NTNU
Kunnskap for en bedre verden

Divide & Conquer

TDT4121 Assignment Lecture 28/09

Part 1

- Understand the divide and conquer algorithm design method
- Understand mergesort
- Understand how ideas from mergesort are used to solve the Counting Inversions problem
- Understand the Finding the Closest Pair of Points problem

Understand the divide and conquer algorithm design method – Three steps

- → 1. Break the input into several parts
- → 2. Solve the problem in each part recursively
- → 3. Combine these solutions to an overall solution

NTNU
Kunnskap for en bedre verden

$$5*2+3/6$$

NTNU
Kunnskap for en bedre verden

5*2+3/6

→ 1. 5*2 AND 3/6

→ 2. 10 AND 0.5

→ 3. 10.5

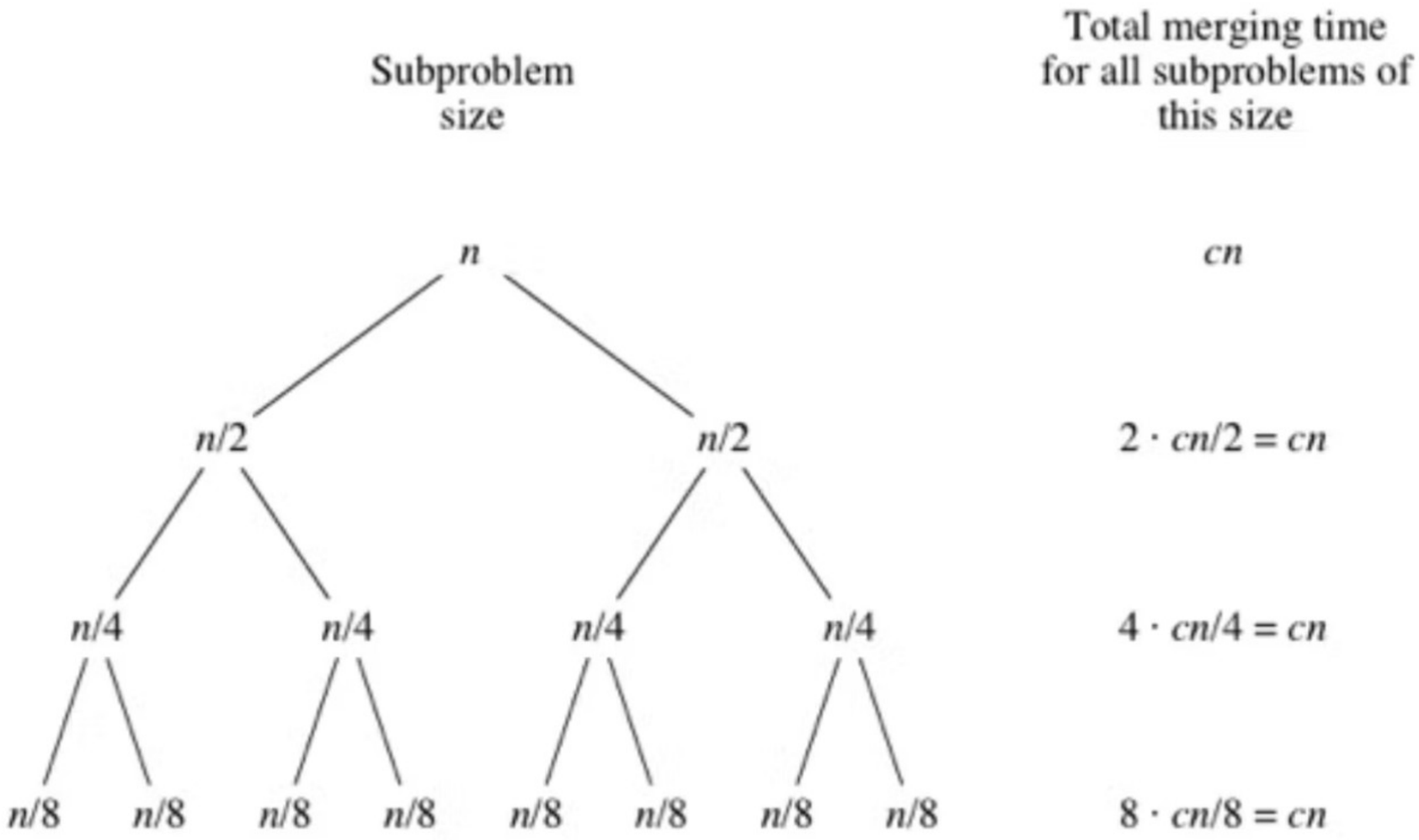
NTNU
Kunnskap for en bedre verden

Understand Mergesort


Merge sort in 3 minutes
Share
Merge Sort in 3
Watch on YouTube

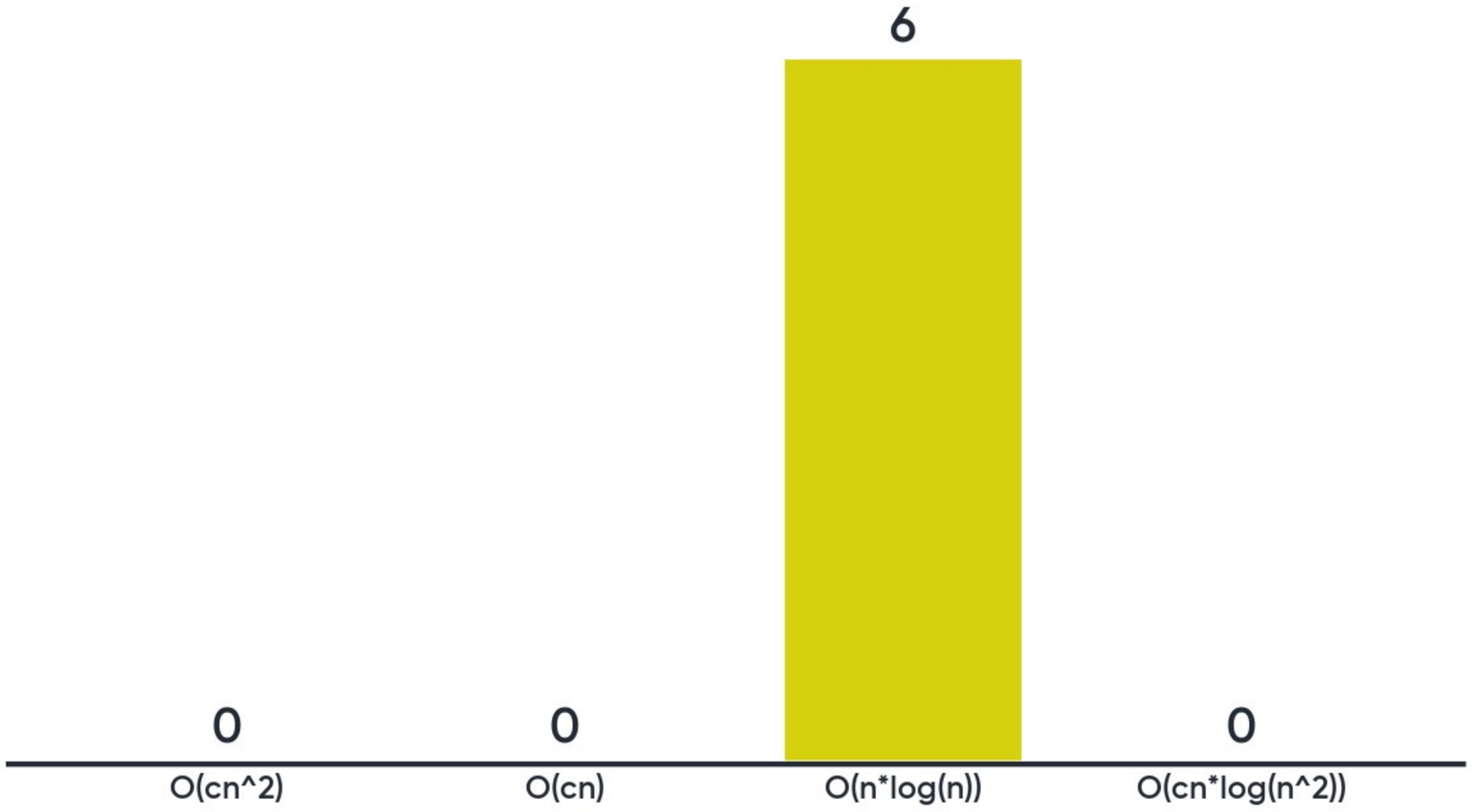
Runtime

- 1. The divide step takes constant time, regardless of the subarray size: O(1)
- 2. The conquer step, where we recursively sort two subarrays of approximately $n/2$ elements each, takes some amount of time
- 3. Combining these three takes O(n) time, by merging together n elements.



