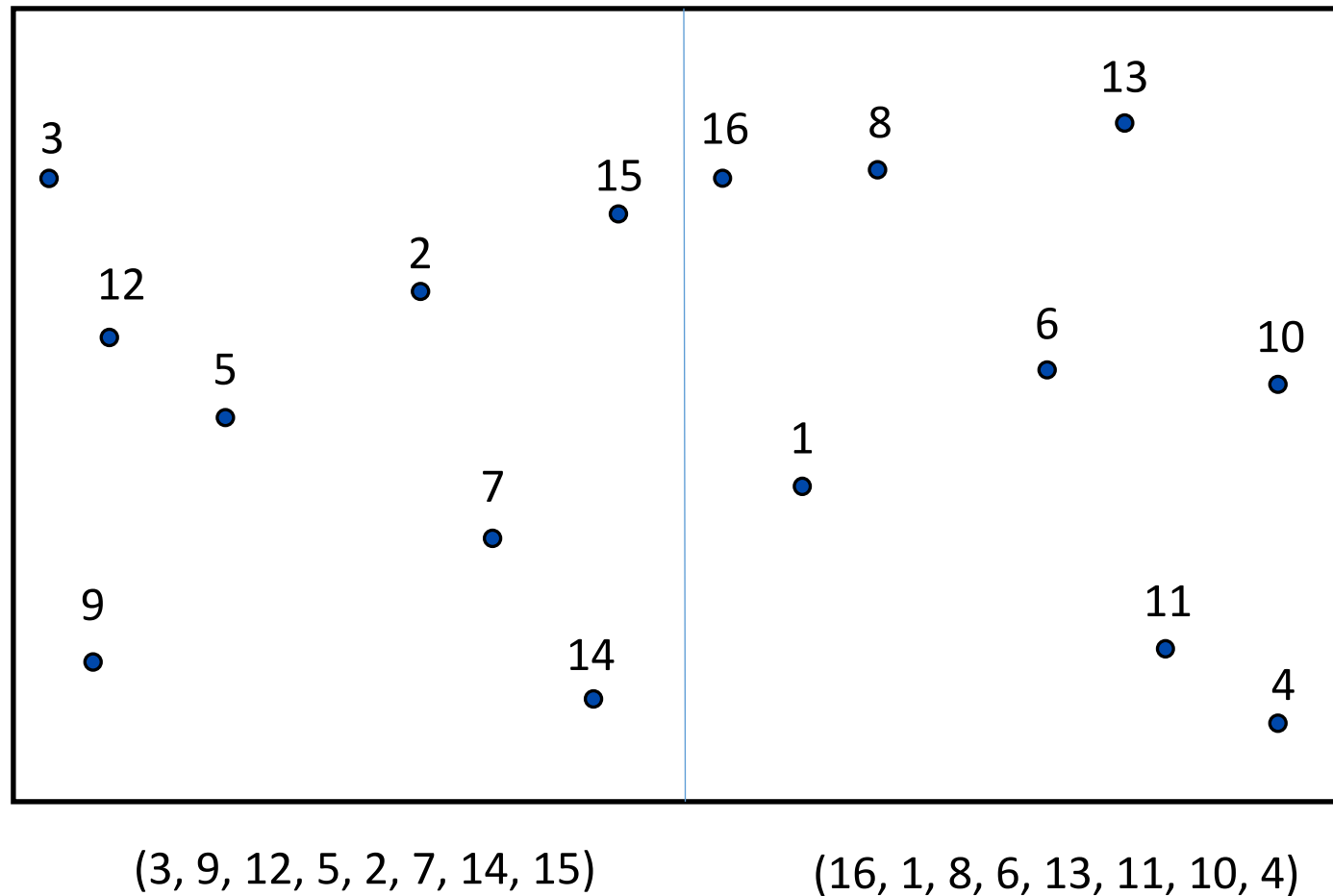
- Sort points by x-coordinate



(3, 9, 12, 5, 2, 7, 14, 15, 16, 1, 8, 6, 13, 11, 10, 4)

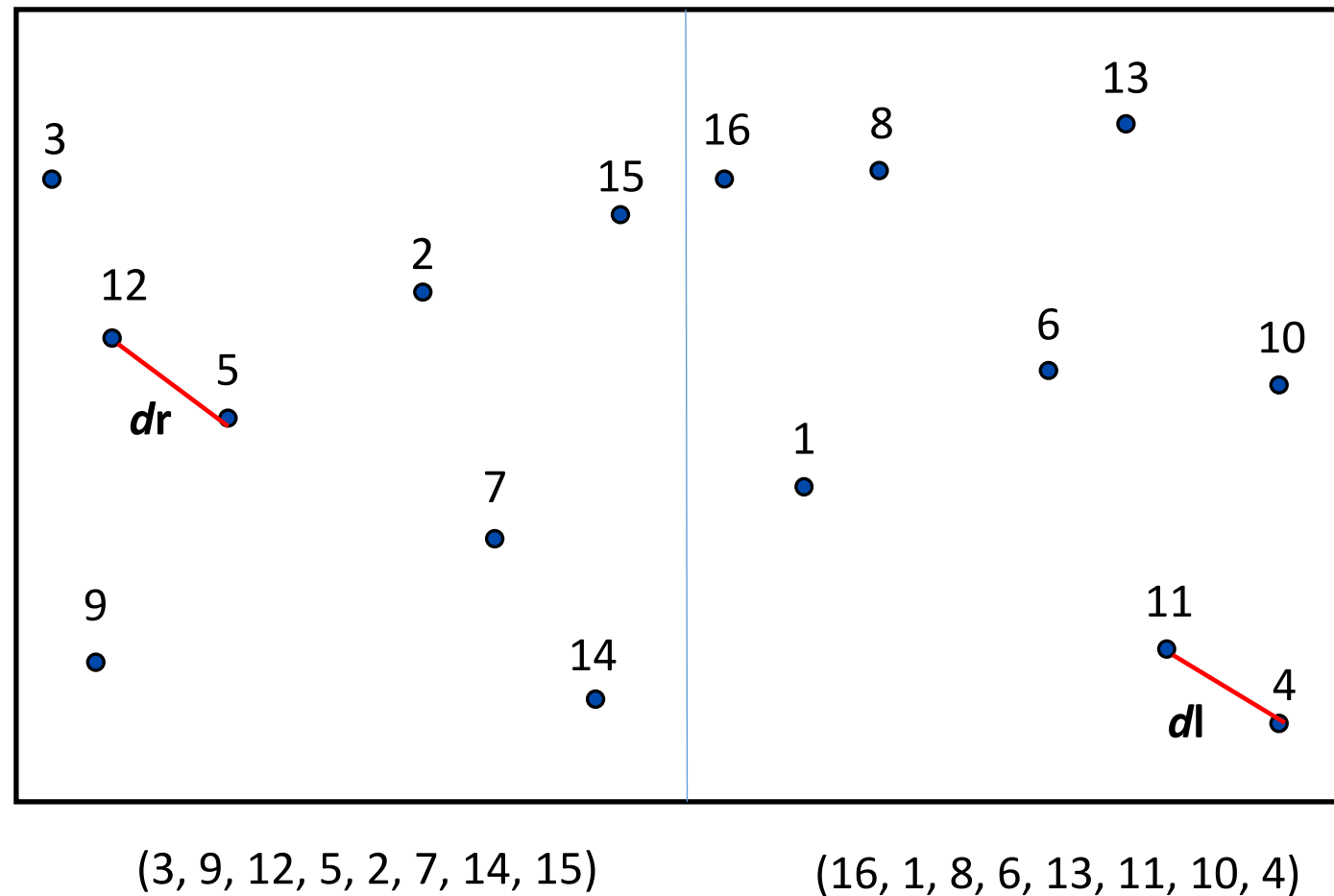
# Closest pair of points

- Divide: draw vertical line  $L$  so that  $n / 2$  points on each side.



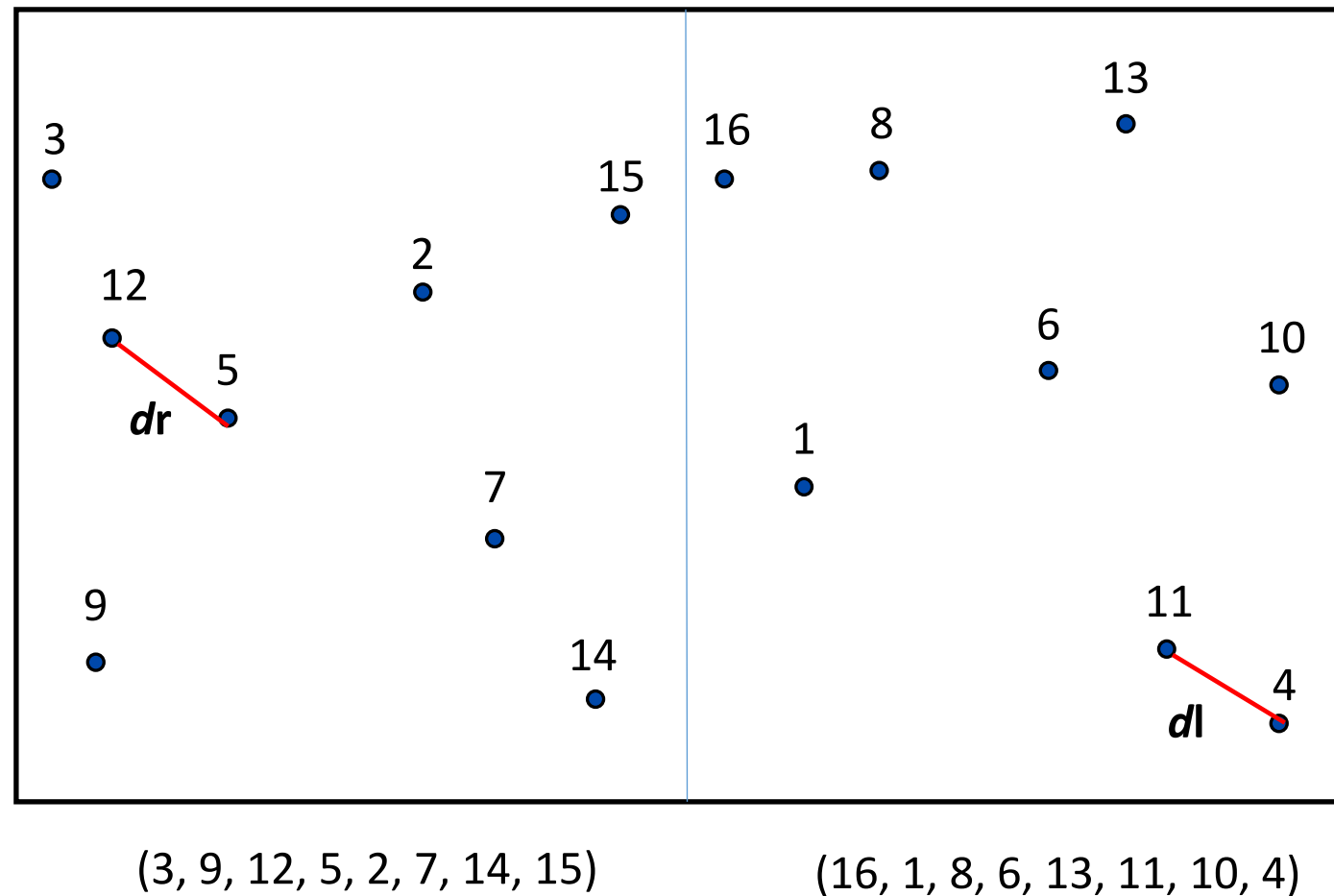
# Closest pair of points

- Conquer: find closest pair in each side recursively.



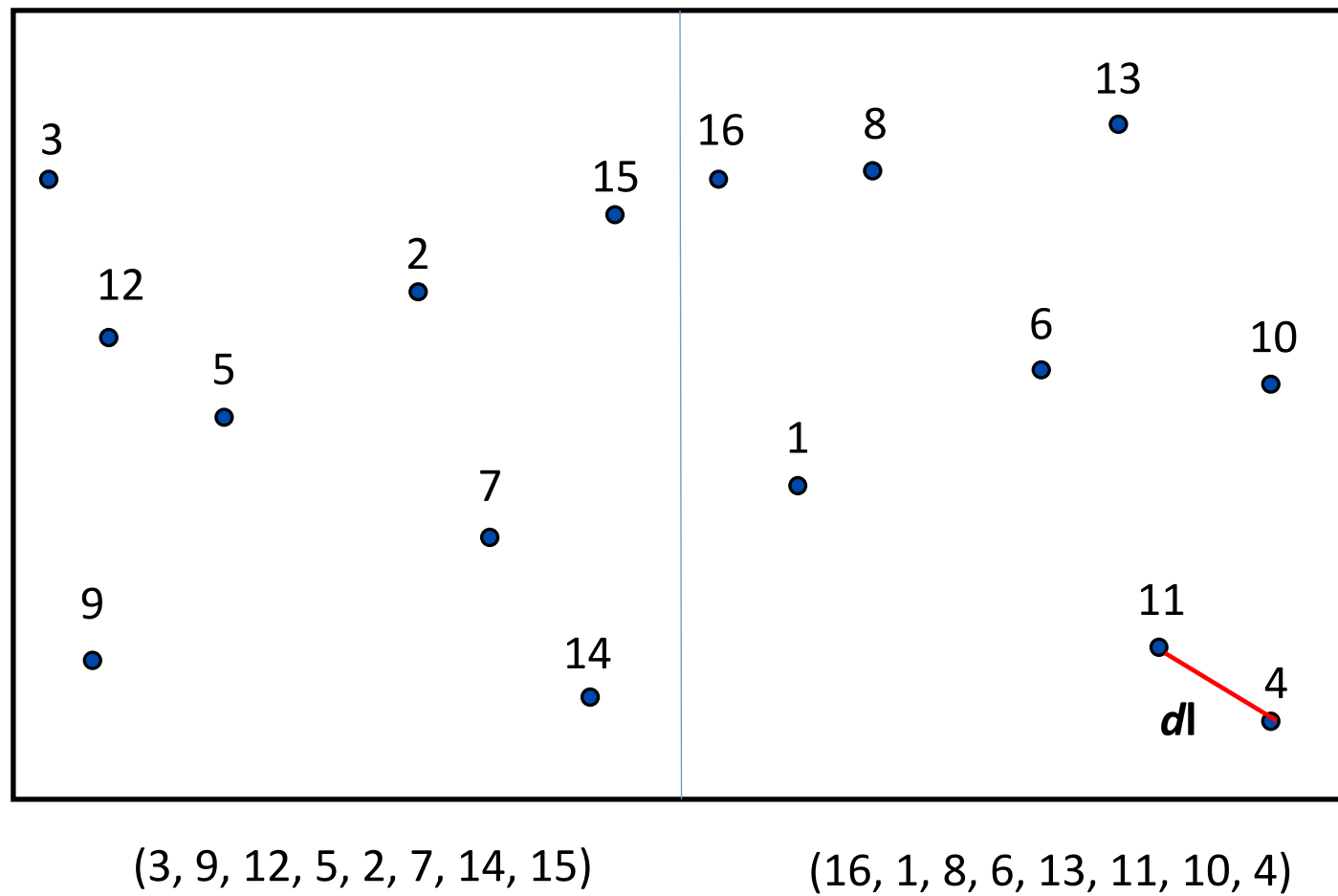
# Closest pair of points

- Conquer: find closest pair in each side recursively.



