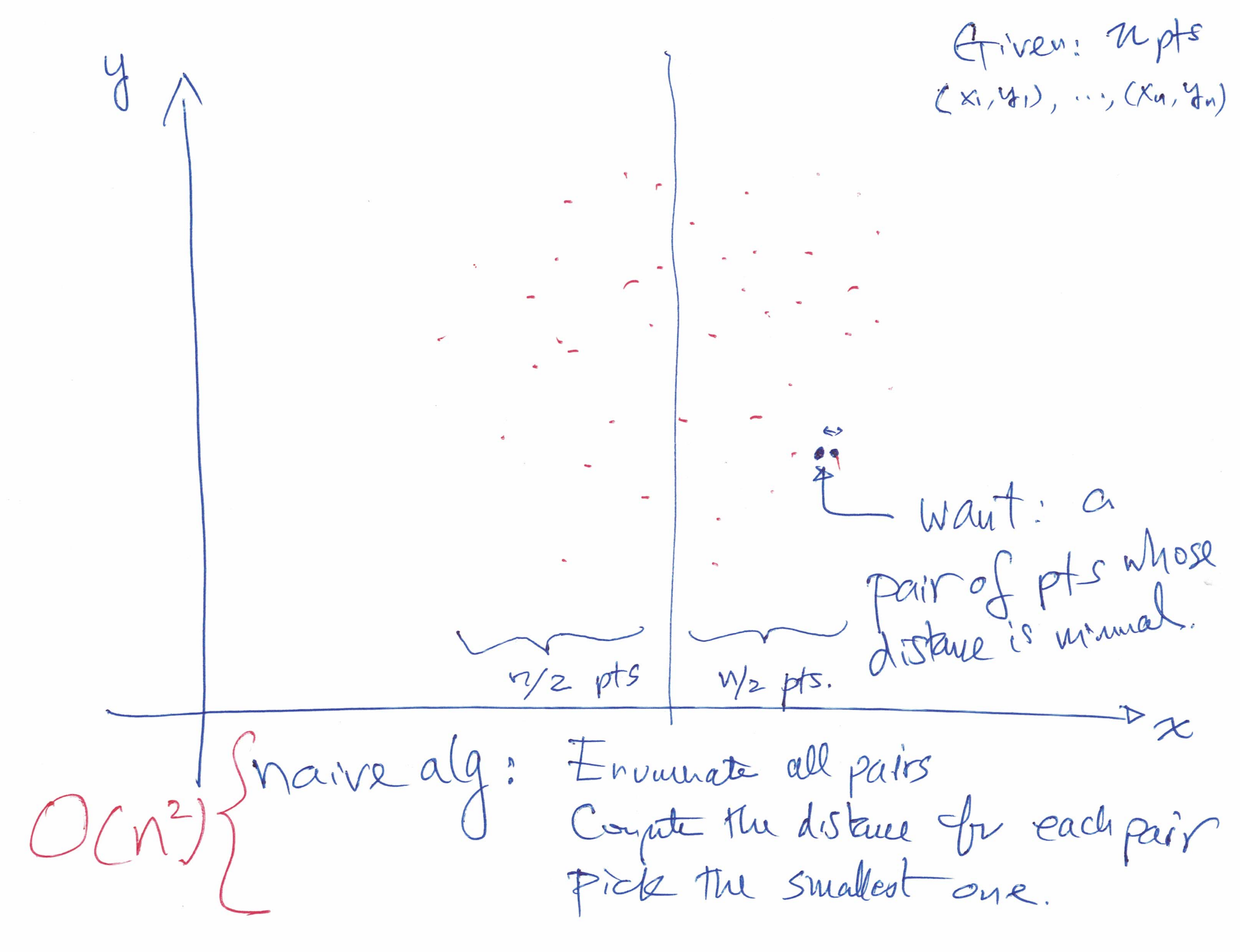
Cpts 515. 9/4/2020

Today: More classic Algs.

① Closest pair ← Divide & Conquer.

② Selection in Linear time.