Hint: cn* height of three = Runtime

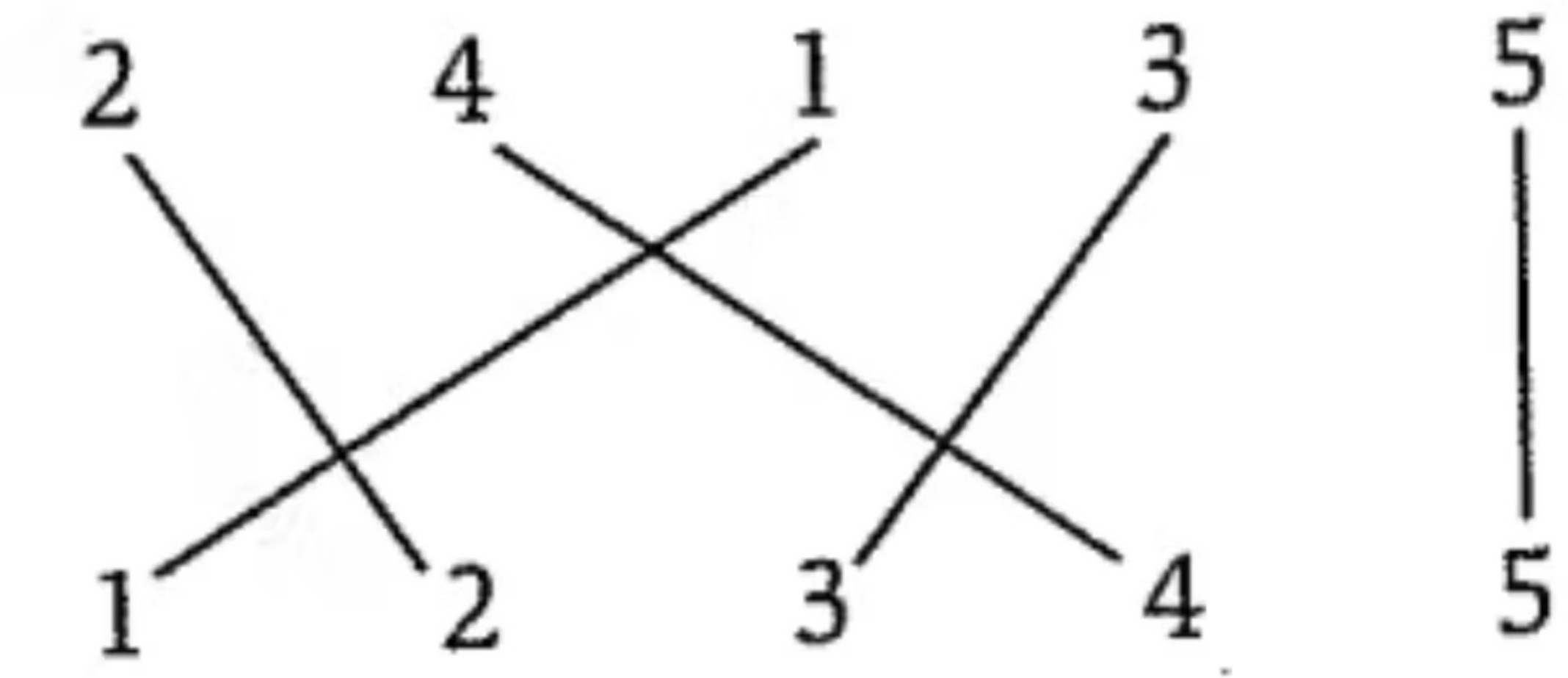
What is the runtime of Mergesort?



NTNU
Kunnskap for en bedre verden

Understand how ideas from mergesort are used to solve the Counting Inversions problem

- → 1. The Counting Inversions problem
- → 2. Using Mergesort to solve it
- → 3. Runtime



inversions are pairs of numbers, in a disordered list, where the larger of the two numbers is to the left of smaller number.

How many inversions?

