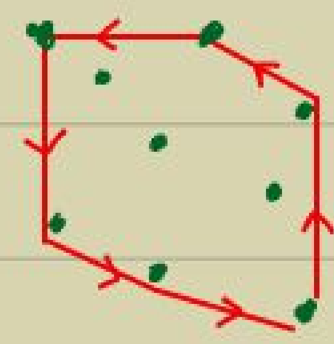
Convex Hull problem

CH(S):

Input: Given n points $S \subseteq \mathbb{R}^2$.

Output: Convex polygon of smallest area enclosing S.

(Think of a rubber band enclosing a set of pins!)



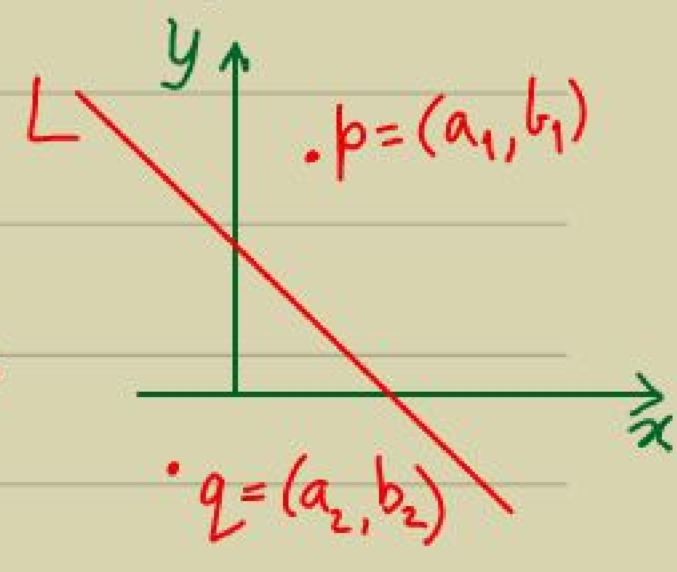
- Brute-force algorithm:

  Find edges such that all the other points in S lie on "one side" (say, left).

  Naively, it takes $O(n^3)$ time.

- Can we do better? (Exploit geometry!)

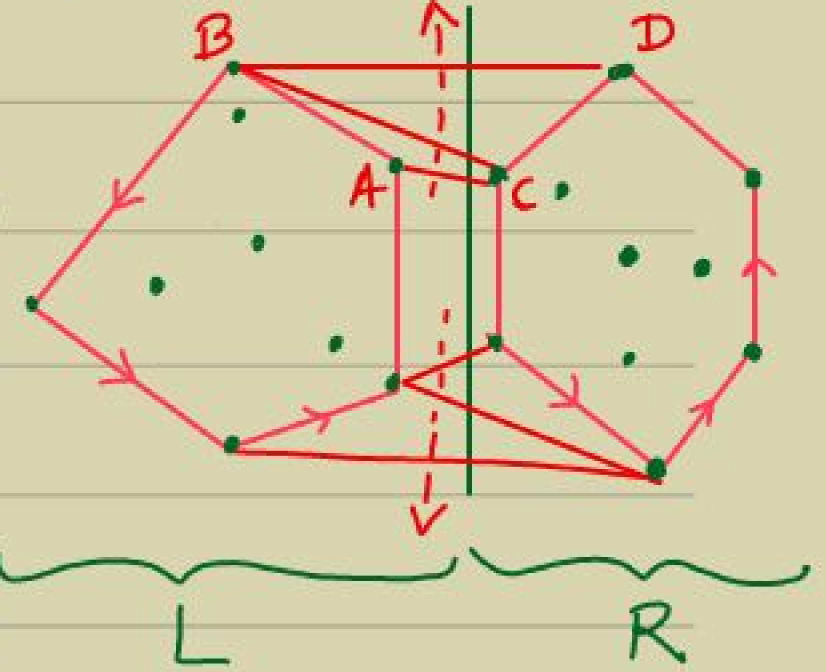
- Qn: Given a line $L: y = mx + c$ & points p, q how do you test whether they are on the same side of L?



▷ On the upper side of L $y > mx + c$ & on the lower side $y < mx + c$.

⇒ In $O(1)$ time we can test whether p, q are on the same side of L.

- Use this & Divide-Conquer to find $CH(S)$.



• Divide: Partition S into L & R by the x-median.

• Solve: $CH(L)$ & $CH(R)$. Assume that the hull has vertices in the anticlockwise direction.

• Combine: Go over $CH(L)$, $CH(R)$ & do a new merge-like process to find the extremal edges.

