374: Lecture 7 9/14/2021

Linear time algorithms

Median selection (selection)

$X = \{x_1, x_2, \ldots, x_n\}$ n real numbers

$X[1 \ldots n]$

The rank of y in X is the number of elements in X smaller or equal to y.

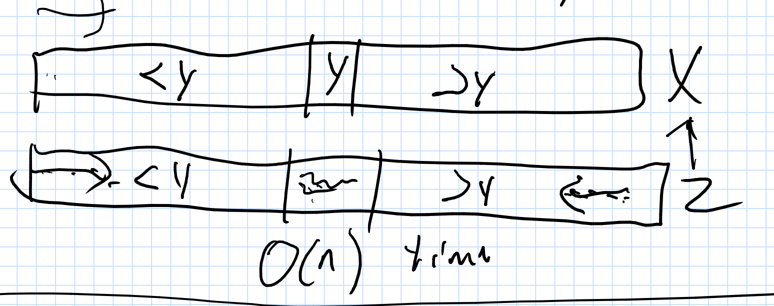
$X = \{17, 13, 26, 97, 10^{100}\}$
2 1 3 4 5

median of X is the element of rank $\lceil |X|/2 \rceil$.

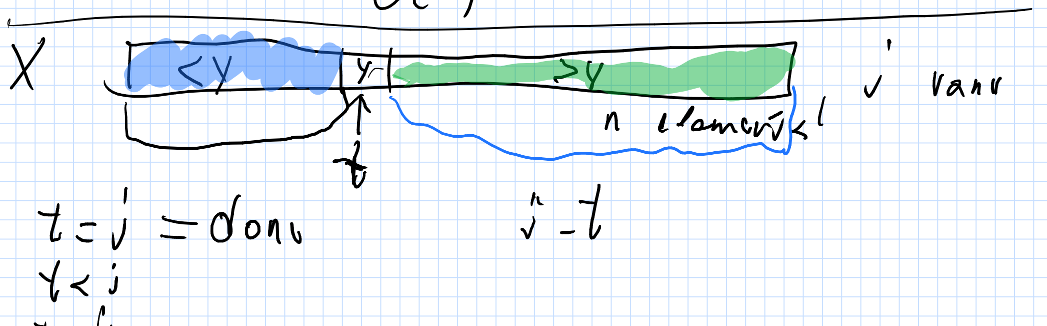
Compute the median of X.

Naive algorithm: sort ⇒ output mid element in sorted array. $O(n \log n)$

partition(X, y): X array y: number



$O(n)$ time

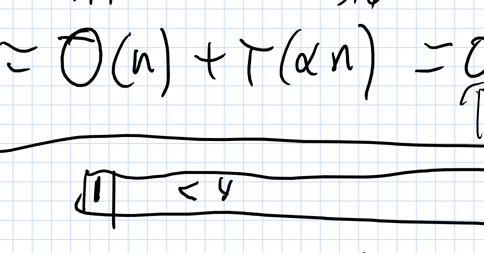


$t = i$ = done $i - t$
$t < i$
$t > i$

$T(n) = O(n) + T(\frac{n}{2})$
$= O(n) + O(\frac{n}{2}) + T(\frac{n}{4})$
$= O(n) + O(\frac{n}{2}) + O(\frac{n}{4}) + O(\frac{n}{8}) \ldots$
$= O(n)$.

