January 17, 2023 1:34 PM

CS 473 Algorithms

https://courses.engr.illinois.edu/cs473

10 HWs (each w. 3 problems)		30%
Midterm 1	(Feb 20 Mon 7p-9p)	20%
Midterm 2	(Apr 3 Mon 7p-9p)	20%
Final Exam		30%

1. no late HW accepted (due Wed 9pm)
2. may work in groups ≤ 3
3. read academic integrity page (cheating penalty severe)

Topics
- Divide & Conquer (e.g. FFT)
- Dynamic Programming
- Randomized Algms
- Optimization (matching, flows, LP)
- NP-completeness & reductions
- Approx. Alg'ms

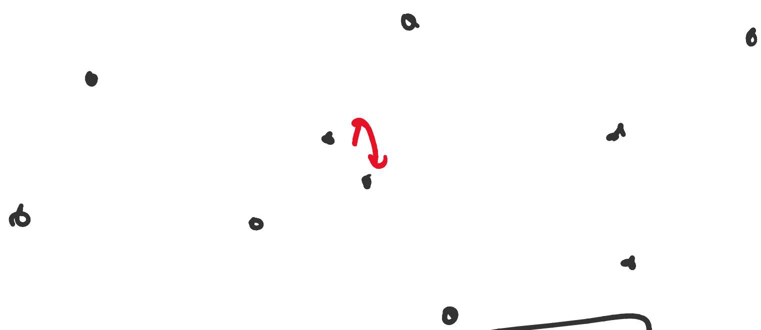
DIVIDE & CONQUER

divide into subproblems of same type
recurse
combine

Closest Pair

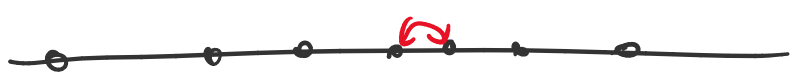
Problem: Given set P of n points in 2D

Problem Given set P of n points in 2D,
find pair $p, q \in P$
with smallest distance
$$d(p,q) = \sqrt{(p.x - q.x)^2 + (p.y - q.y)^2}$$



brute force alg'm: $\boxed{O(n^2)}$ time
better?

in 1D:

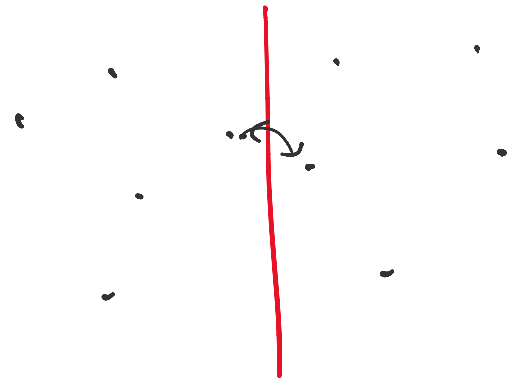


$O(n \log n)$ time by sorting & checking all consecutive pairs

in 2D?

Shamos's Alg'm (1975)

idea 0 - divide by vertical median line



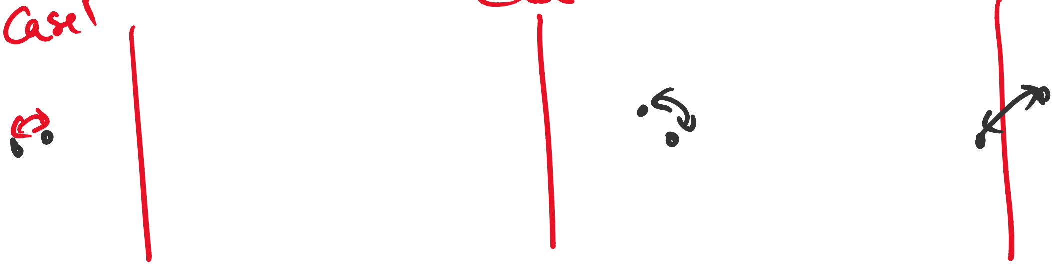
ClosestPair(P):

$O(n \log n)$ by sorting; $\Theta(n)$; $2T(\frac{n}{2})$

1. if $|P| \leq 3$ return answer by brute force
2. x_m = median x-coord.
3. $P_L = \{p \in P : p.x \leq x_m\}$
4. $P_R = \{p \in P : p.x > x_m\}$
5. δ_L = ClosestPair(P_L)
6. δ_R = ClosestPair(P_R)
7. $\delta = \min\{\delta_L, \delta_R\}$

... (not done yet) ...

