8.11.2024

$U = \{x_1, \ldots, x_n\}$

$S_1 \subseteq U$

$S_2 \subseteq U$

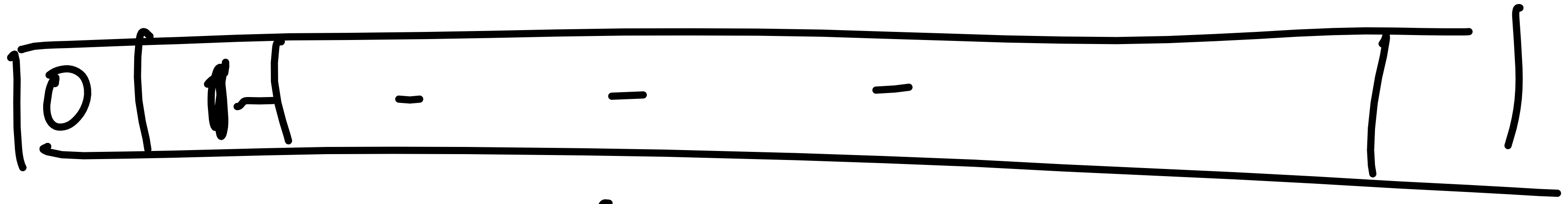
$\vdots$

$S_m \subseteq U$

Select as few subsets as possible that "covers U"

"

union of the chosen sets = U.



U

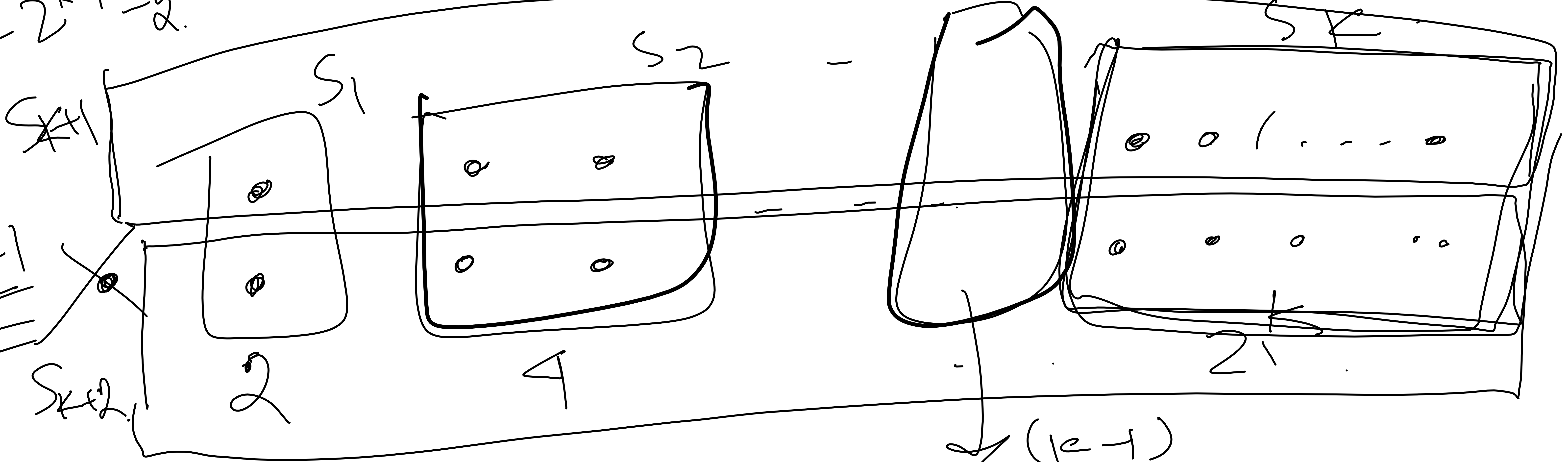
$O(mn)$

$k \log n$

- Is the greedy analysis best? $\Theta(\log n)$
- Is $\Omega(\log n)$ approximation unavoidable?

$|U| = n = 2^{k+1} - 2.$

$\frac{n}{2} = 2^k - 1$



OPT = 2.

S_k first

Greedy: • Selects the 2^k sized set in the first step

• S_{k-1} : 2^{k-1} additional elements

S_{k+1}/S_{k+2} $2^k - 1 - 2^{k-1}$

$= 2^{k-1} - 1$

Greedy selects K subsets $\quad K+1 = \log_2(n+2)$

best solution 2 subsets. $\quad K = \underline{\log_2(n+2) - 1}$

$\frac{1}{2}K = \frac{1}{2}\left[\log_2(n+2) - 1\right] = \Theta(\ln n)$

Very strong -ve result : $\underline{(P \neq NP)}$ under

No algorithm with $(1 - \underline{o(1)}) \log n$.

approximation ratio possible in polynomial

time : $\text{poly}(n, m) : n^3 m^2 \ldots$

$\frac{1}{\log n}$.