0

1

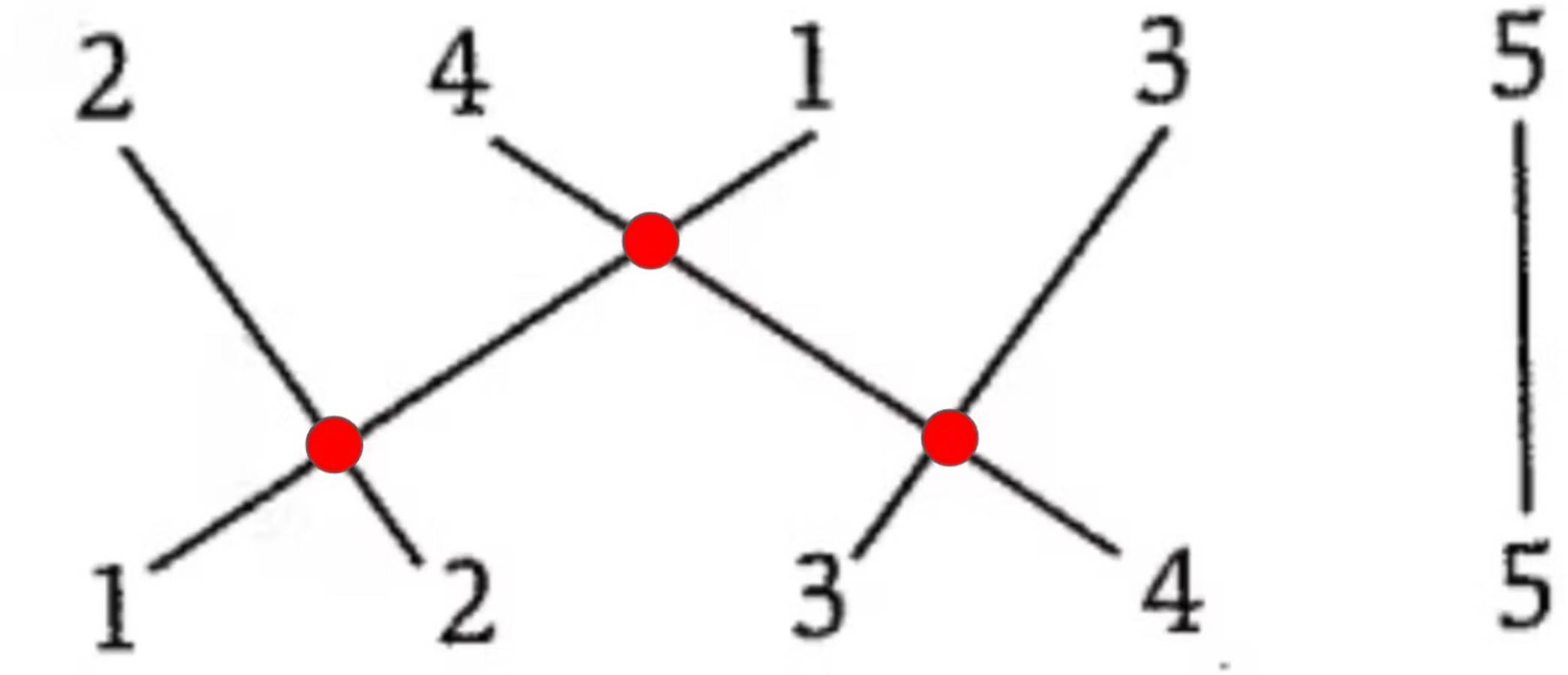
0

2

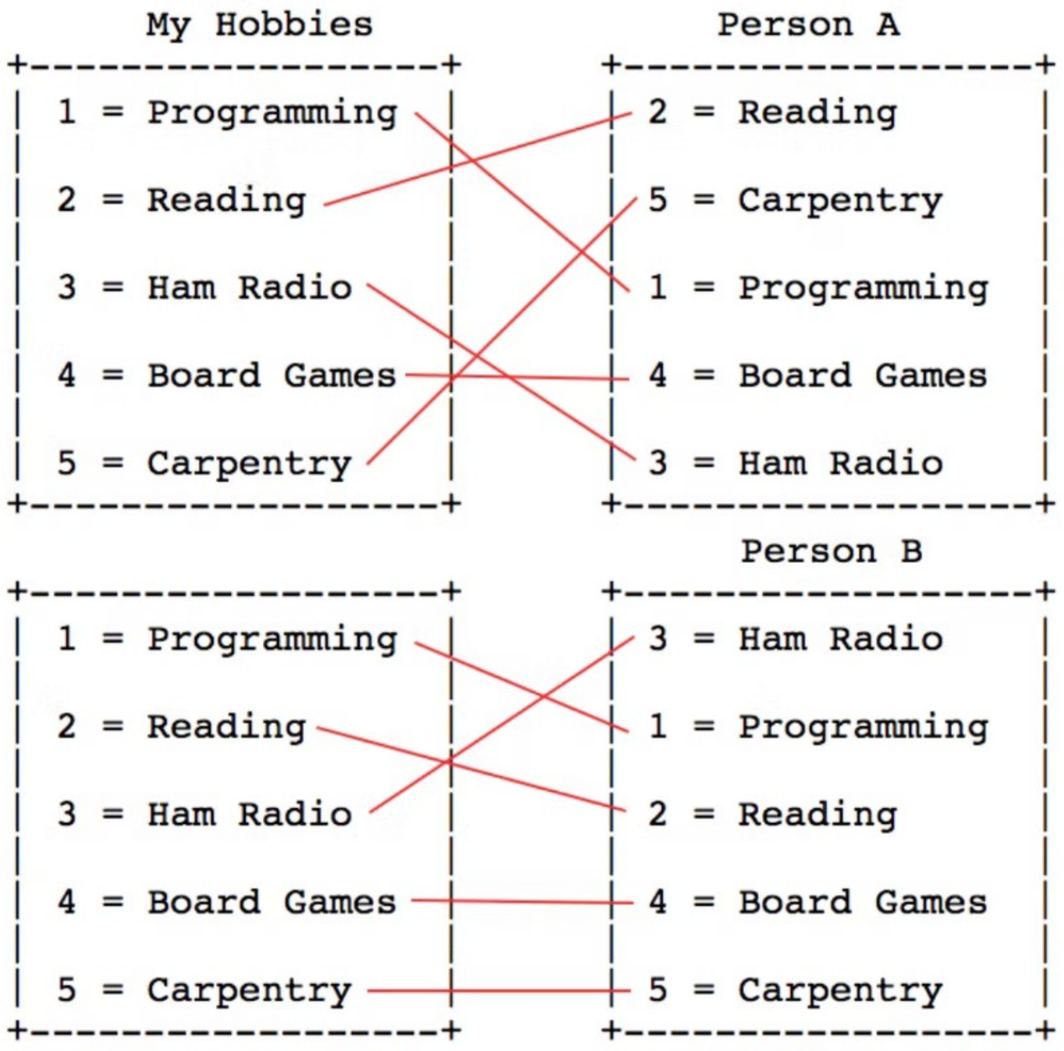
3

0

4



Trick: Count crosses

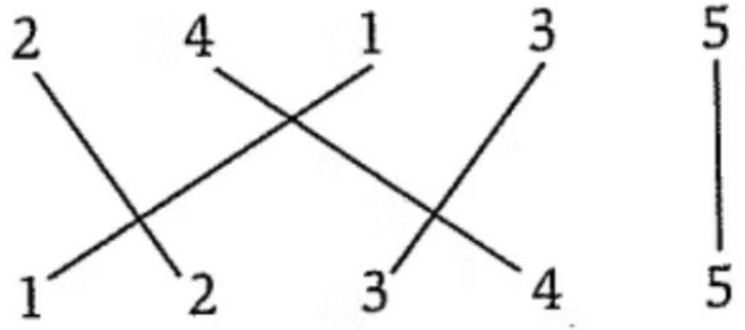


Task: Who should I hang out with?