$$d = \min(dl, dr)$$

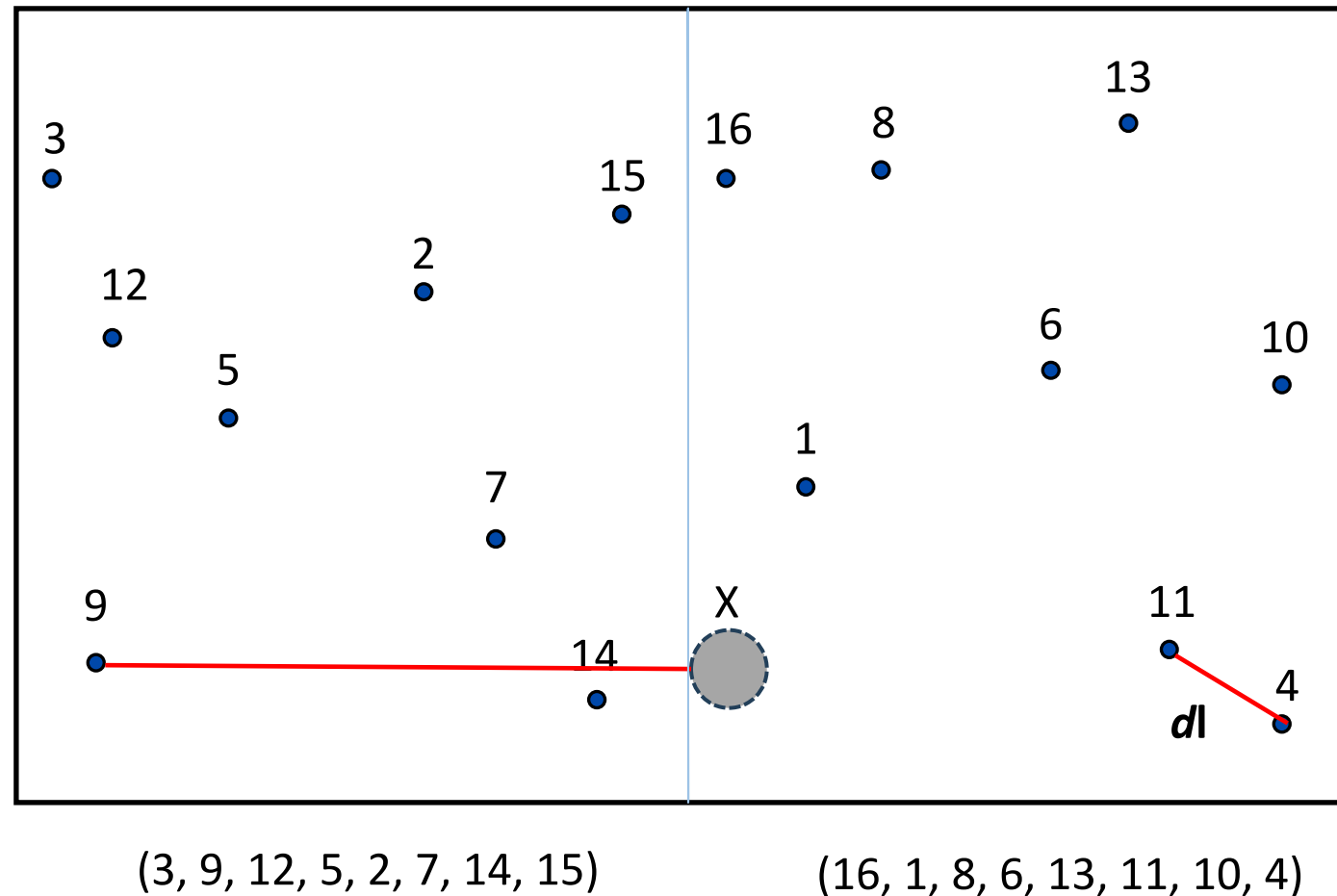
# Closest pair of points





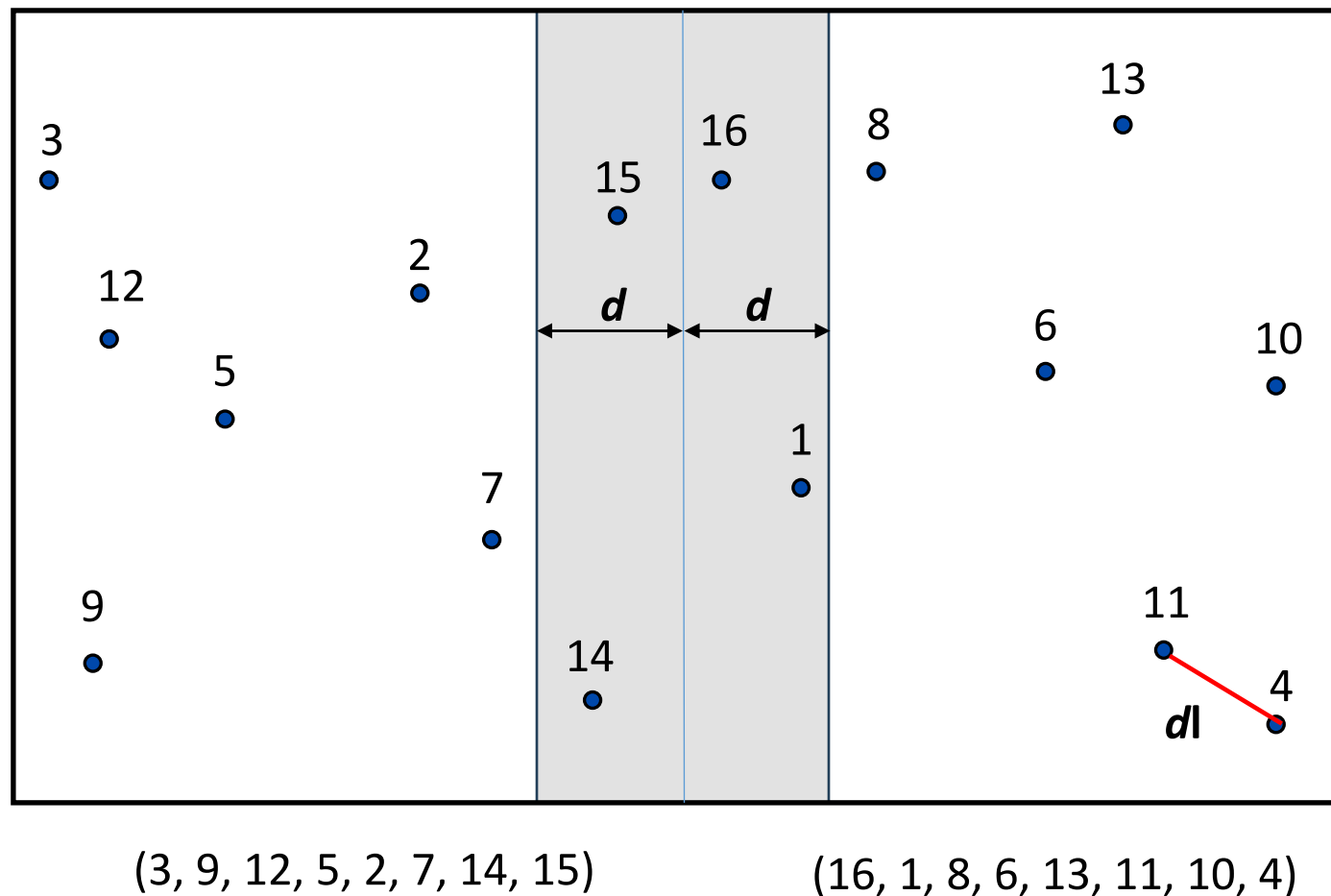
# Closest pair of points

- Closest pair problem. Given  $n$  points in the plane, find a pair of points with the smallest Euclidean distance between them.



# Closest pair of points

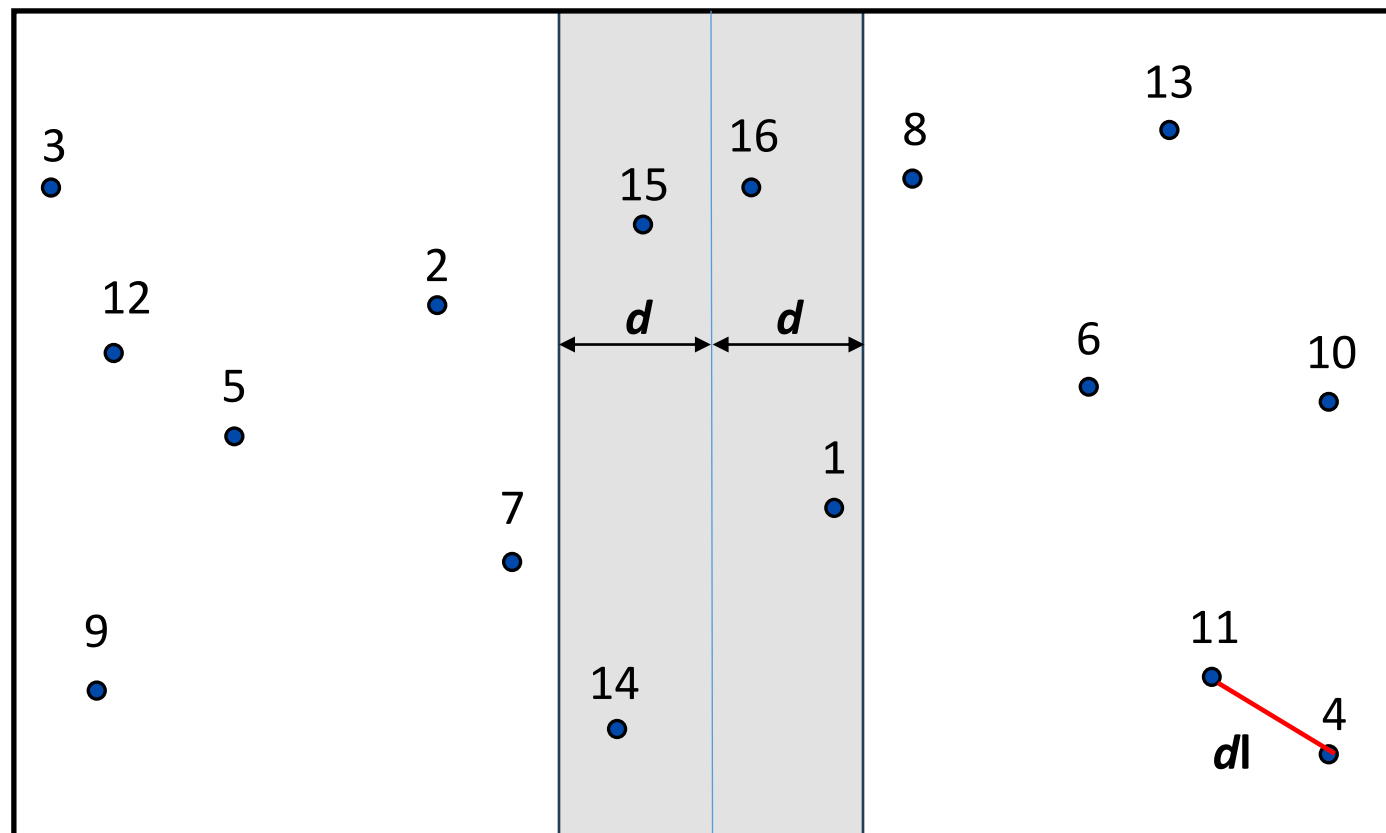
- Observation: suffices to consider only those points within  $d$  of line  $L$



$$d = \min(d_l, d_r)$$

# Closest pair of points

- Observation: suffices to consider only those points within  $d$  of line  $L$

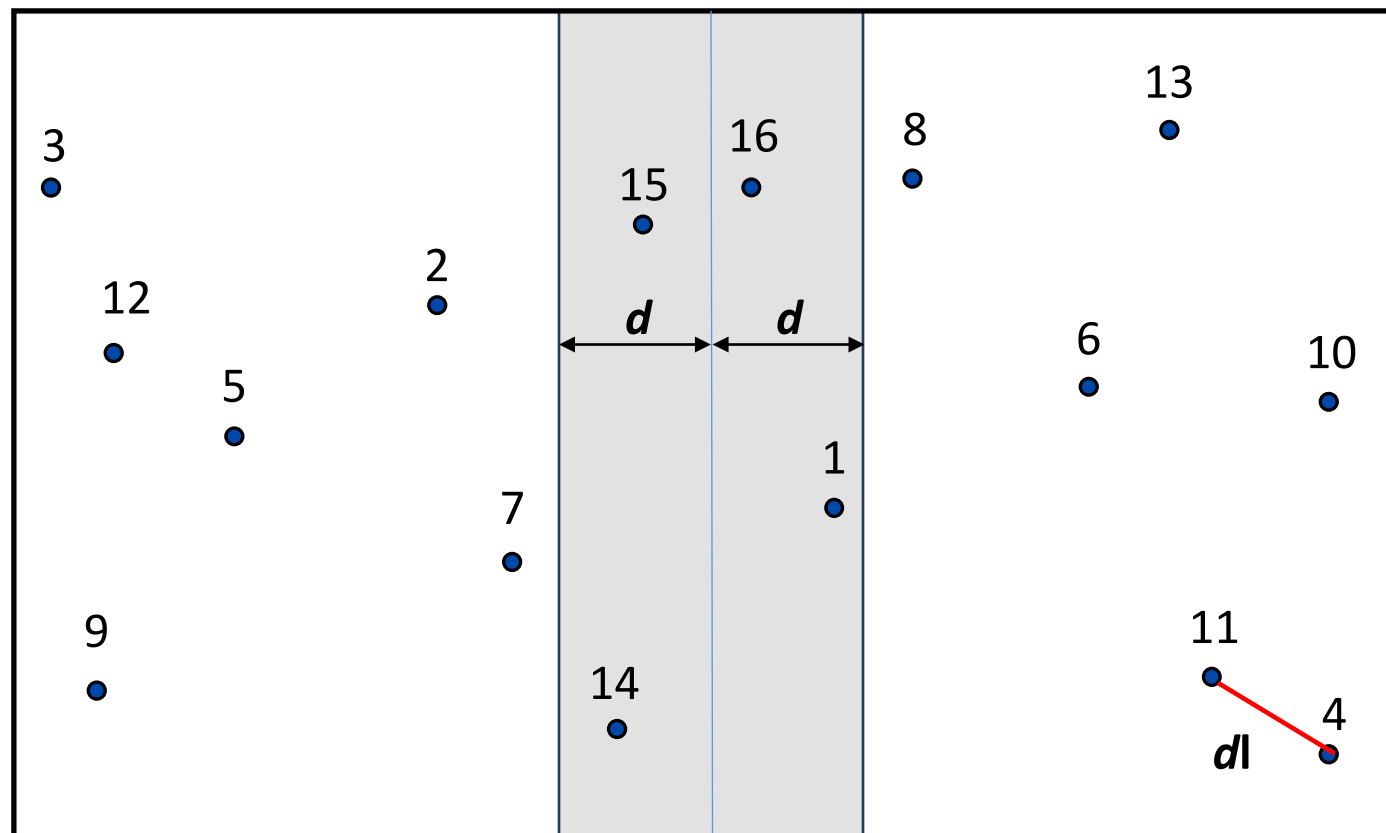


$$d = \min(d_l, d_r)$$

(14,15,16,1)