$T(n) = O(n) + T(\frac{3}{4}n) = O(n)$

$T(n) = O(n) + T(\alpha n) = O(n) \quad \alpha \in (0,1)$

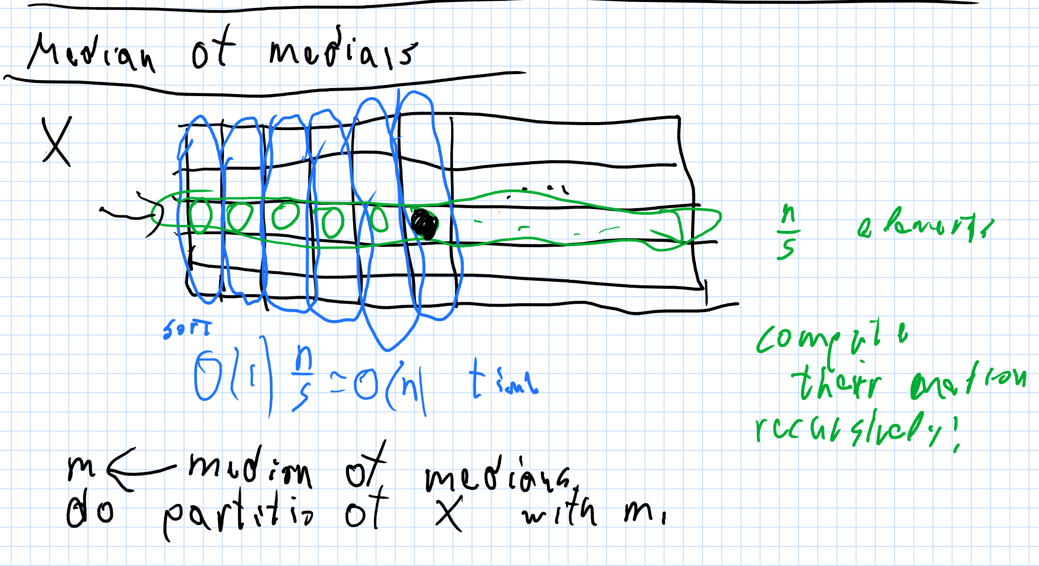


$T(n) = O(n) + T(n-1) = O(n^2)$.

QuickSelect: randomly pick the pivot.

$X[1 \ldots n] \quad X[i]$

Median of medians



$O(1)\frac{n}{5} = O(n)$ time

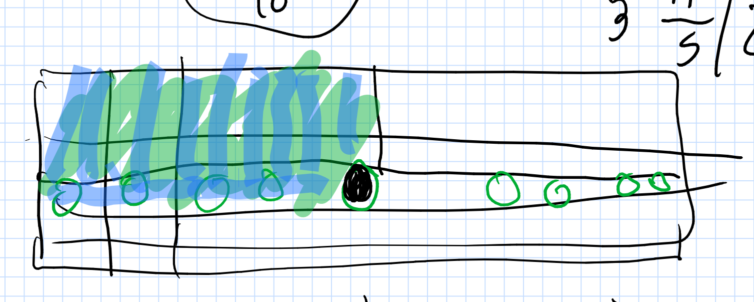
compute their median recursively!

$m \leftarrow$ median of medians.
do partition of X with m.

Claim

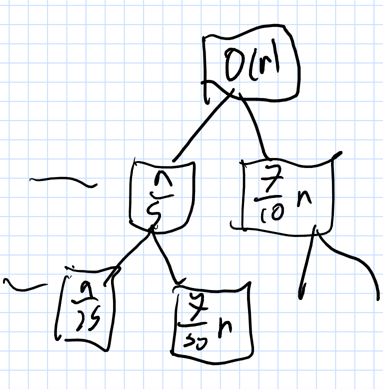
m partition X into two arrays each of size at most $\frac{7}{10}n$