Who should I hang out with?

NTNU
Kunnskap for en bedre verden

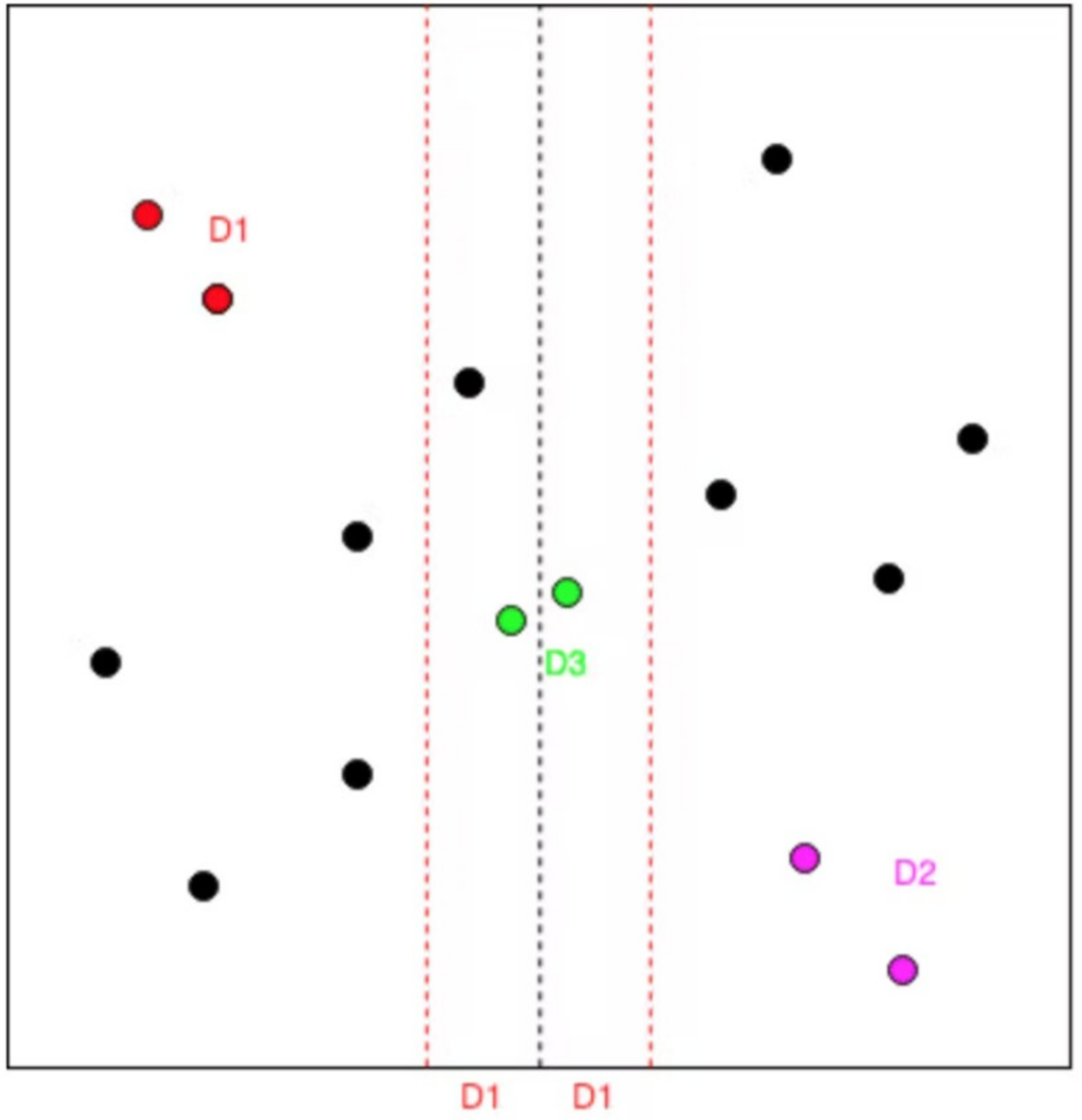
```
Sort-and-Count(L)
  If the list has one element then
      there are no inversions
  Else
      Divide the list into two halves:
         A contains the first ⌈n/2⌉ elements
         B contains the remaining ⌊n/2⌋ elements
      (r_A, A) = Sort-and-Count(A)
      (r_B, B) = Sort-and-Count(B)
      (r, L) = Merge-and-Count(A, B)
   Endif
   Return r = r_A + r_B + r, and the sorted list L
```