.99

.999

"Approximation algorithms"

2

Weighted Set Cover :

$U : \{x_1, \dots, x_n\}$

$S_1 \subseteq U$: $\underline{Cost(S_1)}$

⋮

$S_m \subseteq U$: $\underline{Cost(S_m)}$

trying to cover all element with minimum total cost

$Cost(\ ) \geq 0$

minimize $\underbrace{\dfrac{\text{Cost}(S_i)}{\#\text{ new elements } S_i \text{ is covering}}}_{\text{minimize this over } S_i}$

Claim: This algorithm has approx. ratio

$$H_n = 1 + \frac{1}{2} + \cdots + \frac{1}{n} \approx \log n$$

proof: "Charging arguement"

$C \leftarrow \phi.$

while $(C \neq U)$

• Select S_i with minimum $\dfrac{Cost(S_i)}{|S_i \setminus C|}$

all the newly added items will get this charge.

$C \leftarrow C \cup S_i$

Let : $\underline{e_1, \dots, e_n}$. is the order of insertion of the elements into C.

We'll define the charge due to e_i

$$\alpha_i \leftarrow \text{Charge of } e_i$$

Total cost of algorithm $= \sum_{i=1}^{n} \alpha_i = K\left(1 + \frac{1}{2} + \dots + \frac{1}{n}\right)$

$= \underline{\underline{K \cdot H_n}}$.

We want to show $\alpha_i \leq \underline{\underline{?}}$

Consider the situation before e_i was inserted.

at that point $|C| \le$ ~~i~~ $i-1$

$\Rightarrow$ remaining elements $\# \geq (n-i+1)$

e_i - - - — - e_n.

Suppose the best solution is K.

Pick out the j sets* that covers

$e_i, \dots, e_n$ with cost $\le K$.

therefore the minimum charge in the selected sets * $\leq \dfrac{K}{(n-i+1)}$

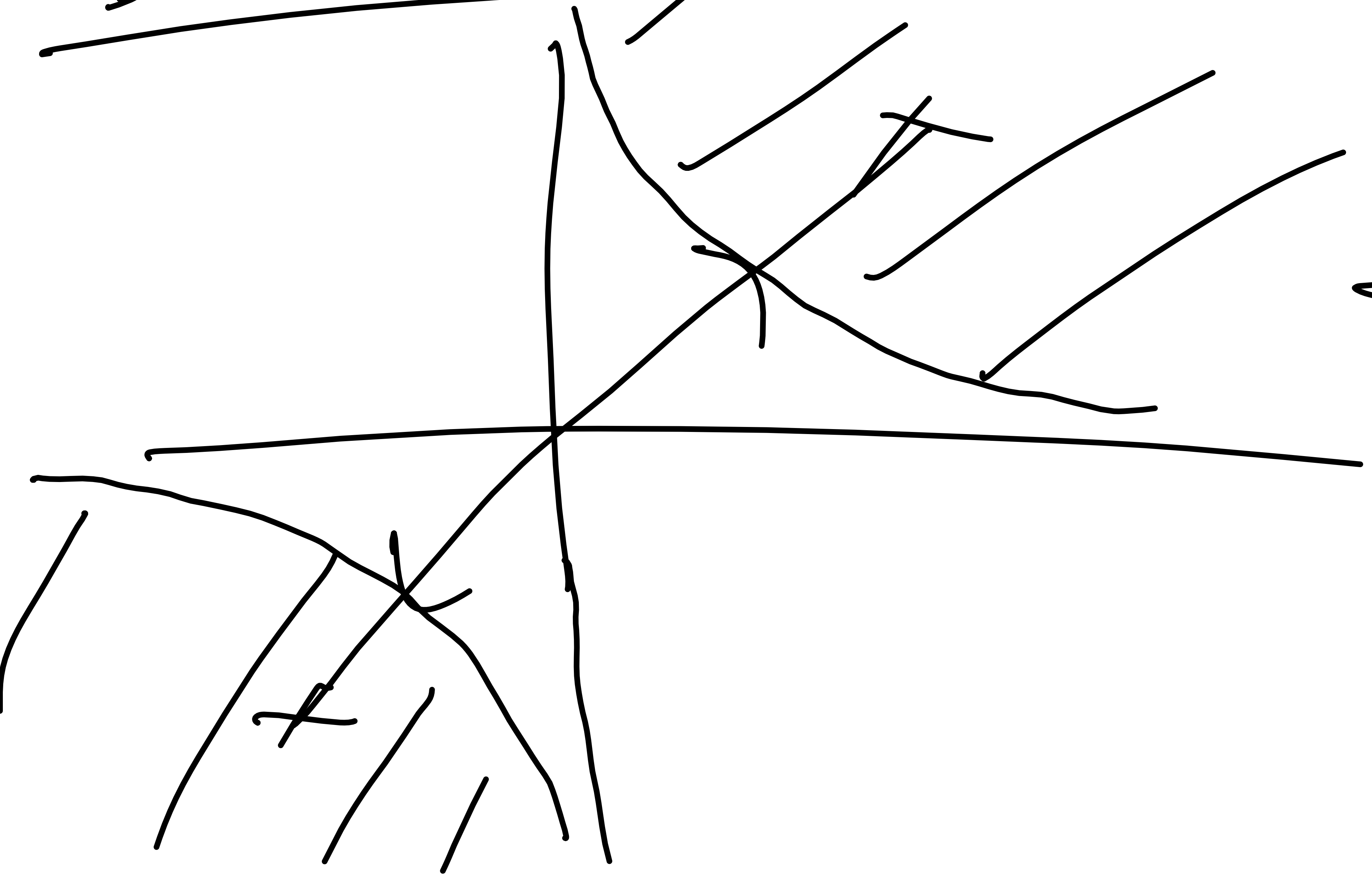
$$\alpha_i \leq \frac{K}{(n-i+1)}. \qquad \square$$