# Closest pair of points

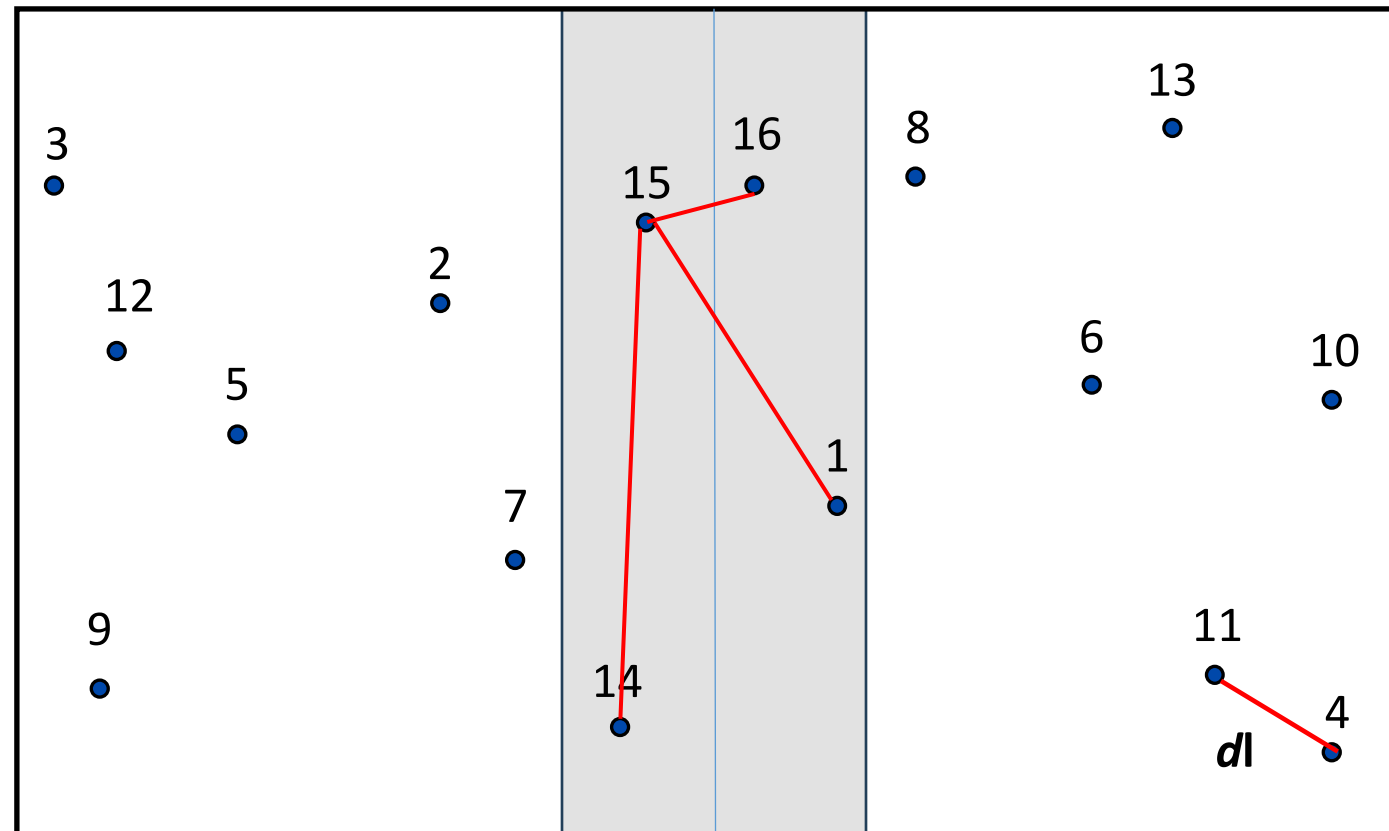
- Reorder based on Y-axis



(16,15,1,14)

# Closest pair of points

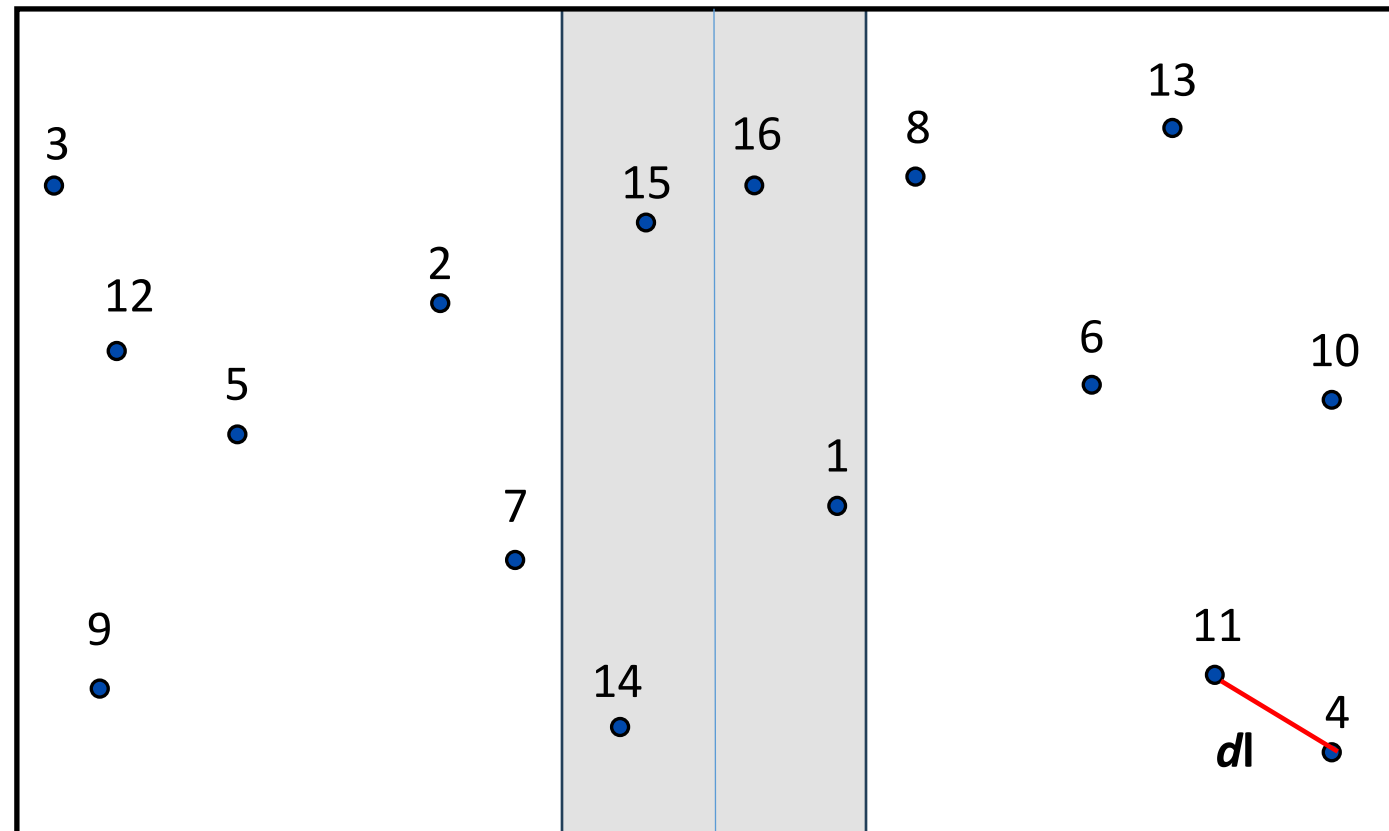
- Reorder based on Y-axis



(16,15,1,14)

# Closest pair of points

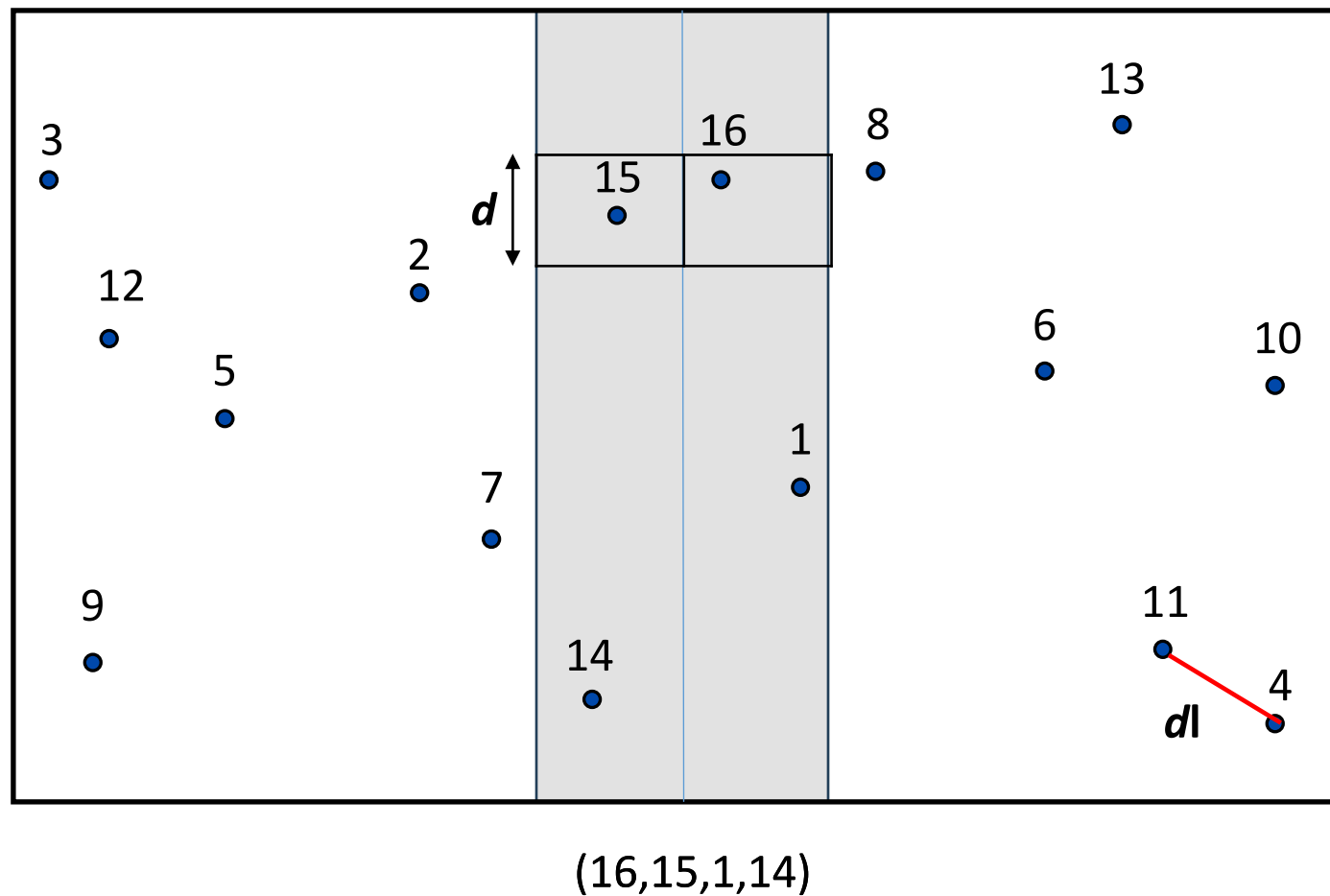
- Reorder based on Y-axis



(16,15,1,14)