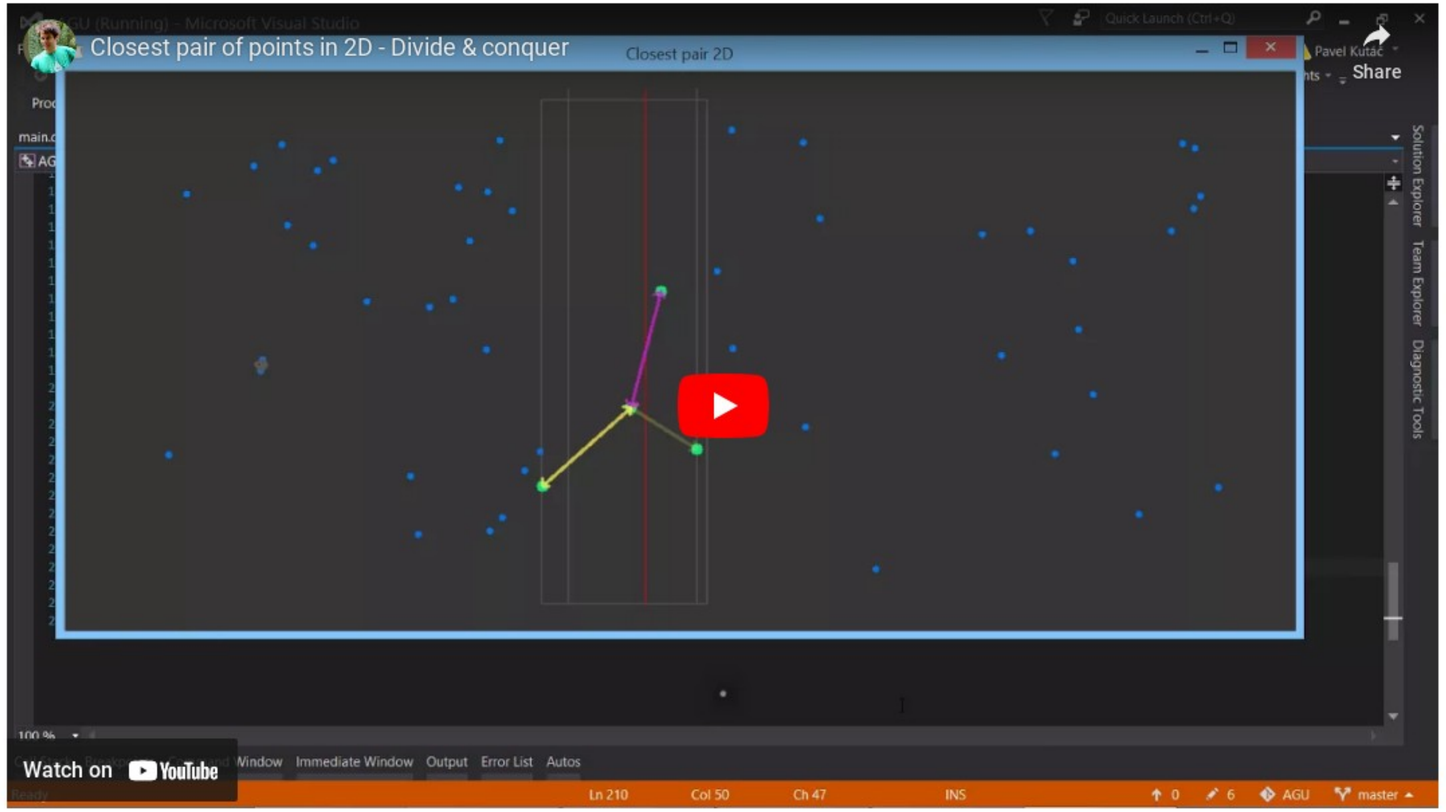


Sort-and-Count algorithm correctly sorts the input list and counts the number of inversions; it runs in O(n log n) time for a list with n elements.