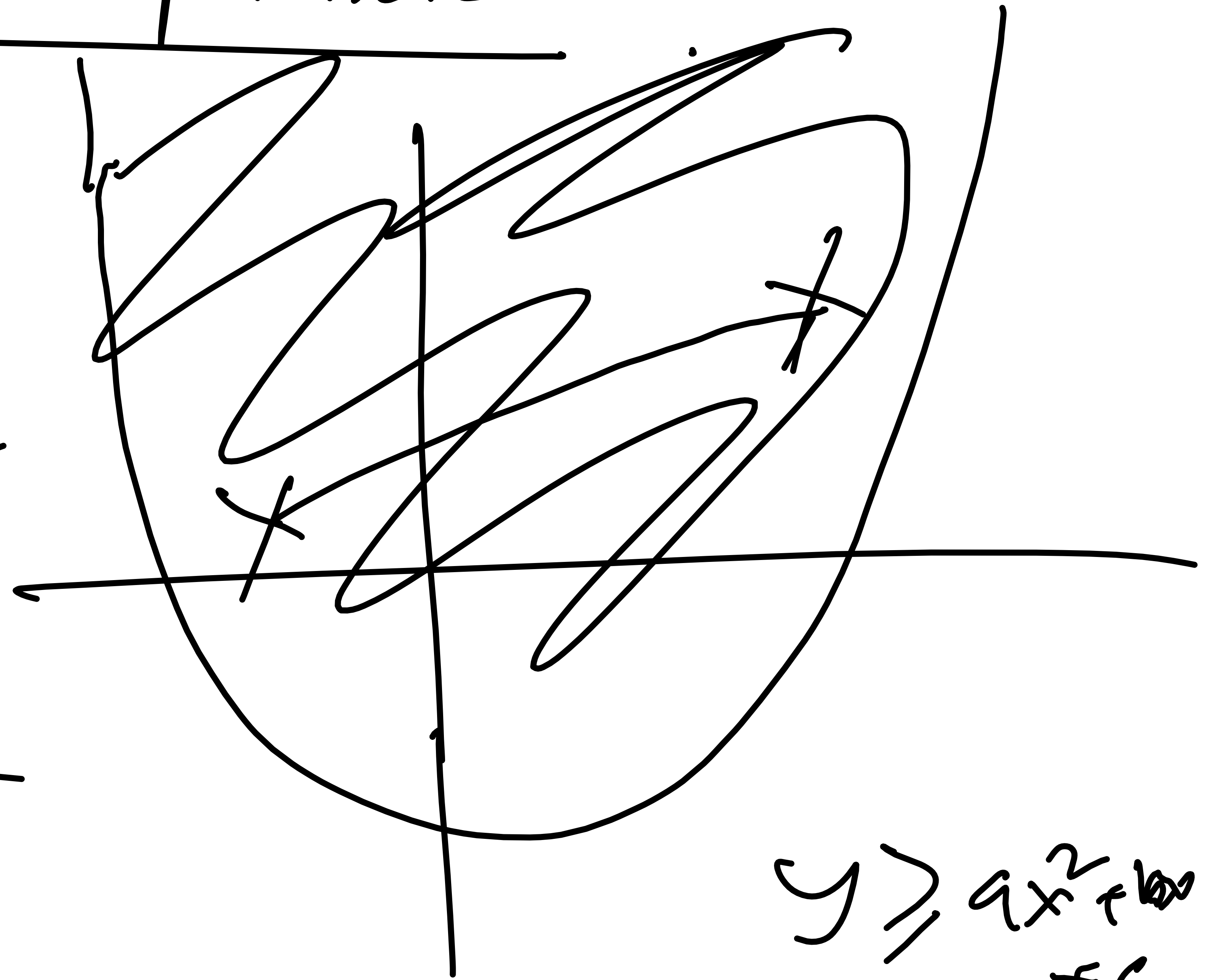
$\Rightarrow$ approx. ration is $\leq \cancel{K\log n}\,.\,K.H_n.$