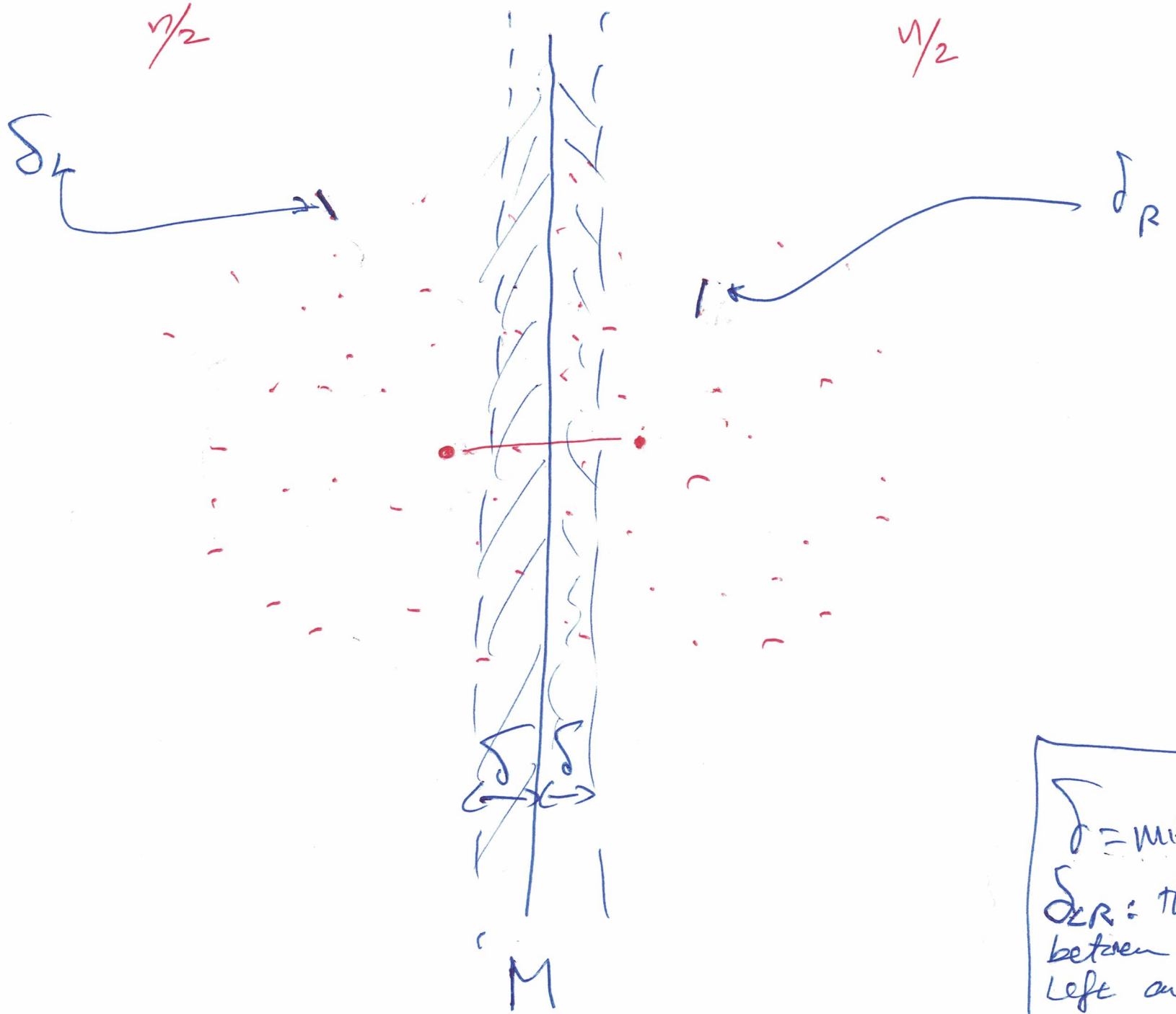

Given: n pts
$(x_1, y_1), \ldots, (x_n, y_n)$
y
x
want: a pair of pts whose distance is minimal.
n/2 pts
n/2 pts.
$O(n^2)$
naive alg:
Enumerate all pairs
Compute the distance for each pair
pick the smallest one.