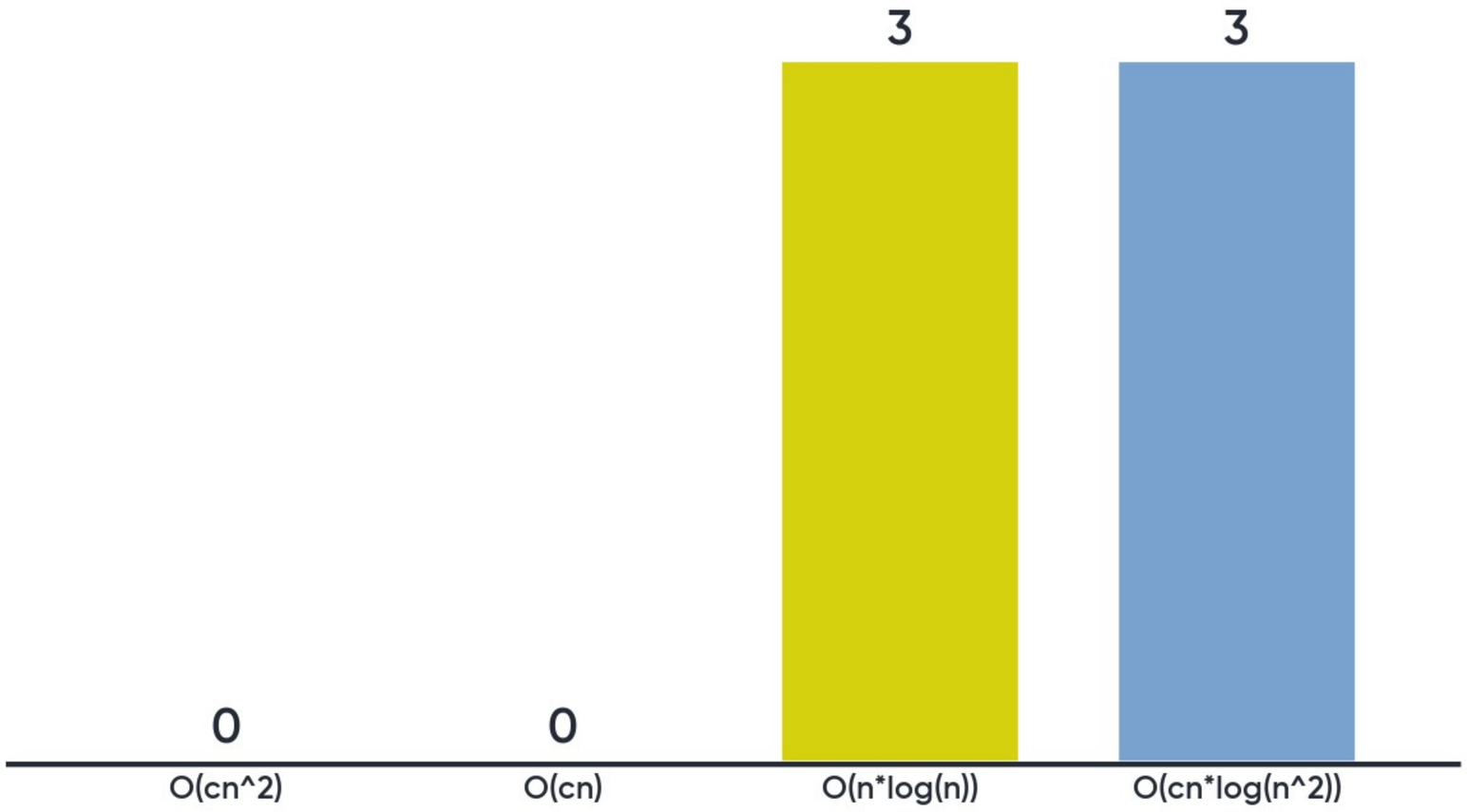
NTNU
Kunnskap for en bedre verden

Understand the Finding the Closest Pair of Points problem


D1
D3
D2
D1
D1


Closest pair of points in 2D - Divide & conquer
Closest pair 2D
Share
Watch on YouTube

What is the runtime of CPOP?



Part 2

- Understand the Integer Multiplication problem
- Solve a recurrence by unrolling
- Solve a recurrence by induction
- Understand the Strassen algorithm

The Integer Multiplication problem

- → Grade school: $O(n^2)$ 👎
- → Karatsuba-Ofman: $O(n^{1.585})$ 👍

```
Recursive-Multiply(x,y):
```

Write $x = x_1 \cdot 2^{n/2} + x_0$

$y = y_1 \cdot 2^{n/2} + y_0$

Compute $x_1 + x_0$ and $y_1 + y_0$

p = Recursive-Multiply$(x_1 + x_0,\ y_1 + y_0)$

$x_1 y_1$ = Recursive-Multiply(x_1, y_1)

$x_0 y_0$ = Recursive-Multiply(x_0, y_0)

Return $x_1 y_1 \cdot 2^n + (p - x_1 y_1 - x_0 y_0) \cdot 2^{n/2} + x_0 y_0$

Karatsuba-Ofman Integer Multiplication

Exercises: Karatsuba

1. Solve $36 \cdot 28$
2. Solve $101_2 \cdot 100_2$

Using Karatsuba, what is $11_2 \cdot 11_2$?