~~Conv~~ Computational Geometry

Convex Hull Computation

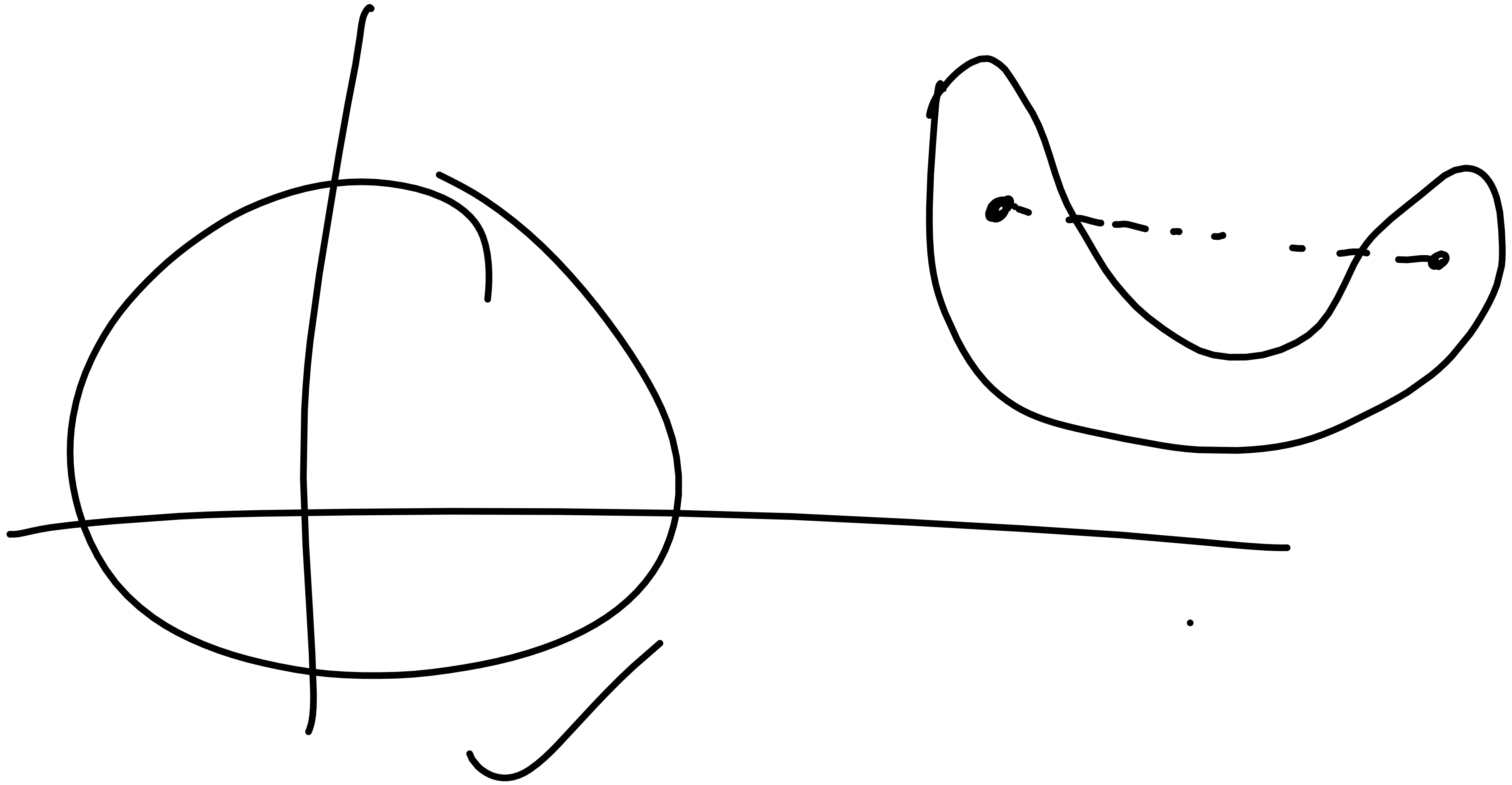
Define ~~Convex~~ Set :