Closest-pair: $O(n \log n)$ on n pts:

↑ Divide & Conquer.

1. Cut the n pts into two parts ⟶ We have a middle line M.

2. Recursively, run closest-pair on the left $n/2$ pts, and on the right $n/2$ pts.



$\delta = \min(\delta_L, \delta_R)$

δ_{LR}: the closest between a pt on Left and a pt on the right.

(cont.) 2. Then, we have

δ_L: closest pair on the left

δ_R: closest pair on the right

Take $\delta := \min(\delta_L, \delta_R)$.

---

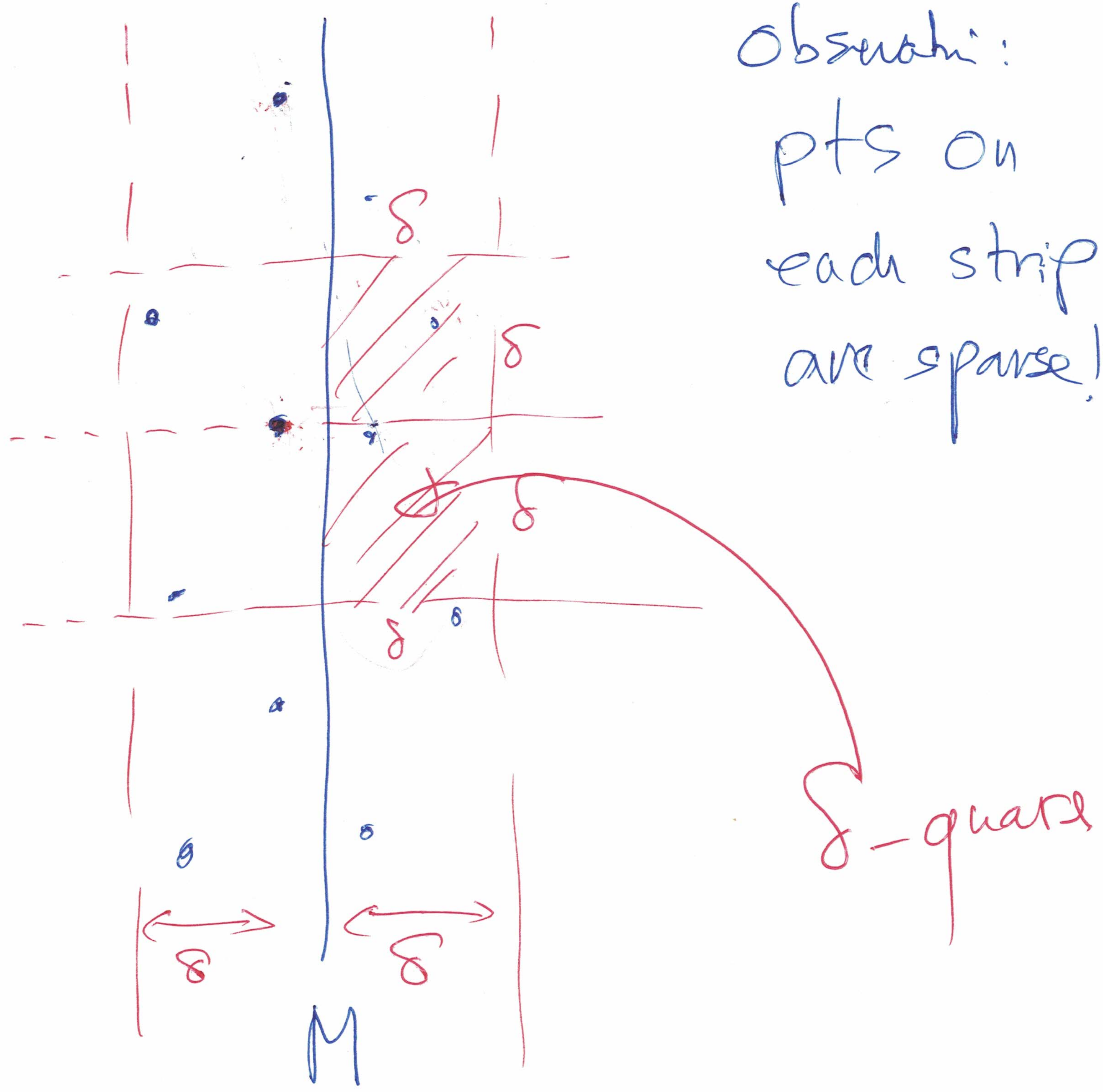
3. Conquer.