# Closest pair of points

- Reorder based on Y-axis

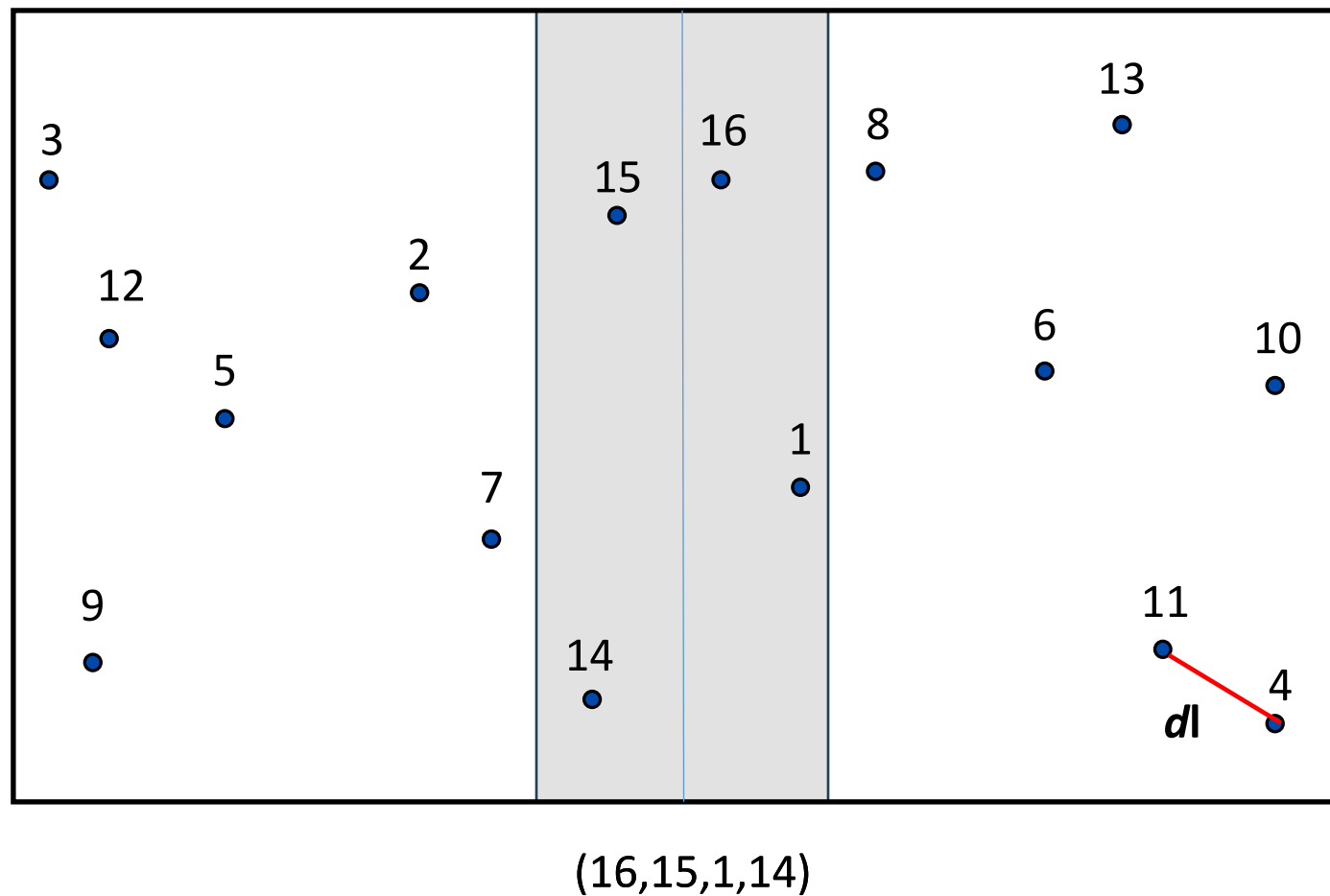


$$d = \min(d_l, d_r)$$

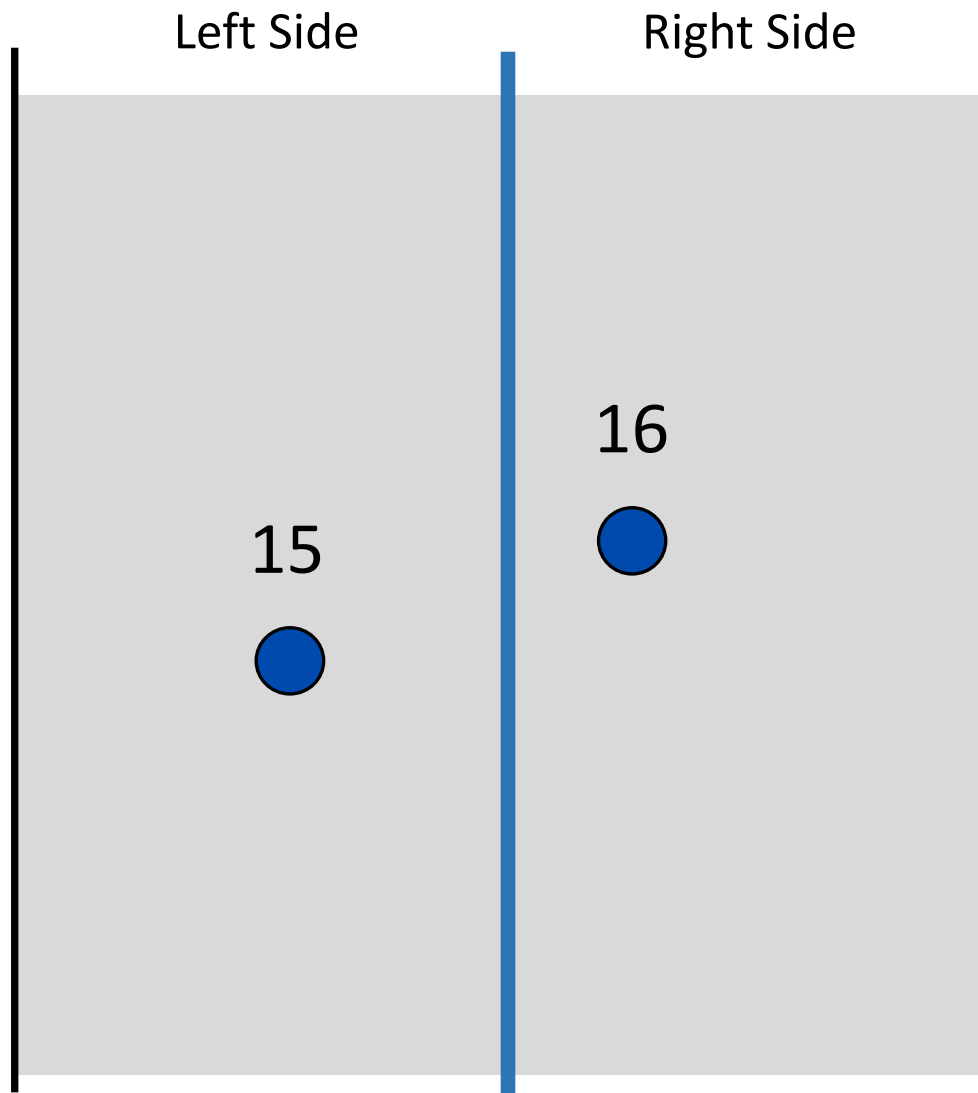


# Closest pair of points

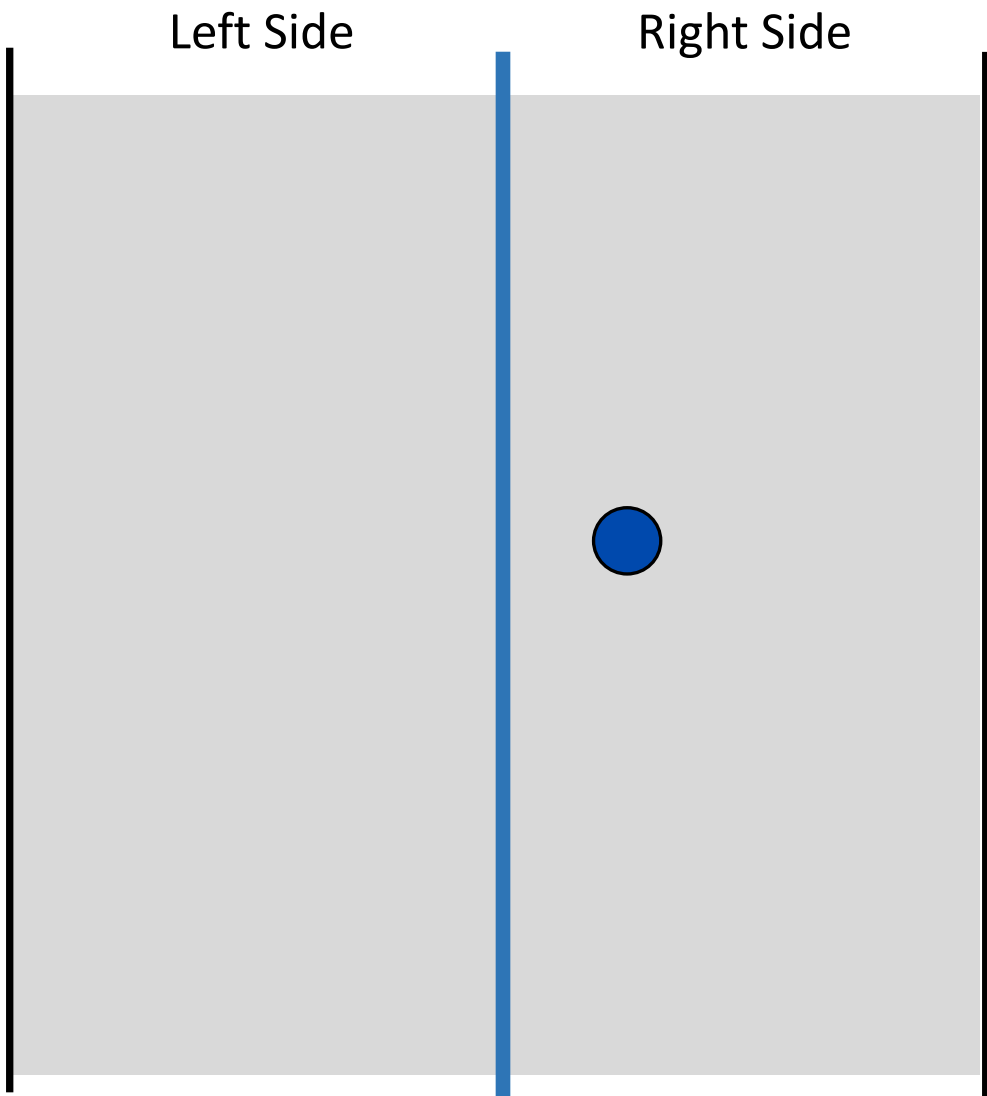
- Reorder based on Y-axis



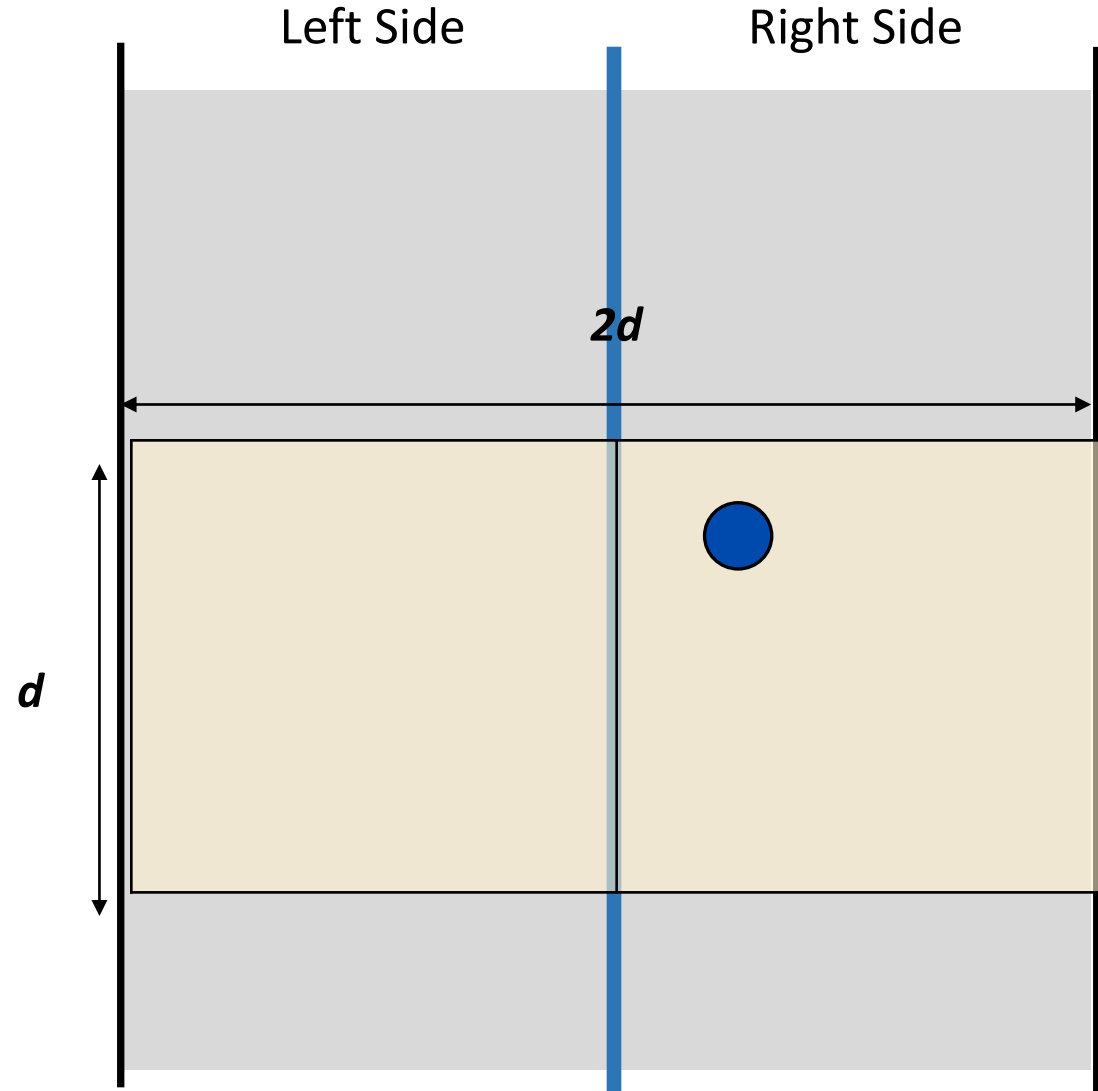
16
15
1
14



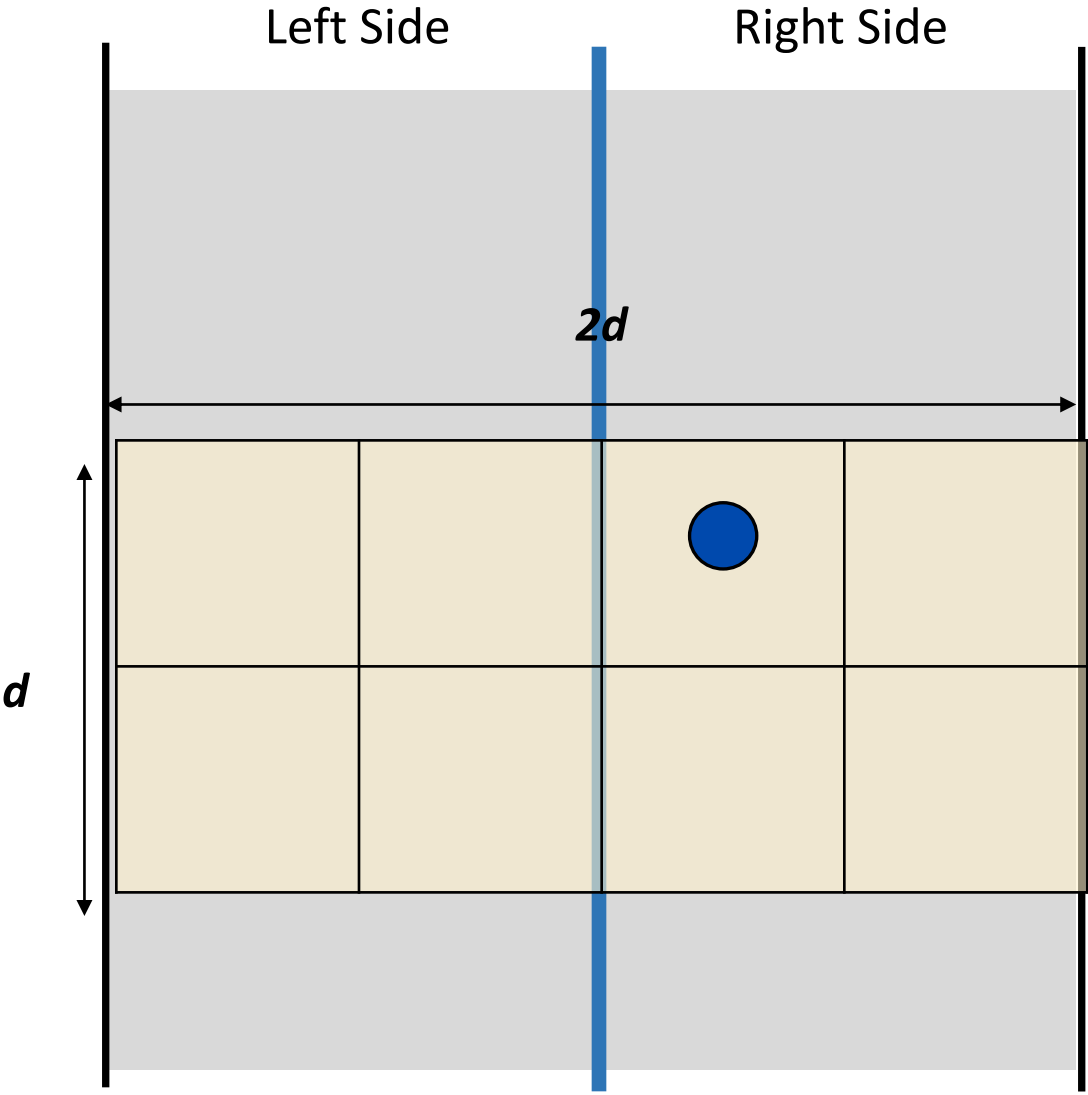
16
15
1
14
$P_4$
$P_5$
$P_6$
$P_7$
$P_8$
...
$P_{\sim n}$