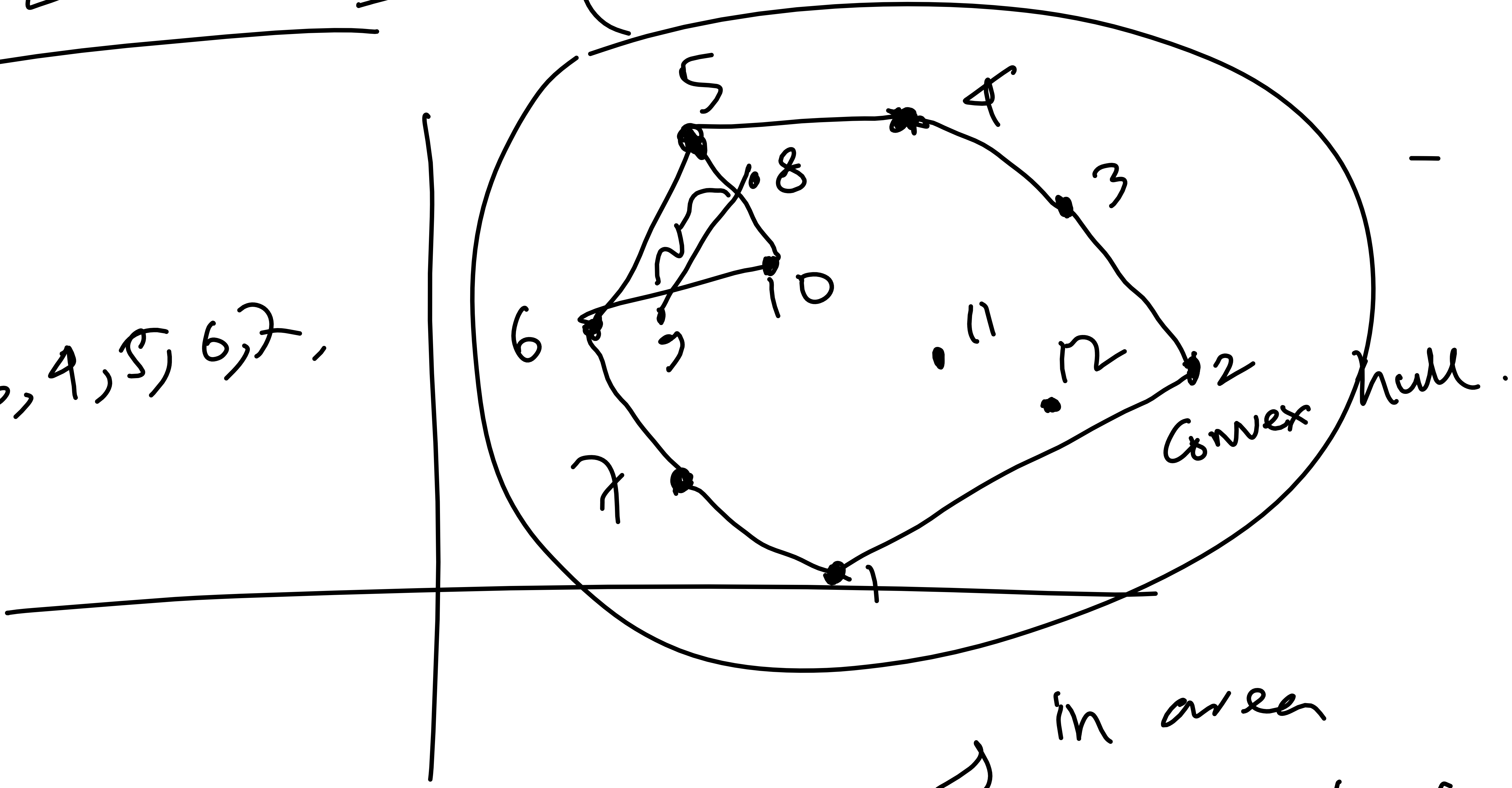
$y \geq ax^2 + bx + c$

(informal) A set is called a convex set if for any two chosen points (x, y) the line going through x, y lies with in the set.



Def^n [Convex Hull] : (we'll restrict ourselves to $\mathbb{R}_2$).

1, 2, 3, 4, 5, 6, 7,



smallest convex set (in area) that includes all the points.