$3 \frac{n}{5} / 2 = \frac{3}{10}n$



$T(n) = O(n) + T(\frac{n}{5}) + T(\frac{7}{10}n) = O(n)$

$T(n) \le cn$

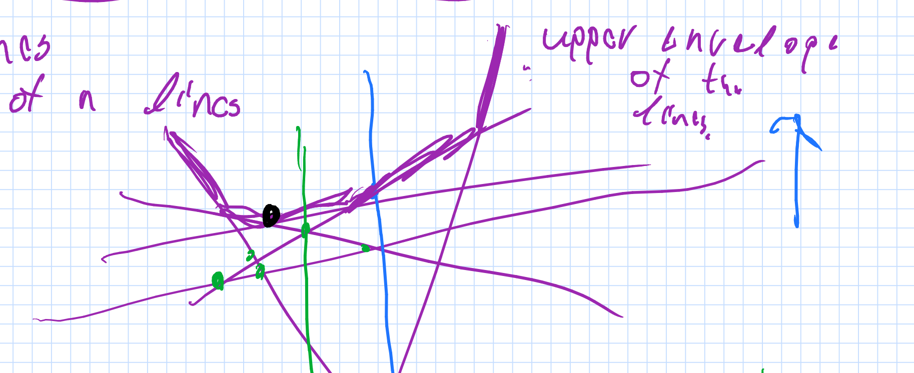


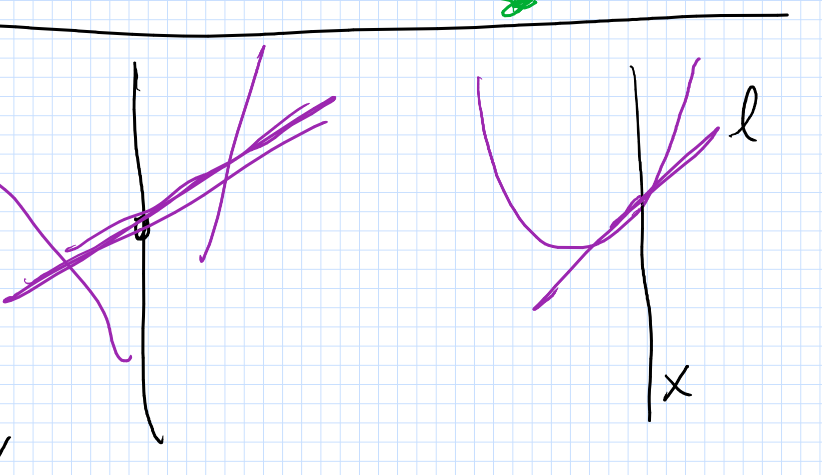
$\frac{1}{5} + \frac{7}{10} = \frac{9}{10} < 1$

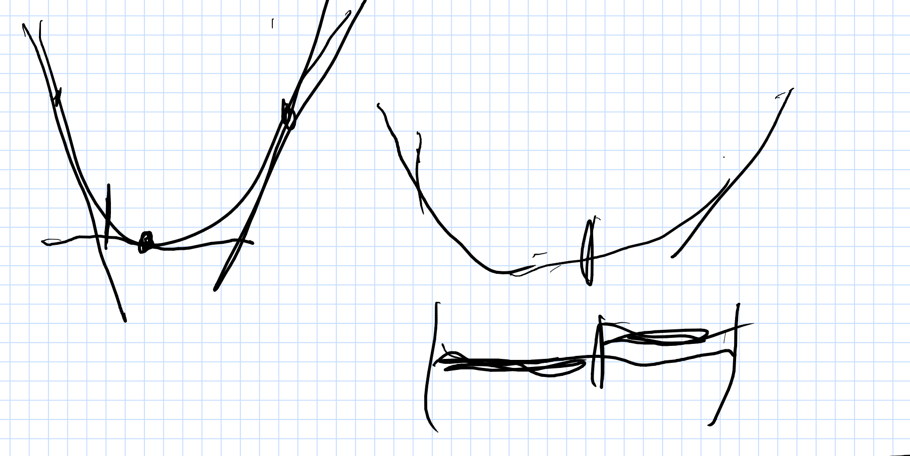
$T(n) = O(n) + T(\frac{n}{5}) + T(\frac{7}{10}n) = O(n)$.
$T(n) \le c'n + c\frac{n}{5} + \frac{7}{10}nc$
$= (c' + \frac{9}{10}c)n \le cn$
$c' + \frac{9}{10}c \le c$
$c \ge 10c'$

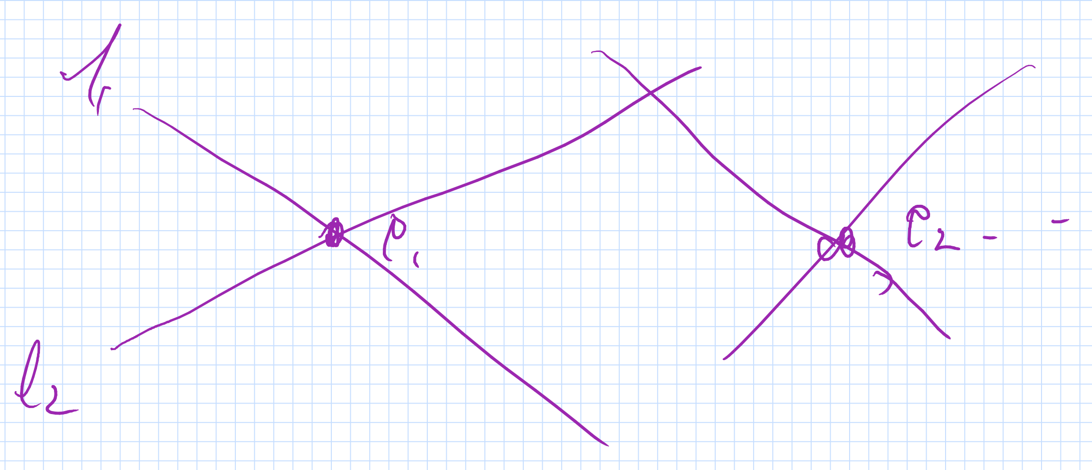
Lowest point above lines.