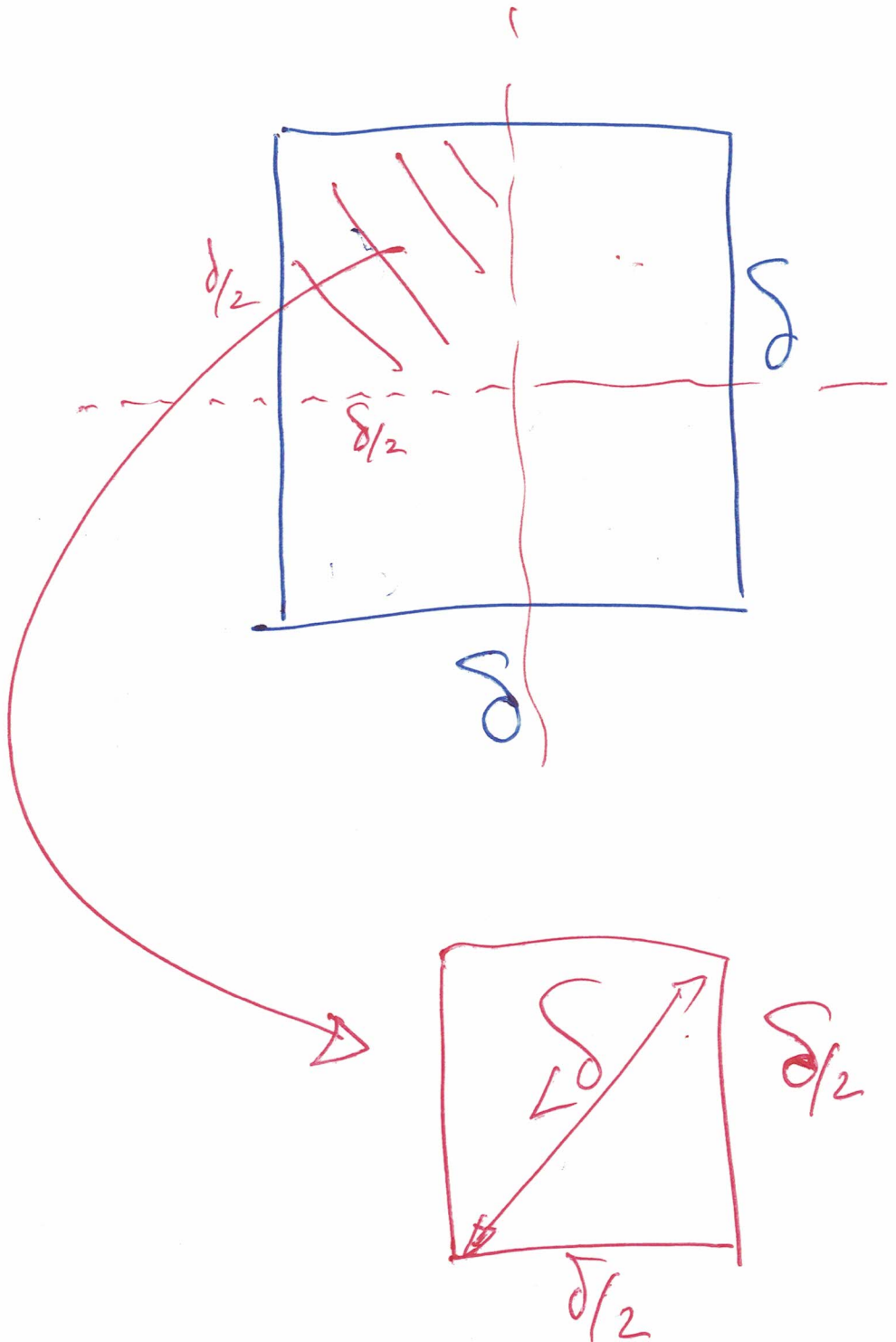
3.1. Create two δ-strips.