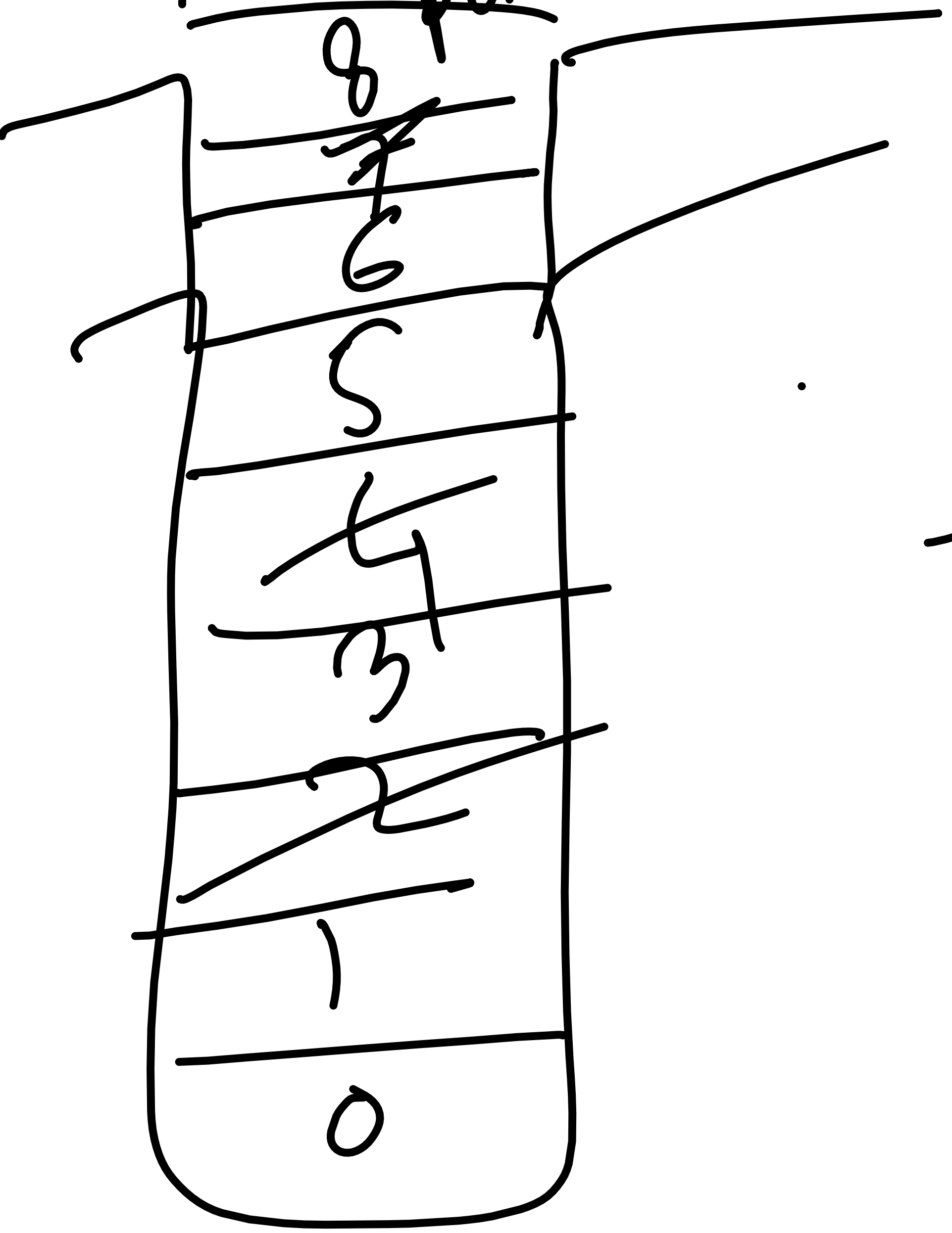
Input: A set of points in $\mathbb{R}^2$

$$(x_1, \ldots, x_n) = P$$

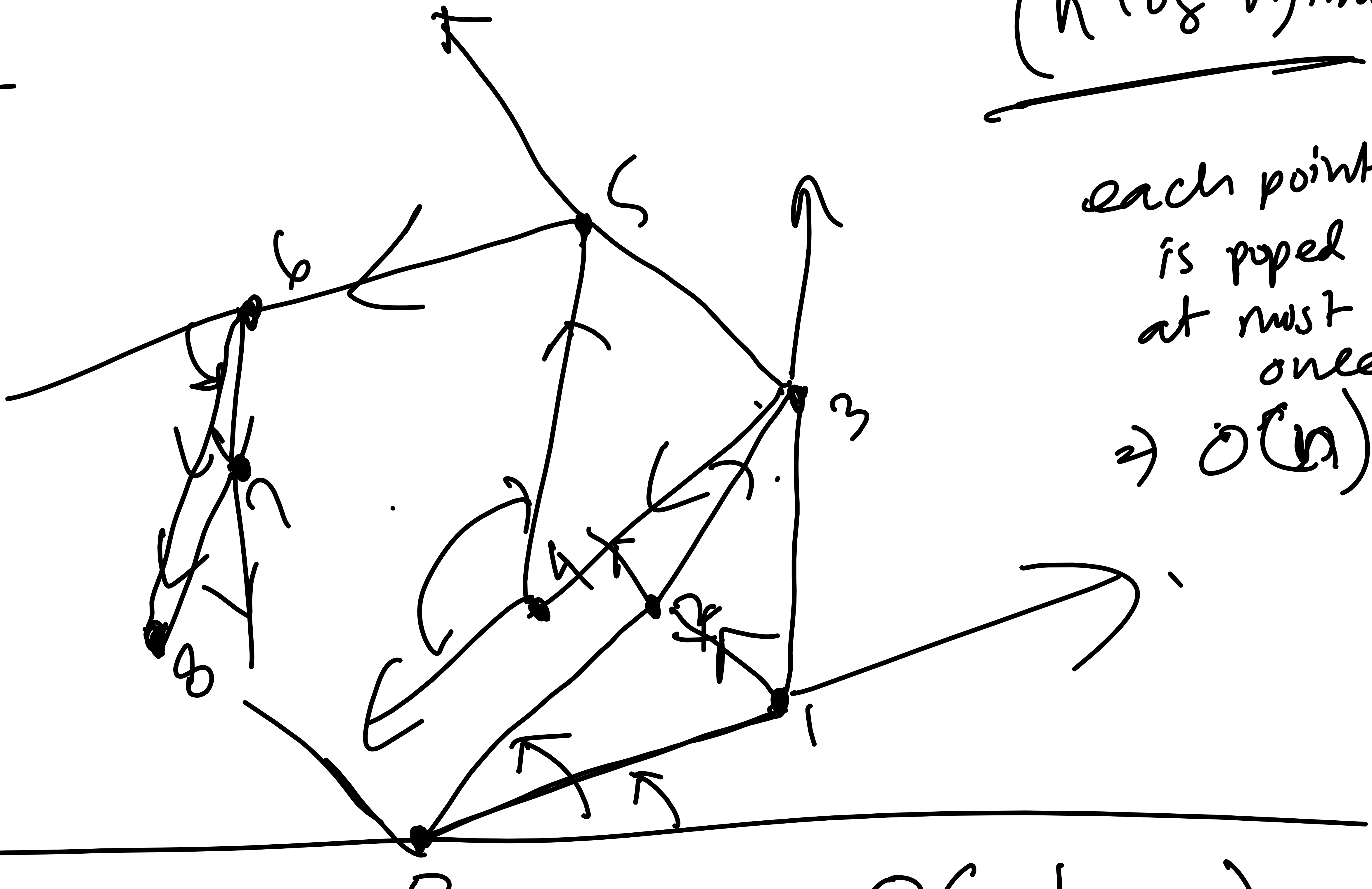
Output: Convex Hull of P.