n lines
L: set of n lines



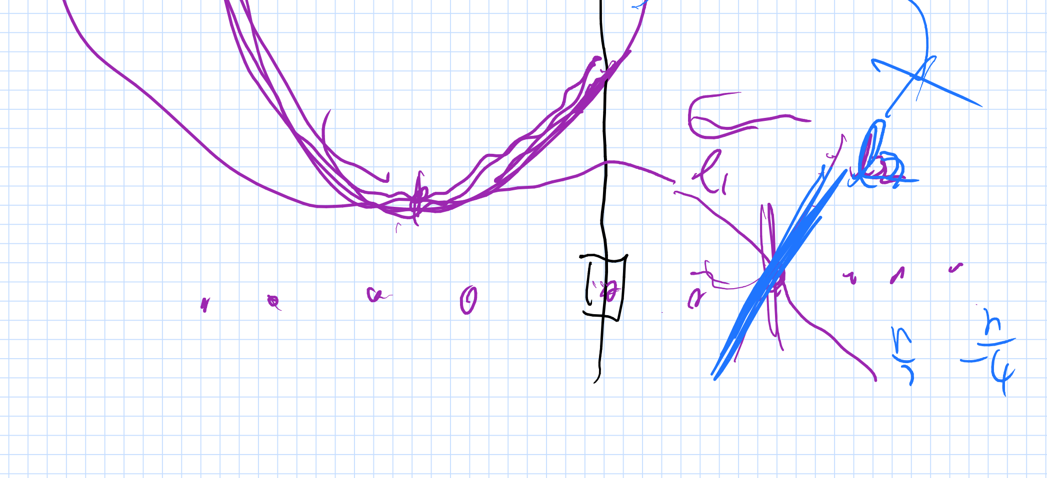






$L \Rightarrow \frac{n}{2}$ pairs intersection points
$P = \frac{n}{2}$ points

$p \leftarrow$ median point of P by the x coordinate



$L \Rightarrow L' \quad |L'| \le \frac{3}{4}n$

$L_1 \overset{O(n)}{\Rightarrow} L_2 \overset{O(\frac{3}{4}n)}{\Rightarrow} L_3$
$\frac{3}{4}n$

$T(n) = O(n) + T(\frac{3}{4}n) = O(n)$

Bottleneck MST edge

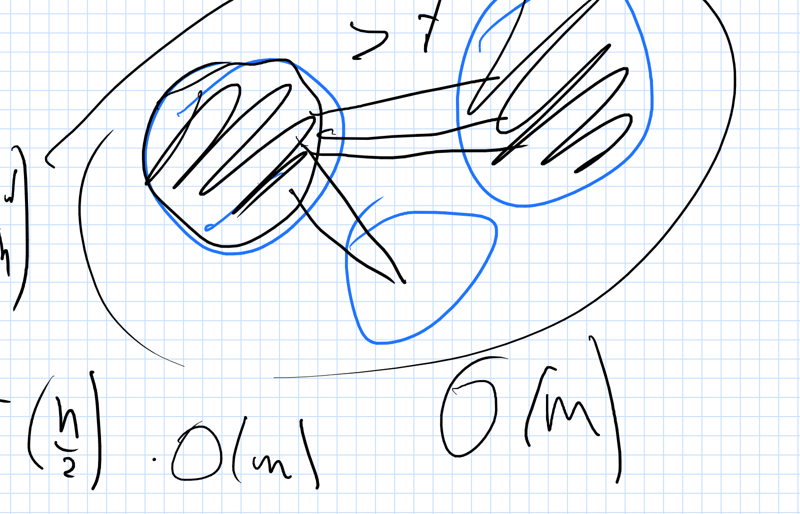
$G = (V, E)$ w: weights on the edges.
Compute heaviest edge in MST.
$O(n \log n + m)$ ⟵
$O(n+m)$: Randomized complicated)

$G_{\le t} = (V, \{uv \in E(G) \mid w(uv) \le t\})$
$b(G) \equiv$ weight of bottleneck edge in MST(G)
$b \equiv b(G)$

$b < t$ $G_{\le t}$ is connected
$b > t$ ⟸ $G_{\le t}$ not c[illegible]
$b = t$ $O(n+m)$
$G_{\le t}$

⟵ | ⟶
t $O(n+m)$

alg:
$t \leftarrow$ median weight of the edges
if $b < t$ then $O(n+m)$
$H \leftarrow G_{\le t}$
alg(H).
If $b = t$ then done
If $b > t$ then
collapse $G_{\le t}$
call alg(collapsed graph)



$T(m) = O(m) + T(\frac{m}{2}) \quad O(m) \quad O(m)$