3.2. tricky! $\Rightarrow$ it takes linear time!

δ
δ
too small
must be

Observation: pts on each strip are sparse!





how many pts can I put into this δ-square s.t. any two pts stay away by ≥ δ?

What we learned:

Divide & Conquer = super efficient conquer part!

Selection in linear time in worst case.

Given: an array A of n numbers & a number i.

Goal: Can you tell me the i-th smallest number in the array A?

Example. 6, 2, 1, -2, 11, 8

What is the 1-st smallest? -2

what is the 2-nd smallest? 1

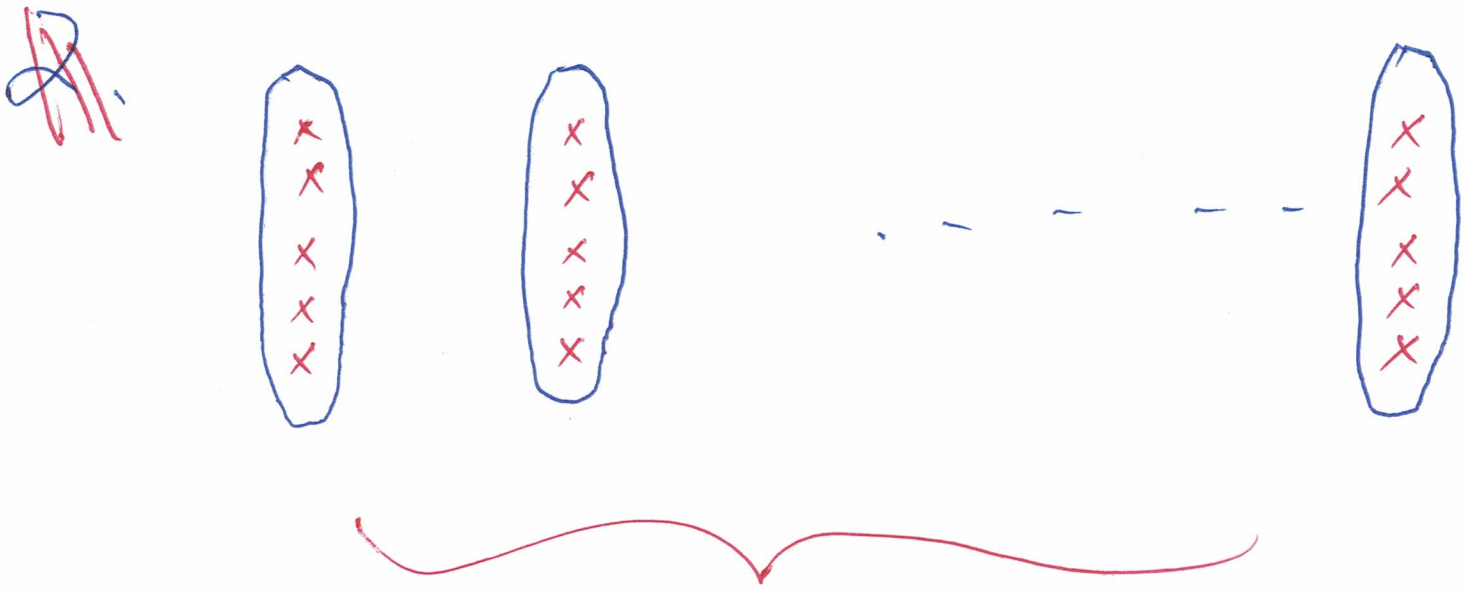
... 3-rd ... ? 2

... 6-th smallest? 11

so select doesn't work.

y
z
synapses

1. Cut the n nodes into $n/5$ groups (each group has 5 pts).



$n/5$ groups.

2. Sort each group.

Weekend challenges:

① Can you sort 5 numbers using maximally 7 comparisons?

↳ is the best!

---

② Can you write a program that generates the method of sorting in ① for $n=5$? (how about $n=10$?)

↳ you need invent a beautiful method, ~~program~~ data structure for the ~~program~~.