(return sorted order of boundary points either clockwise or anti clockwise).

∄ no collinear points.



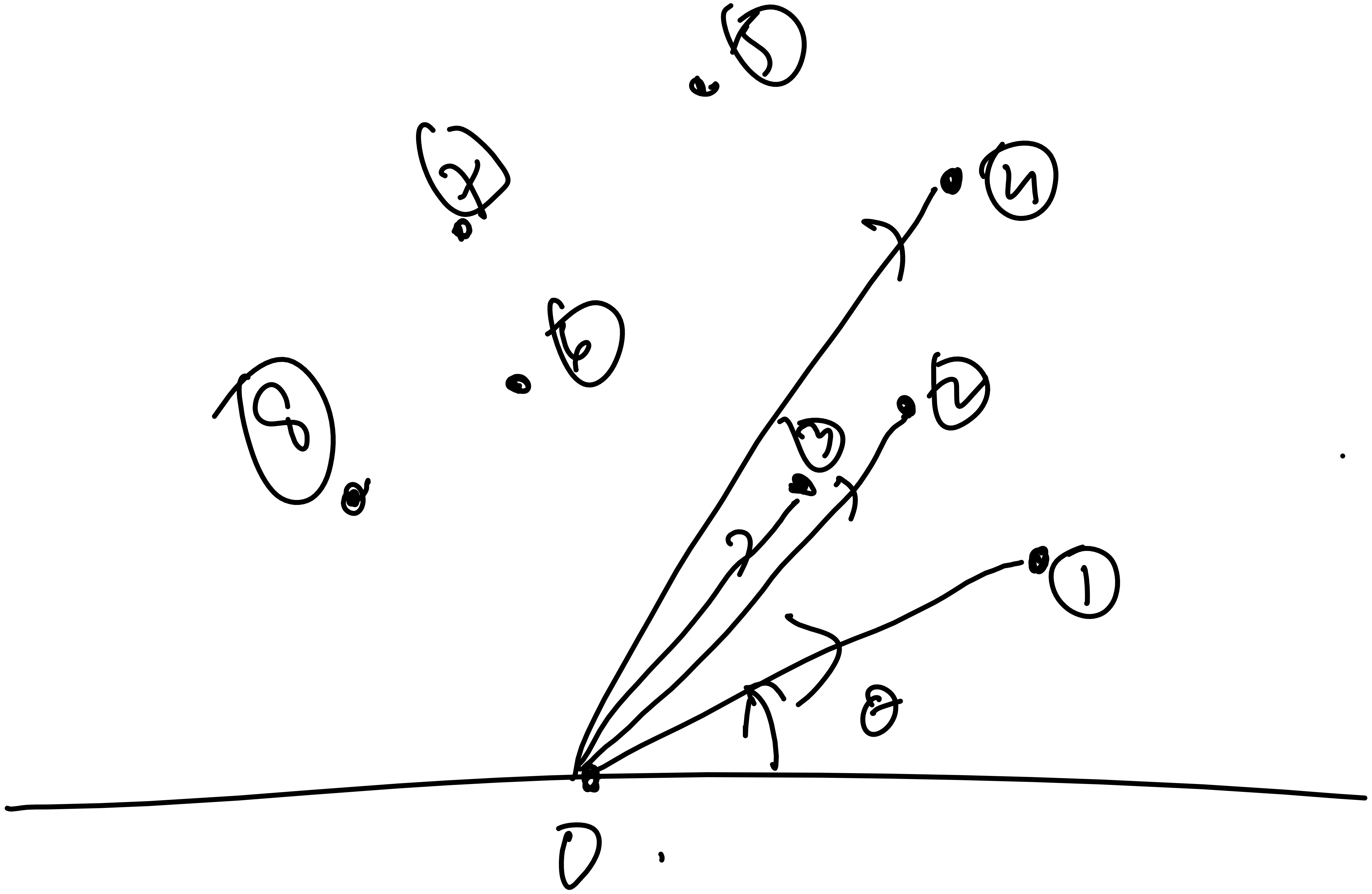
(n log n) time

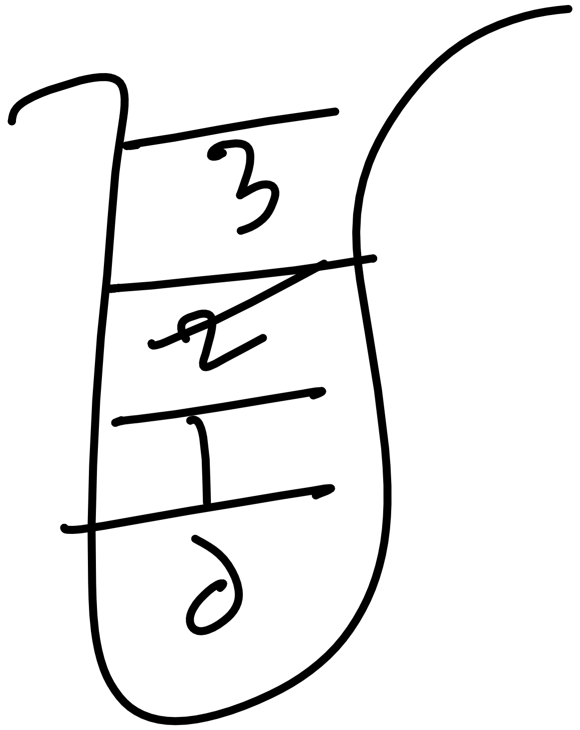