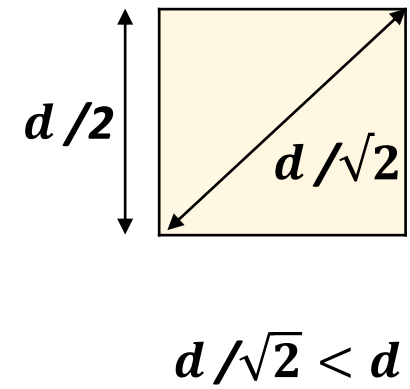
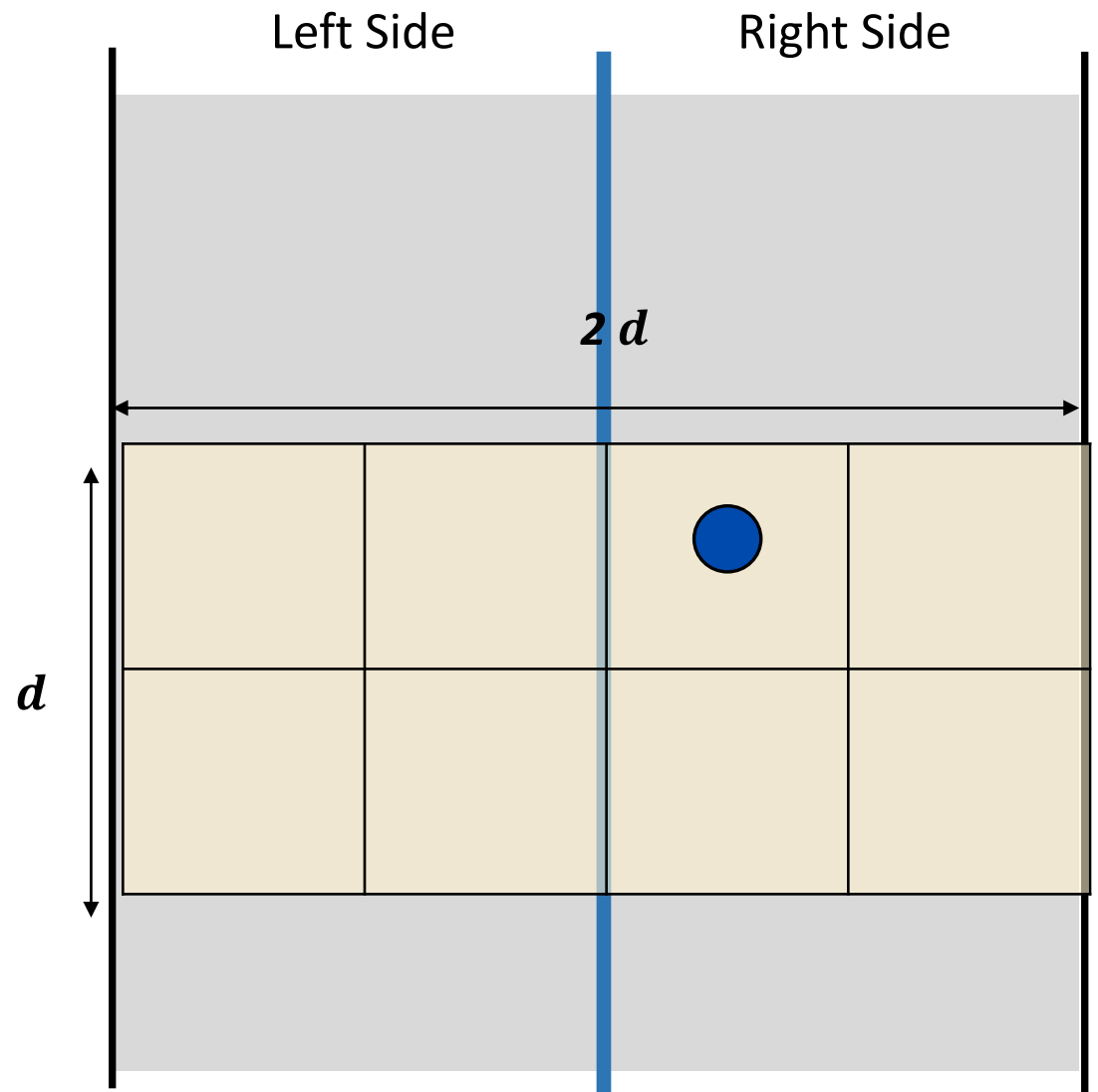
16
15
1
14
$P_4$
$P_5$
$P_6$
$P_7$
$P_8$
...
$P_{\sim n}$



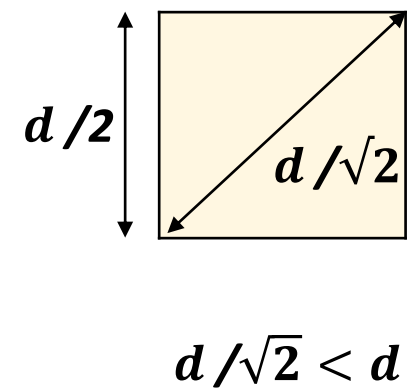
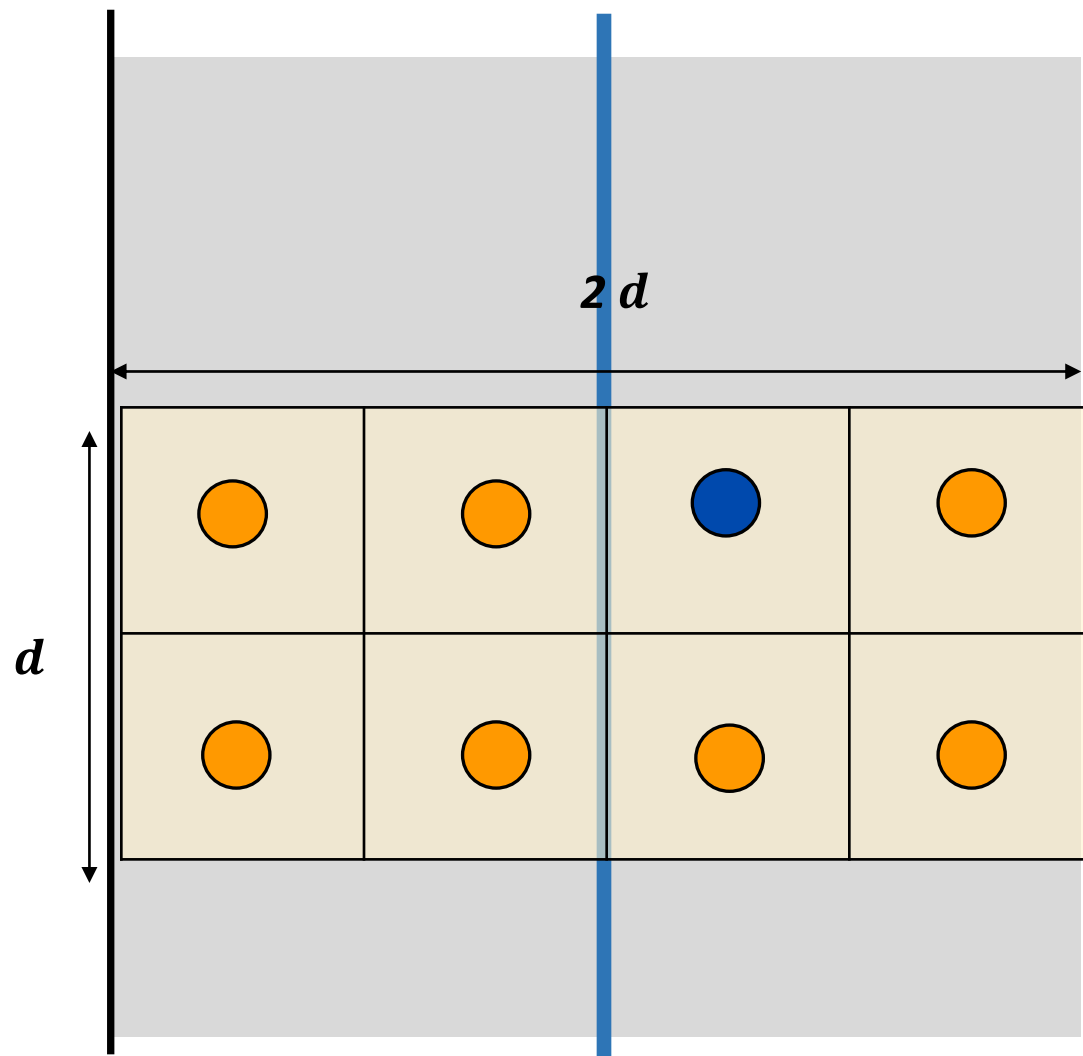
16
15
1
14
$P_4$
$P_5$
$P_6$
$P_7$
$P_8$
...
$P_{\sim n}$



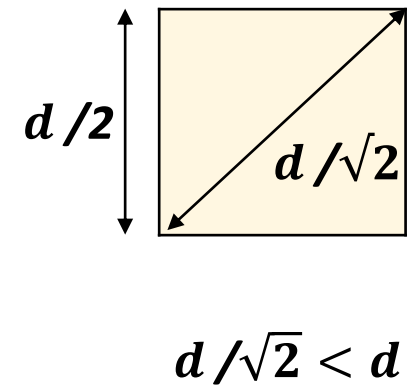
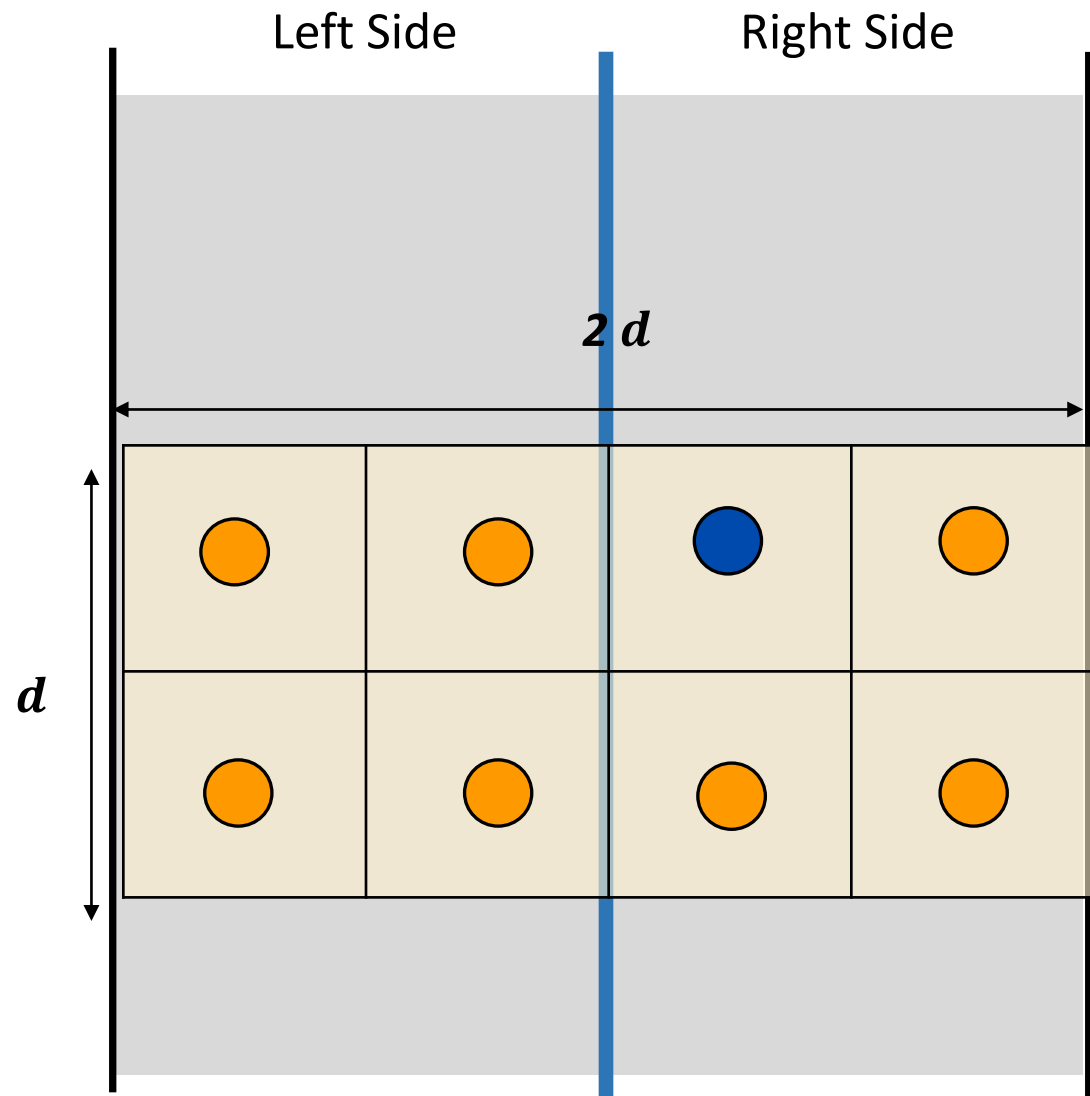
16
15
1
14
$P_4$
$P_5$
$P_6$
$P_7$
$P_8$
...
$P_{\sim n}$