NTNU
Kunnskap for en bedre verden

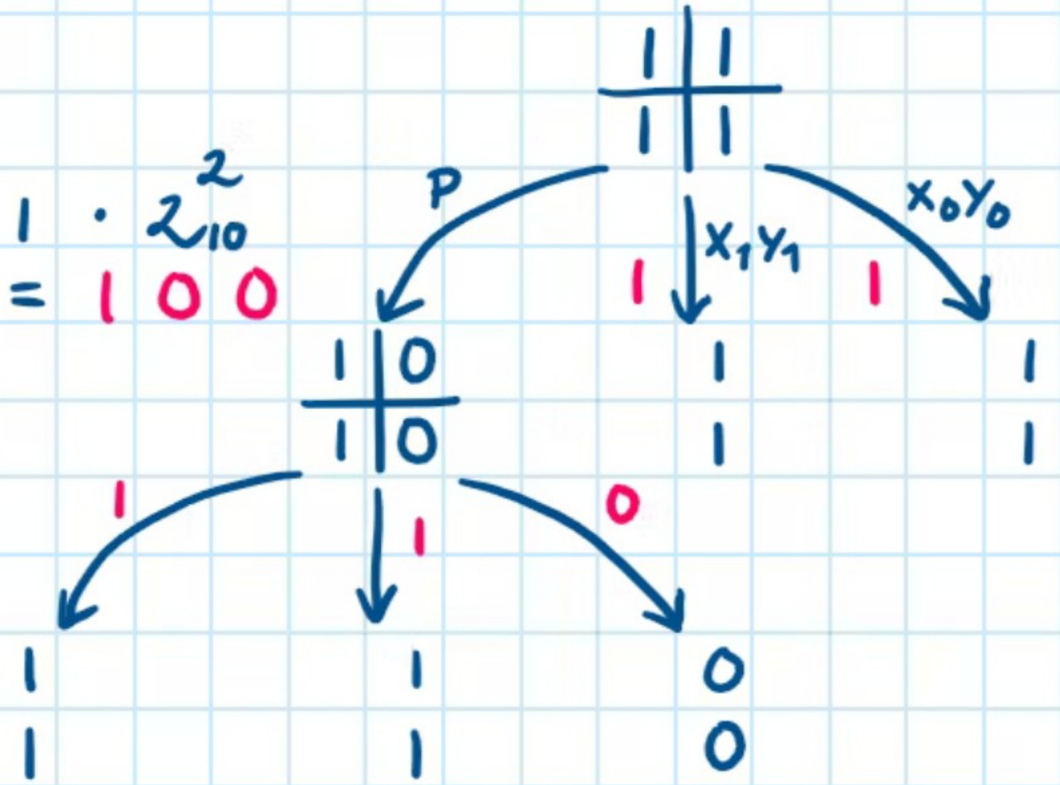
0
1010

1
1001

0
1111

0
0101

$1 \cdot 2^2_{10} + (100 - 1 - 1) \cdot 2_{10} + 1$
$= 100 + 100 + 1 = 1001$



What recurrence describes the Karatsuba algorithm?

NTNU
Kunnskap for en bedre verden

0
$T(n) = 4T(n/2) + cn$

2
$T(n) = 3T(n/2) + cn$

1
$T(n) = 3T(n/3) + cn$

0
$T(n) = 4T(n/3) + cn$

Exercises: Recurrences

1. Solve the Karatsuba recurrence by unrolling
2. Confirm your answer by substitution

Solve a recurrence by unrolling

1. Analyze the first few levels
2. Identify a pattern
3. Sum over all levels of recursion

Solve a recurrence by induction

1. Inductive hypothesis
2. Show inductive step

Exercises: Complexity

1. Plot the number of operations of Karatsuba for $1 < n < 900$
2. Confirm by plot that the complexity is $O(n^2)$
3. Confirm by plot that the complexity is $\Theta(n^{1.585})$

The Strassen algorithm

→ Matrix Multiplication in $O(n^{2.81})$

Exercises: Matrix Multiplication

- → Implement a naive algorithm and find the complexity
- → Implement a D&Q algorithm and find the recurrence
- → Implement Strassen and find the recurrence

What is the complexity of the naive matrix multiplication algorithm?

NTNU
Kunnskap for en bedre verden

0
O(n^2)

0
O(n*log(n))

0
O(n^log(n))

2
O(n^3)

What recurrence describes the D&Q matrix multiplication algorithm?

NTNU
Kunnskap for en bedre verden

0
$T(n) = 8T(n/2) + cn$

0
$T(n) = 4T(n/2) + cn^2$

2
$T(n) = 8T(n/2) + cn^2$

0
$T(n) = 4T(n/4) + cn$

What recurrence describes the Strassen matrix multiplication algorithm?

NTNU
Kunnskap for en bedre verden

1
$T(n) = 7T(n/2) + cn^2$

0
$T(n) = 8T(n/2) + cn$

0
$T(n) = 8T(n/2) + cn^2$

0
$T(n) = 7T(n/2) + cn$

2.12. How many lines, as a function of n (in $\Theta(\cdot)$ form), does the following program print? Write a recurrence and solve it. You may assume n is a power of 2.

```
function f(n)
  if n > 1:
    print_line(``still going'')
    f(n/2)
    f(n/2)
```

What recurrence describes the number of lines that will be printed?

NTNU
Kunnskap for en bedre verden

0
$T(n) = T(2n), T(1) = 1$

2
$T(n) = 2T(n/2) + 1, T(1) = 0$

0
$T(n) = 2T(n), T(0) = 1$

0
$T(n) = 2T(n/4) - 1, T(0) = 1$

How many lines will be printed?

NTNU
Kunnskap for en bedre verden

2
Theta(log(n))

0
Theta(n*log(n))

0
Theta(n)

0
Theta(n^2)

Ask me anything

NTNU
Kunnskap for en bedre verden

0 questions
0 upvotes