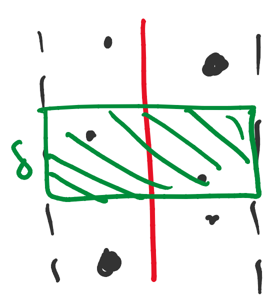
Case 1 Case 2 Case 3



still $\frac{n}{2} \times \frac{n}{2} = \Theta(n^2)$ pairs to check!

1st idea – only need to check pairs inside strip of width 2δ

$$x_m - \delta \leq x \leq x_m + \delta$$



2nd idea: only need to check pairs inside some $2\delta \times \delta$ rectangle

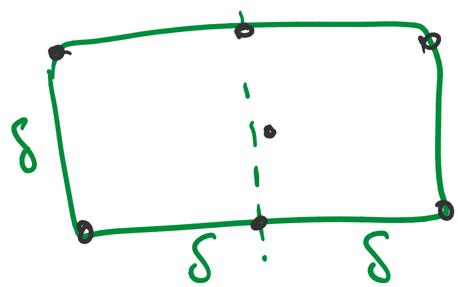
$\overleftrightarrow{\delta}\ \overleftrightarrow{\delta}$

$$x_m - \delta \le x \le x_m + \delta$$

$$t \le y \le t + \delta \qquad \text{for some } t$$

Lemma Each such $2\delta \times \delta$ rectangle may contain $\le \cancel{8}$ pts 6?

Pf:



left $\delta \times \delta$ squares ≤ 4 pts
right " " "

$\square$

Nn c c

⋮

$O(n\log n)$ 8. $\langle p_1, \ldots, p_\ell \rangle$ = points in $\{p \in P : x_m - \delta \le p.x \le x_m + \delta\}$ sorted in y-coord

$O(n)$
9. for $i = 1$ to ℓ do
10. for $j = i+1, i+2, \ldots$ as long as $p_j.y \le p_i.y + \delta$ ≤ 7 iterations
11. $\delta = \min\{\delta, d(p_i, p_j)\}$
12. return δ

Analysis:

$$T(n) = \begin{cases} 2T\left(\frac{n}{2}\right) + O(n\log n) & \text{if } n > 3 \\ O(1) & \text{else} \end{cases}$$

$$T(n) = \begin{cases} \ldots \\ O(1) & \text{else} \end{cases}$$

$n \log n \rightarrow n \log n$

$\frac{n}{2}\log\frac{n}{2} \quad \frac{n}{2}\log\frac{n}{2} \rightarrow n \log \frac{n}{2}$

$\frac{n}{4}\log\frac{n}{4} \quad \ldots \quad \frac{n}{4}\log\frac{n}{4} \rightarrow n \log \frac{n}{4}$

$\vdots$

$$T(n) \leq \boxed{O(n \log^2 n)}$$

Rmk: can be improved by pre-sorting

$$\Rightarrow T(n) = 2T\left(\frac{n}{2}\right) + O(n)$$

$$\Rightarrow O(n \log n) + O(n \log n) = \boxed{O(n \log n)}$$

3D: $O(n \log n)$

4D: $O(n \log n)$

5D: $\vdots$

Polynomial Multiplication / Convolution

Problem Given 2 polynomials $P(x)$, $Q(x)$

Problem Given 2 polynomials $P(x)$, $Q(x)$ in one var x, of deg $n-1$, compute new polynomial $P(x)\cdot Q(x)$.

e.g. $(x^2+x+5)\cdot(3x^2+x+4)$

$$= 3x^4 + (1\cdot1+1\cdot3)x^3 + (1\cdot4+1\cdot1+5\cdot3)x^2 + (1\cdot4+5\cdot1)x + 5\cdot4$$

$$= 3x^4 + 4x^3 + 20x^2 + 9x + 20$$

in general, $P(x) = a_{n-1}x^{n-1} + \dots + a_1x + a_0$

$Q(x) = b_{n-1}x^{n-1} + \dots + b_1x + b_0$

$$P(x)\cdot Q(x) = c_{2n-2}x^{2n-2} + \dots + c_1x + c_0$$

where $\underbrace{c_k = \sum_{j=0}^{k} a_j b_{k-j}}$ $k = 0, \dots, 2n-2$

$\langle c_0, \dots, c_{2n-2}\rangle$ called convolution of sequences $\langle a_0, \dots, a_{n-1}\rangle$, $\langle b_0, \dots, b_{n-1}\rangle$

Obvious alg'm: $\boxed{O(n^2)}$ time

better?