16
15
1
14
$P_4$
$P_5$
$P_6$
$P_7$
$P_8$
...
$P_{\sim n}$

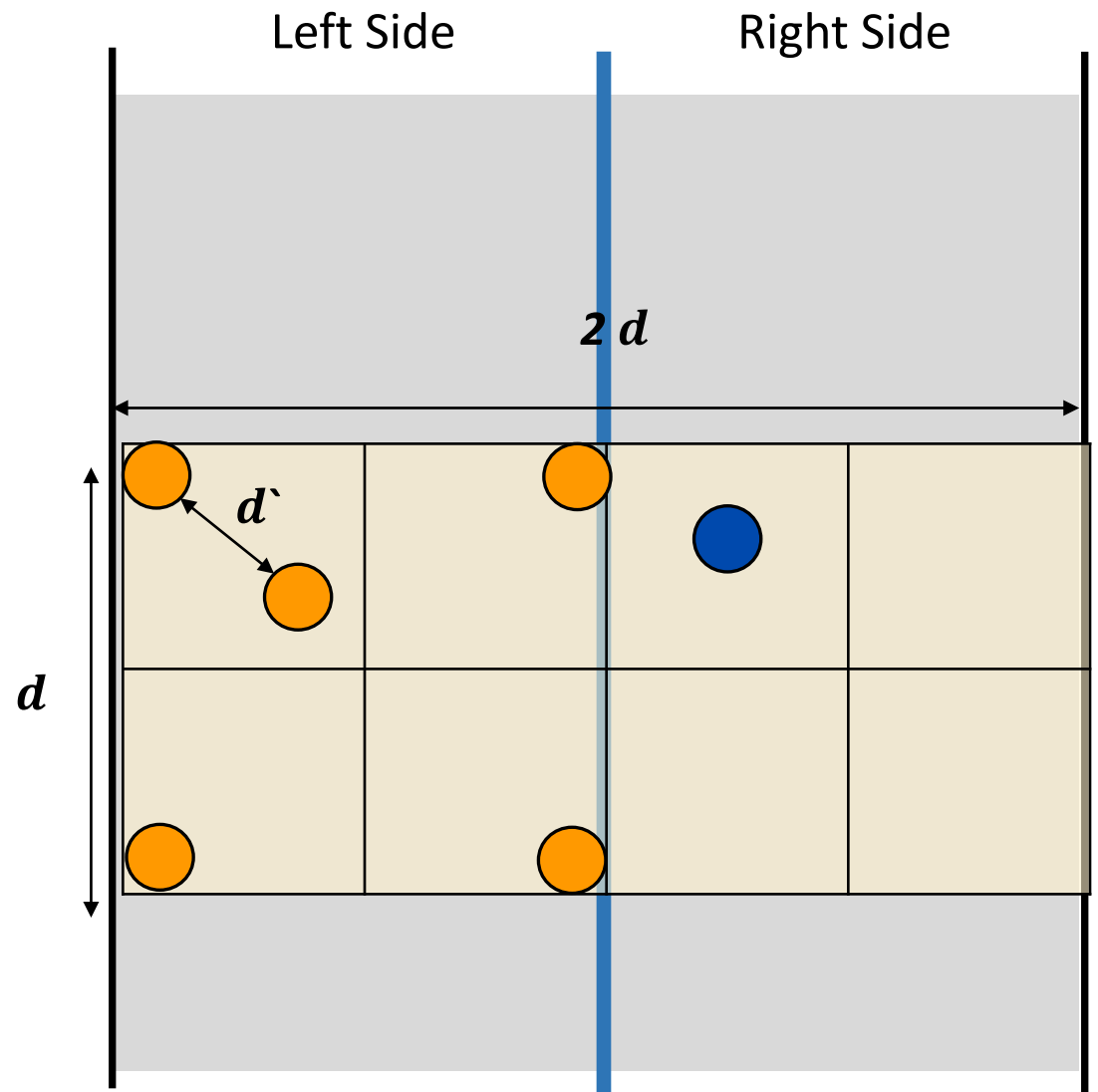


16
15
1
14
$P_4$
$P_5$
$P_6$
$P_7$
$P_8$
...
$P_{\sim n}$



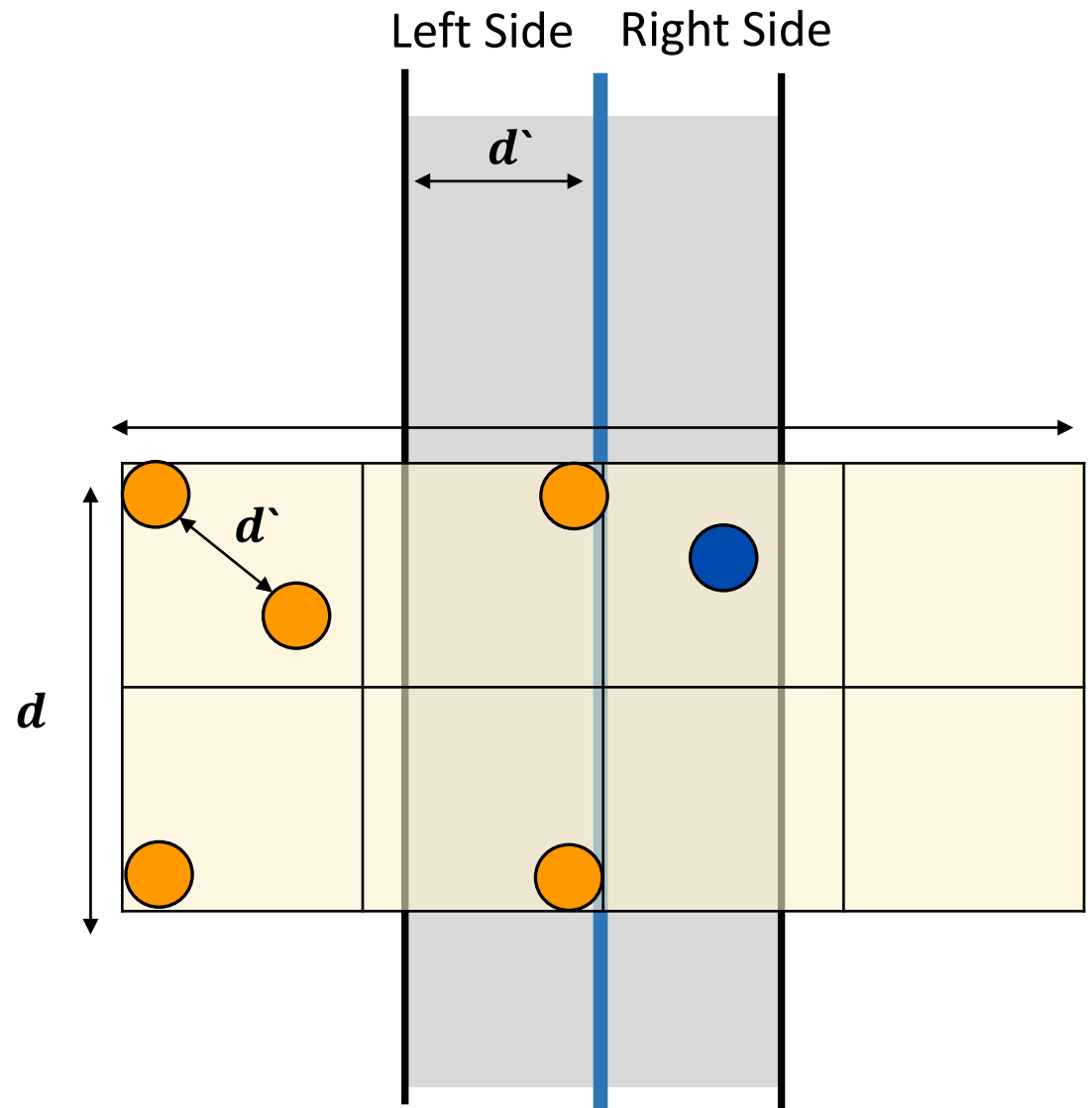


16
15
1
14
$P_4$
$P_5$
$P_6$
$P_7$
$P_8$
...
$P_{\sim n}$



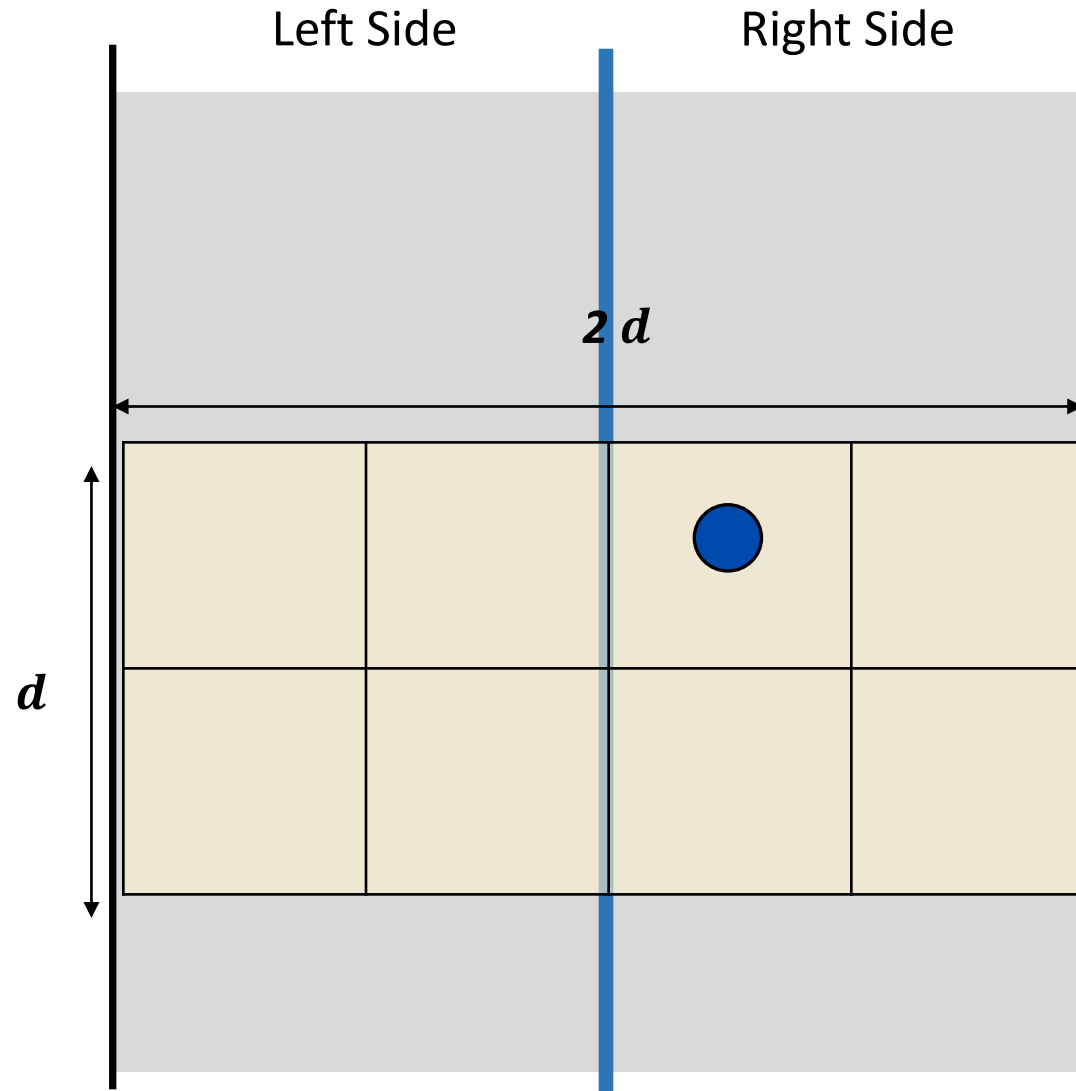
$$d\` < d$$

16
15
1
14
$P_4$
$P_5$
$P_6$
$P_7$
$P_8$
...
$P_{\sim n}$



$$d\` < d$$

16
15
1
14
$P_4$
$P_5$
$P_6$
$P_7$
$P_8$
...
$P_{\sim n}$



# Closest Pair: Live Poll 1

What is the complexity of this algo?

- A.  $n^2$*
- B.  $n^2 \log^2 n$*
- C.  $n \log n$*
- D.  $n \log^2 n$*



Scan the QR code to  
vote or go to  
<https://forms.office.com/r/zKJjPb1tCK>

### Closet Pair: Live Poll 1

Only people in my organization can respond, Record name

#### 1. What is the complexity of this algo?

$n^2$

3%

$n^2 (\log)^2 n$

8%

$n \log n$

53%

$n (\log)^2 n$

36%

36 responses



Scan the QR code to  
vote or go to  
<https://forms.office.com/r/zKJjPb1tCK>

# Finding Closest Point Algorithm

Sort points based on X-coordinates  $\longrightarrow O(n \log n)$

**CLOSEST-PAIR**( $p_1, p_2, \dots, p_n$ )

---

Compute vertical line  $L$  such that half the points are on each side of the line.  $\longrightarrow O(n)$

$\delta_1 \leftarrow$  **CLOSEST-PAIR**(points in left half).  $\longrightarrow T(n/2)$

$\delta_2 \leftarrow$  **CLOSEST-PAIR**(points in right half).  $\longrightarrow T(n/2)$

$\delta \leftarrow \min \{ \delta_1, \delta_2 \}$ .  $\longrightarrow O(1)$

$A \leftarrow$  list of all points closer than  $\delta$  to line  $L$ .  $\longrightarrow O(n)$

Sort points in  $A$  by y-coordinate.  $\longrightarrow O(n \log n)$

Scan points in  $A$  in y-order and compare distance between each point and next 7 neighbors.  $\longrightarrow O(n)$

If any of these distances is less than  $\delta$ , update  $\delta$ .  $\longrightarrow O(1)$

**RETURN**  $\delta$ .

---



# Finding Closest Point Algorithm

$$O(n \log n) + O(\log n) \cdot (O(n) + O(1) + O(n) + O(n \log n) + O(n) + O(1))$$

$$O(n \log^2 n)$$



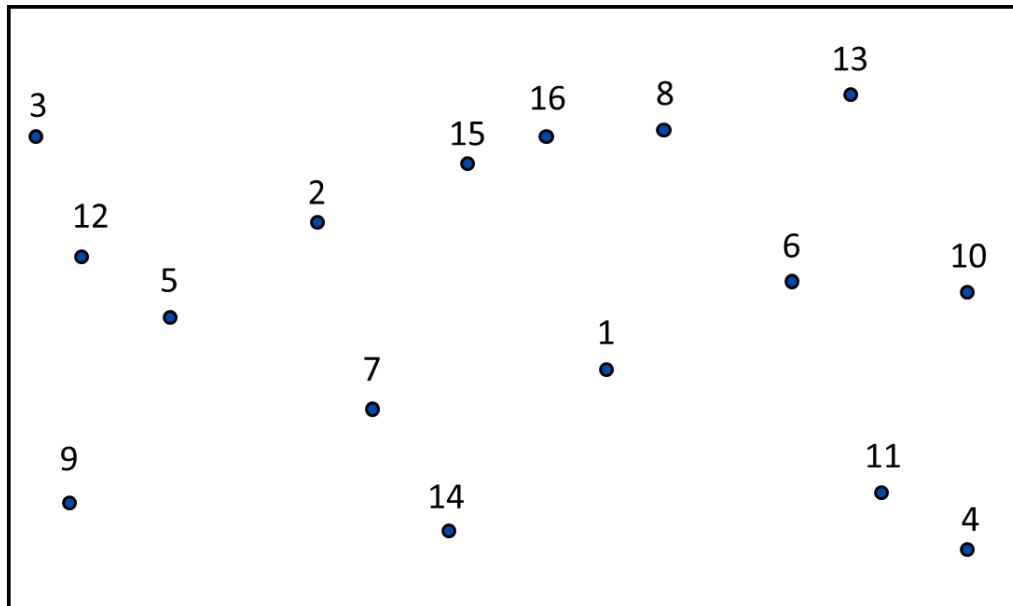
## One Last Thing....