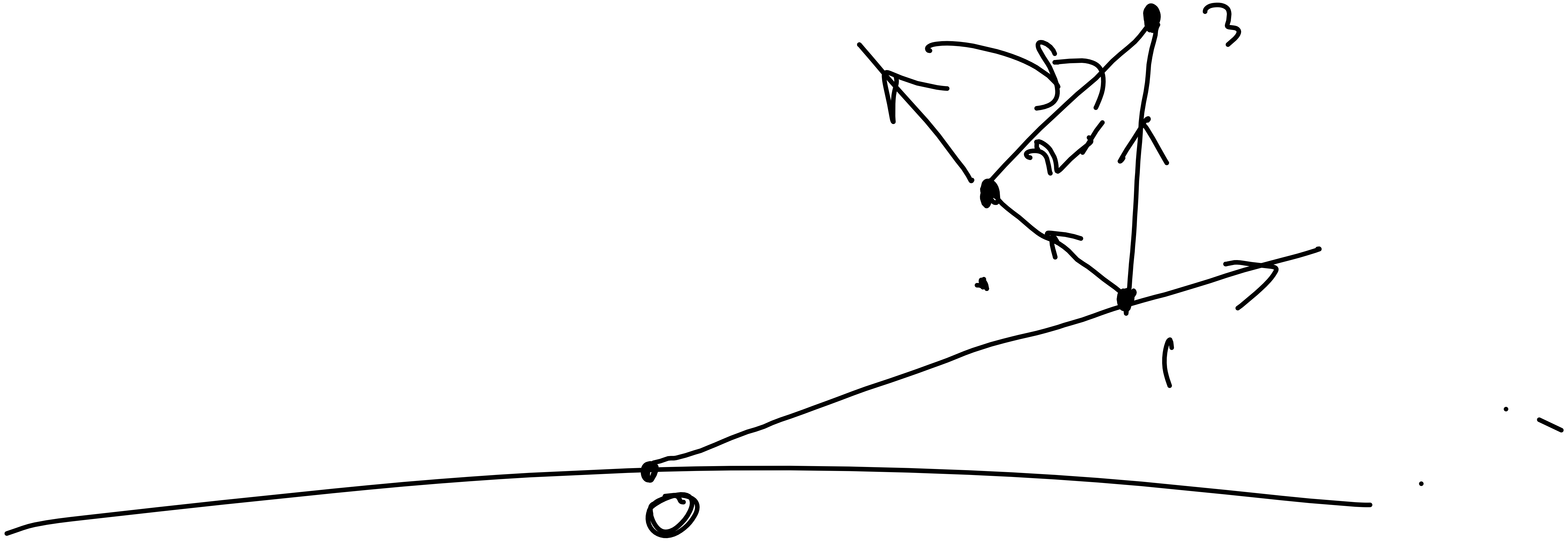
each point is poped at most once

$\Rightarrow O(n)$

total time complexity : $O(n \log n)$


1
2
3
4
5
6
7
8
θ
O


3
2
1
0


3
2
1
0