---

**Can we bring it down to  $n \log n$ ?**

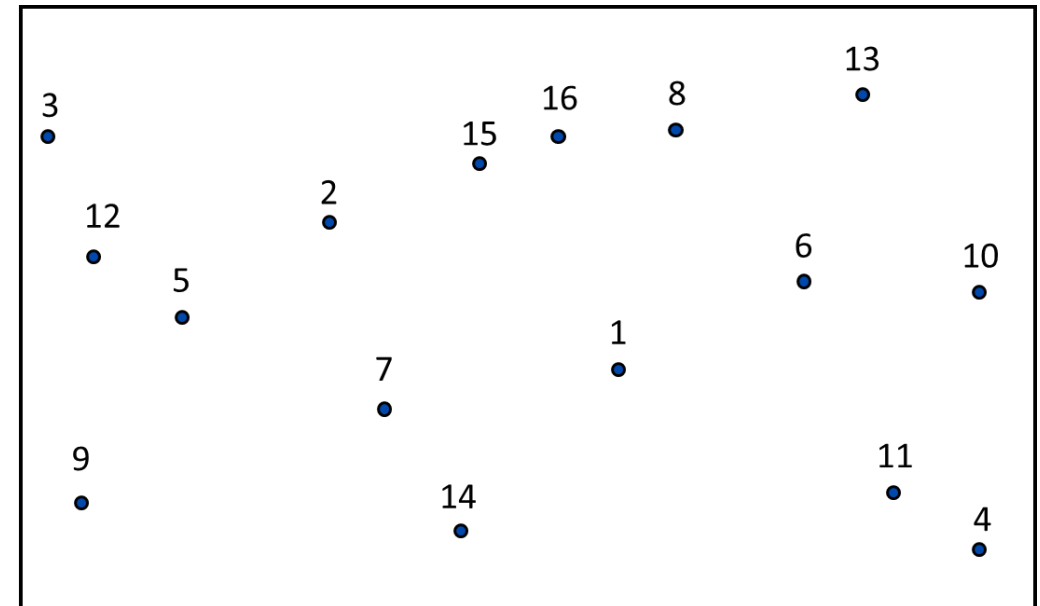


## Sorted X-axis



(3, 9, 12, 5, 2, 7, 15, 14, 16, 1, 8, 6, 13, 11, 10, 4)

## Sorted Y-axis



(13, 8, 16, 3, 15, 2, 12, 6, 10, 5, 1, 7, 9, 11, 14, 4)

Sort points based on X-coordinates  $\longrightarrow O(n \log n)$

**CLOSEST-PAIR**( $p_1, p_2, \dots, p_n$ )

---

Compute vertical line  $L$  such that half the points  
are on each side of the line.  $\longrightarrow O(n)$

$\delta_1 \leftarrow$  **CLOSEST-PAIR**(points in left half).  $\longrightarrow T(n/2)$

$\delta_2 \leftarrow$  **CLOSEST-PAIR**(points in right half).  $\longrightarrow T(n/2)$

$\delta \leftarrow \min \{ \delta_1, \delta_2 \}$ .  $\longrightarrow O(1)$

$A \leftarrow$  list of all points closer than  $\delta$  to line  $L$ .  $\longrightarrow O(n)$

Sort points in  $A$  by y-coordinate.  $\longrightarrow O(n \log n)$

Scan points in  $A$  in y-order and compare distance between each point and next  
7 neighbors.  $\longrightarrow O(n)$

If any of these distances is less than  $\delta$ , update  $\delta$ .  $\longrightarrow O(1)$

**RETURN**  $\delta$ .

---

Sort points based on X-coordinates  $\longrightarrow O(n \log n)$   
Sort points based on Y-coordinates  $\longrightarrow O(n \log n)$

**CLOSEST-PAIR**( $p_1, p_2, \dots, p_n$ )

Compute vertical line  $L$  such that half the points  
are on each side of the line.  $\longrightarrow O(n)$

$\delta_1 \leftarrow$  **CLOSEST-PAIR**(points in left half).  $\longrightarrow T(n/2)$

$\delta_2 \leftarrow$  **CLOSEST-PAIR**(points in right half).  $\longrightarrow T(n/2)$

$\delta \leftarrow \min \{ \delta_1, \delta_2 \}$ .  $\longrightarrow O(1)$

$A \leftarrow$  list of all points closer than  $\delta$  to line  $L$ .  $\longrightarrow O(n)$

Sort points in  $A$  by  $y$ -coordinate.  $\longrightarrow O(n \log n)$

Scan points in  $A$  in  $y$ -order and compare distance between each point and next  
7 neighbors.  $\longrightarrow O(n)$

If any of these distances is less than  $\delta$ , update  $\delta$ .  $\longrightarrow O(1)$

**RETURN**  $\delta$ .

Sort points based on X-coordinates  $\longrightarrow O(n \log n)$   
Sort points based on Y-coordinates  $\longrightarrow O(n \log n)$

**CLOSEST-PAIR**( $p_1, p_2, \dots, p_n$ )

Compute vertical line  $L$  such that half the points  
are on each side of the line.  $\longrightarrow O(n)$

$\delta_1 \leftarrow$  **CLOSEST-PAIR**(points in left half).  $\longrightarrow T(n/2)$

$\delta_2 \leftarrow$  **CLOSEST-PAIR**(points in right half).  $\longrightarrow T(n/2)$

$\delta \leftarrow \min \{ \delta_1, \delta_2 \}$ .  $\longrightarrow O(1)$

$A \leftarrow$  list of all points closer than  $\delta$  to line  $L$ .  $\longrightarrow O(n)$

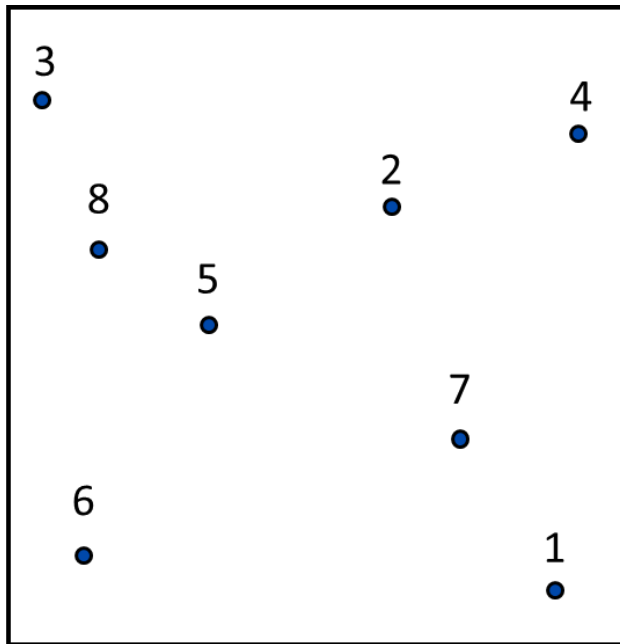
Sort points in  $A$  by  $y$ -coordinate.  $\longrightarrow O(n)$

Scan points in  $A$  in  $y$ -order and compare distance between each point and next  
7 neighbors.  $\longrightarrow O(n)$

If any of these distances is less than  $\delta$ , update  $\delta$ .  $\longrightarrow O(1)$

**RETURN**  $\delta$ .

Sorted X-axis



( 3 , 6 , 8 , 5 , 2 , 7 , 1 , 4 )

{(1,8),(2,2),(3,4),(4,5),(5,6),(7,4),(8,1),(8,7)}

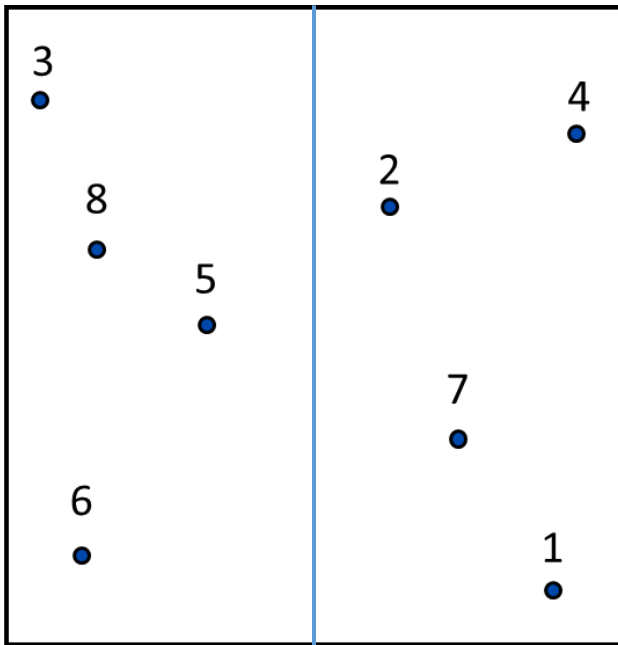
Sorted Y-axis



( 1 , 6 , 7 , 5 , 8 , 2 , 4 , 3 )

{(8,1),(2,2),(7,4),(4,5),(3,6),(5,6),(8,7),(1,8)}

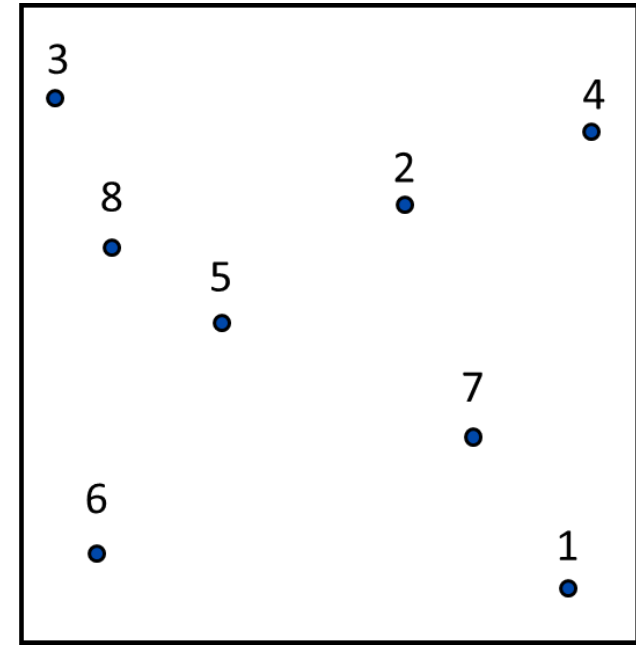
Sorted X-axis



( 3 , 6 , 8 , 5 , 2 , 7 , 1 , 4 )

{(1,8),(2,2),(3,4),(4,5),(5,6),(7,4),(8,1),(8,7)}

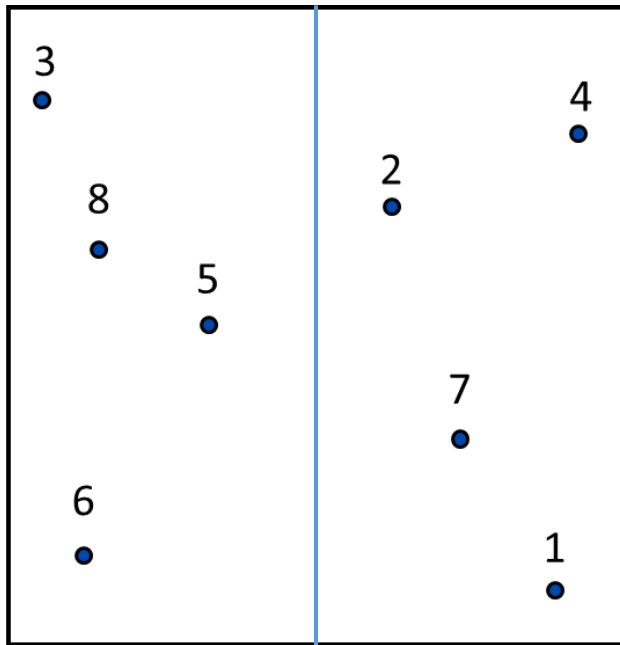
Sorted Y-axis



( 1 , 6 , 7 , 5 , 8 , 2 , 4 , 3 )

{(8,1),(2,2),(7,4),(4,5),(3,6),(5,6),(8,7),(1,8)}

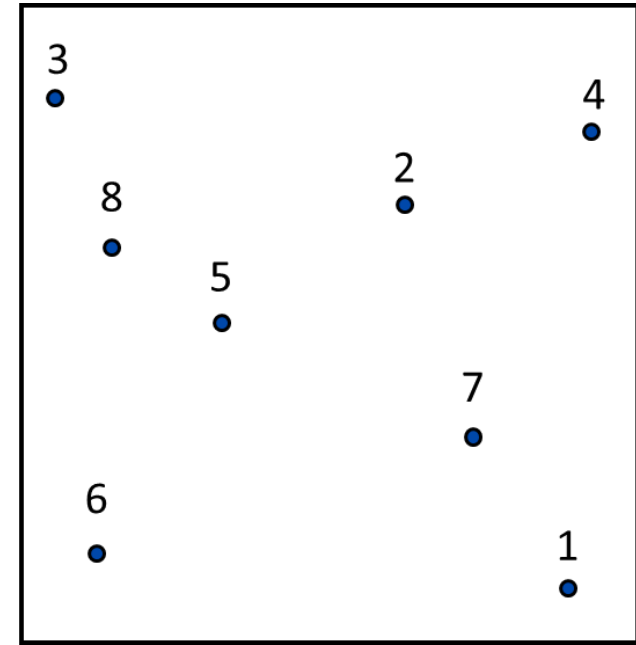
Sorted X-axis



$(3, 6, 8, 5) (2, 7, 1, 4)$

$\{(1,8), (2,2), (3,4), (4,5)\} \{(5,6), (7,4), (8,1), (8,7)\}$

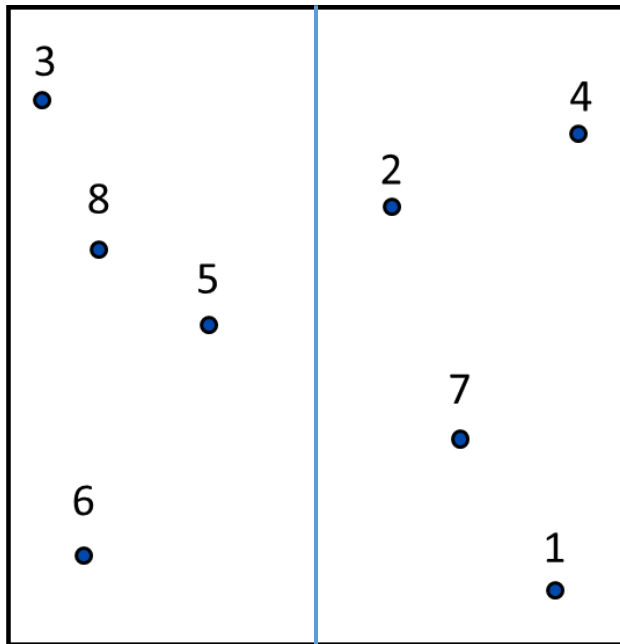
Sorted Y-axis



$(1, 6, 7, 5, 8, 2, 4, 3)$

$\{(8,1), (2,2), (7,4), (4,5), (3,6), (5,6), (8,7), (1,8)\}$

## Sorted X-axis



( 3 , 6 , 8 , 5 ) ( 2 , 7 , 1 , 4 )

{(1,8),(2,2),(3,4),(4,5)} {(5,6),(7,4),(8,1),(8,7)}

( 1 , 6 , 7 , 5 , 8 , 2 , 4 , 3 )

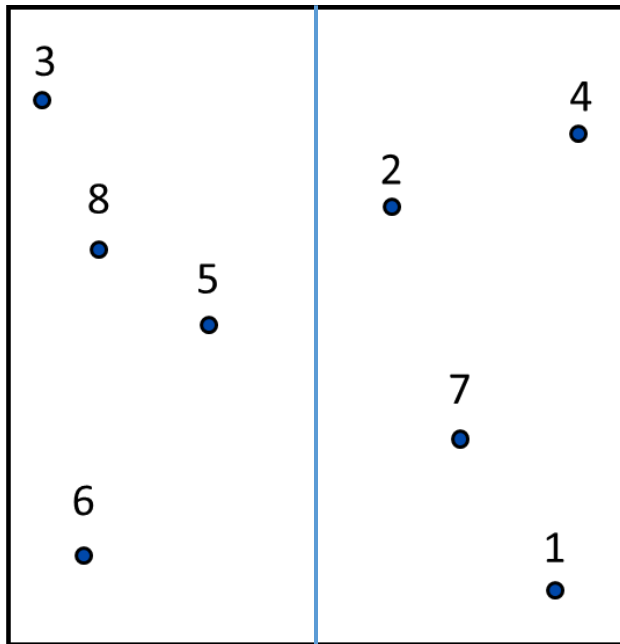
{(8,1),(2,2),(7,4),(4,5),(3,6),(5,6),(8,7),(1,8)}

If (***X\_coordinate*** =< 4)

else ()



## Sorted X-axis



$(3, 6, 8, 5) (2, 7, 1, 4)$

$\{(1,8), (2,2), (3,4), (4,5)\} \{(5,6), (7,4), (8,1), (8,7)\}$

$(1, 6, 7, 5, 8, 2, 4, 3)$

$\{(8,1), (2,2), (7,4), (4,5), (3,6), (5,6), (8,7), (1,8)\}$

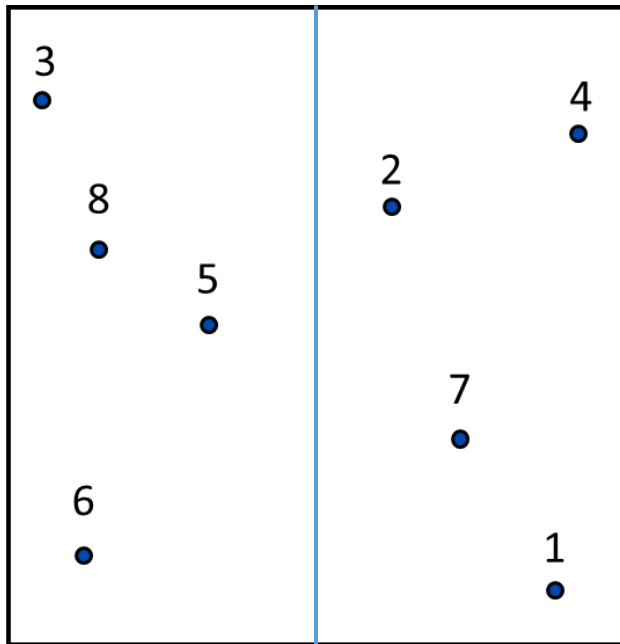


If ( $X\_coordinate \leq 4$ )

else ()

1  
(8,1)

## Sorted X-axis



( 3 , 6 , 8 , 5 ) ( 2 , 7 , 1 , 4 )

{(1,8),(2,2),(3,4),(4,5)} {(5,6),(7,4),(8,1),(8,7)}

( 1 , 6 , 7 , 5 , 8 , 2 , 4 , 3 )

{(8,1),(2,2),(7,4),(4,5),(3,6),(5,6),(8,7),(1,8)}



If (***X\_coordinate*** <= 4)

6

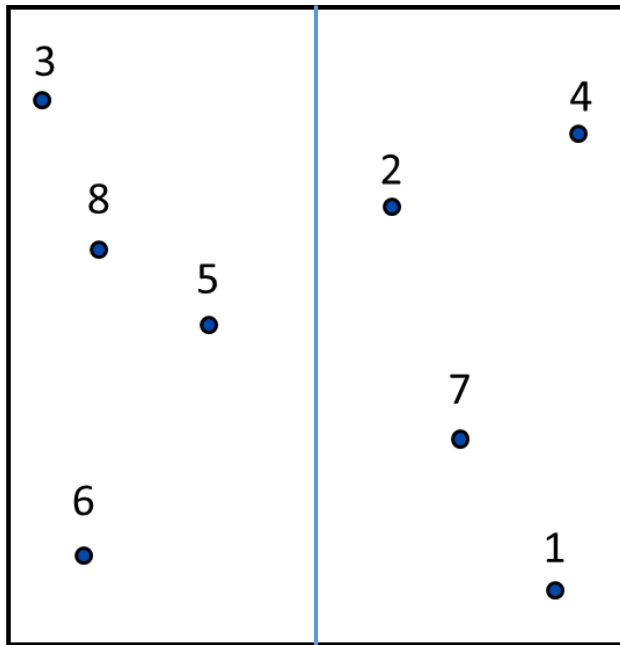
(2,2)

else ()

1

(8,1)

## Sorted X-axis



( 3 , 6 , 8 , 5 ) ( 2 , 7 , 1 , 4 )

{(1,8),(2,2),(3,4),(4,5)} {(5,6),(7,4),(8,1),(8,7)}

( 1 , 6 , 7 , 5 , 8 , 2 , 4 , 3 )

{(8,1),(2,2),(7,4),(4,5),(3,6),(5,6),(8,7),(1,8)}



If (***X\_coordinate*** <= 4)

6

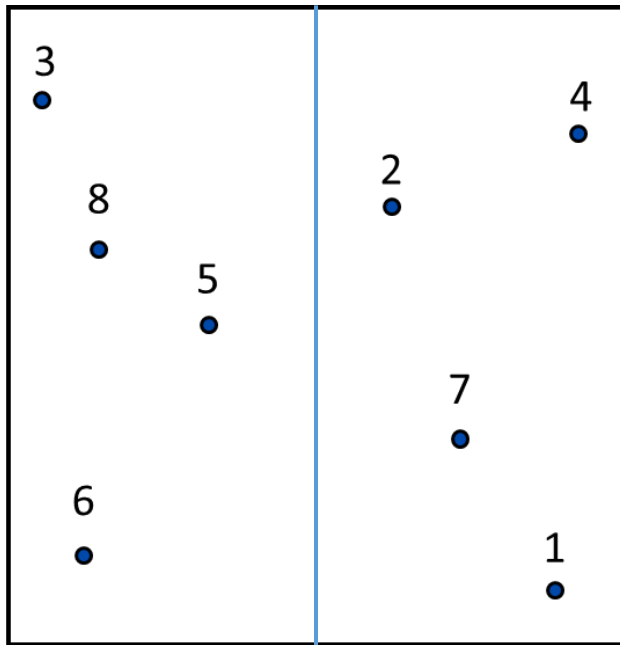
(2,2)

else ()

1 7

(8,1) (7,4)

## Sorted X-axis



$(3, 6, 8, 5) (2, 7, 1, 4)$

$\{(1,8), (2,2), (3,4), (4,5)\} \{(5,6), (7,4), (8,1), (8,7)\}$

$(1, 6, 7, 5, 8, 2, 4, 3)$

$\{(8,1), (2,2), (7,4), (4,5), (3,6), (5,6), (8,7), (1,8)\}$



If ( $X\_coordinate \leq 4$ )

6 5

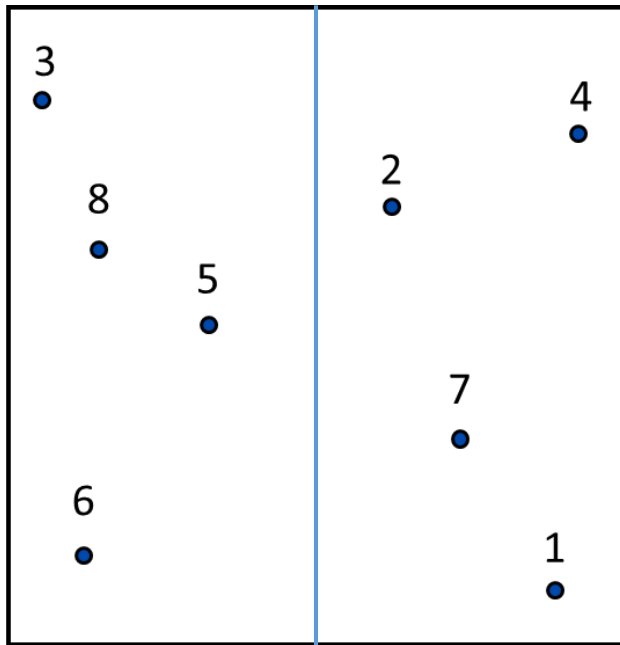
$(2,2) (4,5)$

else ()

1 7

$(8,1) (7,4)$

## Sorted X-axis



$(3, 6, 8, 5) (2, 7, 1, 4)$

$\{(1,8), (2,2), (3,4), (4,5)\} \{(5,6), (7,4), (8,1), (8,7)\}$

$(1, 6, 7, 5, 8, 2, 4, 3)$

$\{(8,1), (2,2), (7,4), (4,5), (3,6), (5,6), (8,7), (1,8)\}$



If ( $X\_coordinate \leq 4$ )

6 5 8

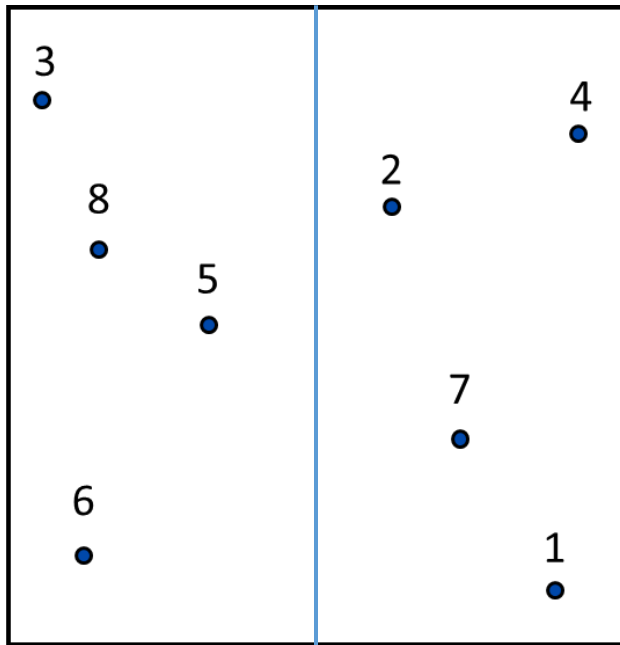
$(2,2) (4,5) (3,6)$

else ()

1 7

$(8,1) (7,4)$

## Sorted X-axis



$(3, 6, 8, 5) (2, 7, 1, 4)$

$\{(1,8), (2,2), (3,4), (4,5)\} \{(5,6), (7,4), (8,1), (8,7)\}$

$(1, 6, 7, 5, 8, 2, 4, 3)$

$\{(8,1), (2,2), (7,4), (4,5), (3,6), (5,6), (8,7), (1,8)\}$



If  $(X\_coordinate \leq 4)$

6 5 8

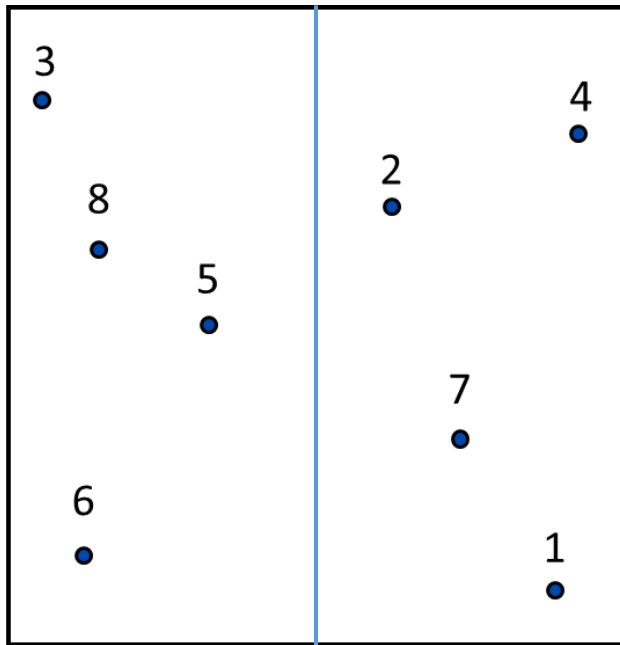
$(2,2) (4,5) (3,6)$

else ()

1 7 2

$(8,1) (7,4) (5,6)$

## Sorted X-axis



$(3, 6, 8, 5) (2, 7, 1, 4)$   
 $\{(1,8), (2,2), (3,4), (4,5)\} \{(5,6), (7,4), (8,1), (8,7)\}$

$(1, 6, 7, 5, 8, 2, 4, 3)$   
 $\{(8,1), (2,2), (7,4), (4,5), (3,6), (5,6), (8,7), (1,8)\}$



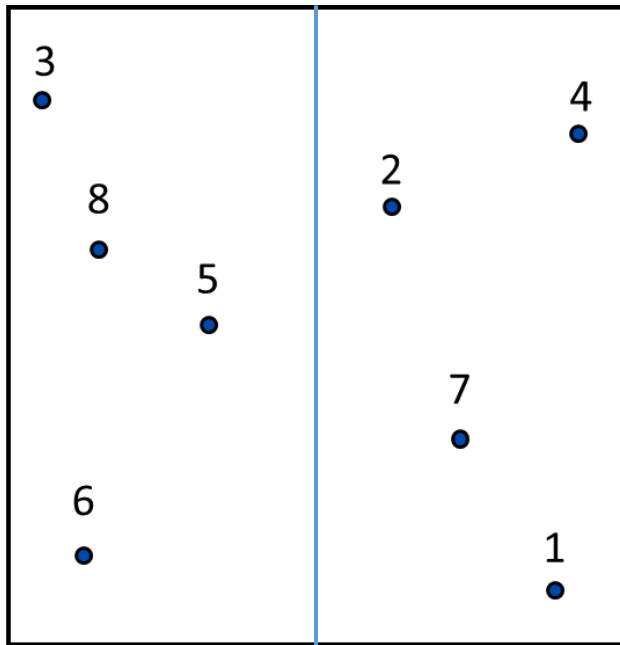
If ( $X\_coordinate \leq 4$ )

6 5 8  
 $(2,2) (4,5) (3,6)$

else ()

1 7 2 4  
 $(8,1) (7,4) (5,6) (8,7)$

## Sorted X-axis



$(3, 6, 8, 5) (2, 7, 1, 4)$

$\{(1,8), (2,2), (3,4), (4,5)\} \{(5,6), (7,4), (8,1), (8,7)\}$

$(1, 6, 7, 5, 8, 2, 4, 3)$

$\{(8,1), (2,2), (7,4), (4,5), (3,6), (5,6), (8,7), (1,8)\}$



If ( $X\_coordinate \leq 4$ )

6 5 8 3

$(2,2) (4,5) (3,6) (1,8)$

else ()

1 7 2 4

$(8,1) (7,4) (5,6) (8,7)$



# Thanks a lot



If you are taking a Nap, **wake up**.....Lecture Over