More pointers in $CH(L)$ resp. $CH(R)$

Eg. for the cross-edge AC check whether B is above. If YES then pick BC.

For BC find the neighbour that is above (say, D). Pick BD.

Each decision involves only 6 vertices

As BD has no neighbour above, it

becomes an extremal edge.

Analogously, find the second extremal edge.

Exercise: Proof of correctness.

▷ The recurrence for time $T(n)$ is:

$$T(n) = 2T(n/2) + O(n).$$

$$\Rightarrow \quad T(n) = O(n \lg n).$$

Theorem: $CH(S)$ is computable in $O(|S| \lg |S|)$ time.

[Ignoring integer sizes.]

- Write the detailed pseudocode.

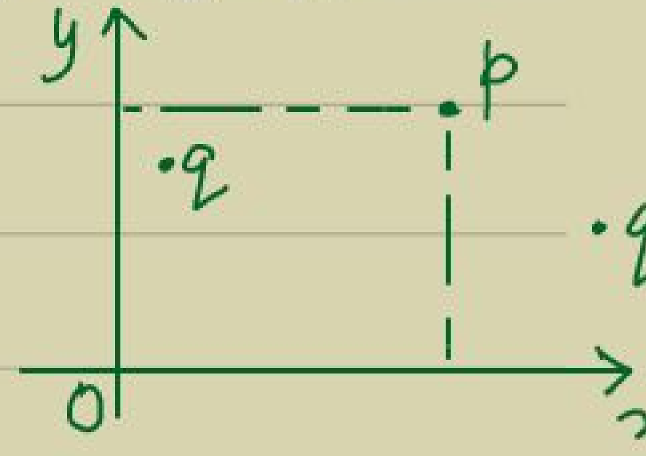
- Can the time be improved?

- The algorithm is by Preparata & Hong (1977).

→ CH is a basic algorithm in the area of computational geometry.

Non-dominated points problem

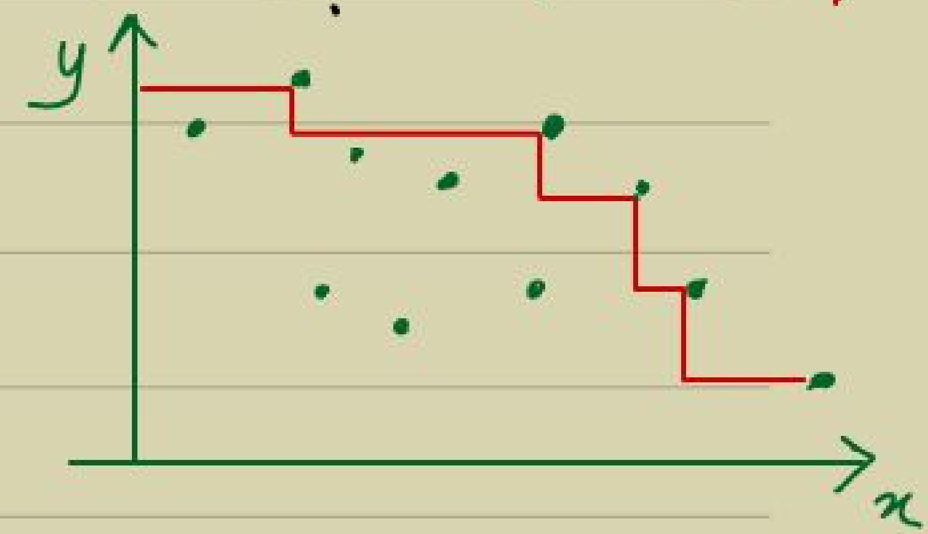
- **Defn**: A point p dominates q if $x(p) > x(q)$ & $y(p) > y(q)$.



Input: Given n points $S \subset \mathbb{R}^2$.
Output: Points p in S that are not dominated by any point in S. (in a way extremal)

- Brute-force algorithm:
  Go over every $p \in S$ & compare with each $q \in S$.
  Takes $O(|S|^2)$ time.
- It's used in comp. geometry, game theory & databases.
- Can geometry help?
- What is the structure of non-dominated points?

- They form a staircase! (Exercise)



- Thus, these are extremal points in a sense.

▷ A point with max. x-coordinate is a non-dominated point.

- <u>Idea 1</u>: Among the points in S with the max. x-coordinate, pick the point p with max. y-coordinate.
    - Declare p non-dominated.
    - Delete all the points $q \in S$ with $y(q) < y(p)$.
    - Repeat till $S \neq \phi$.

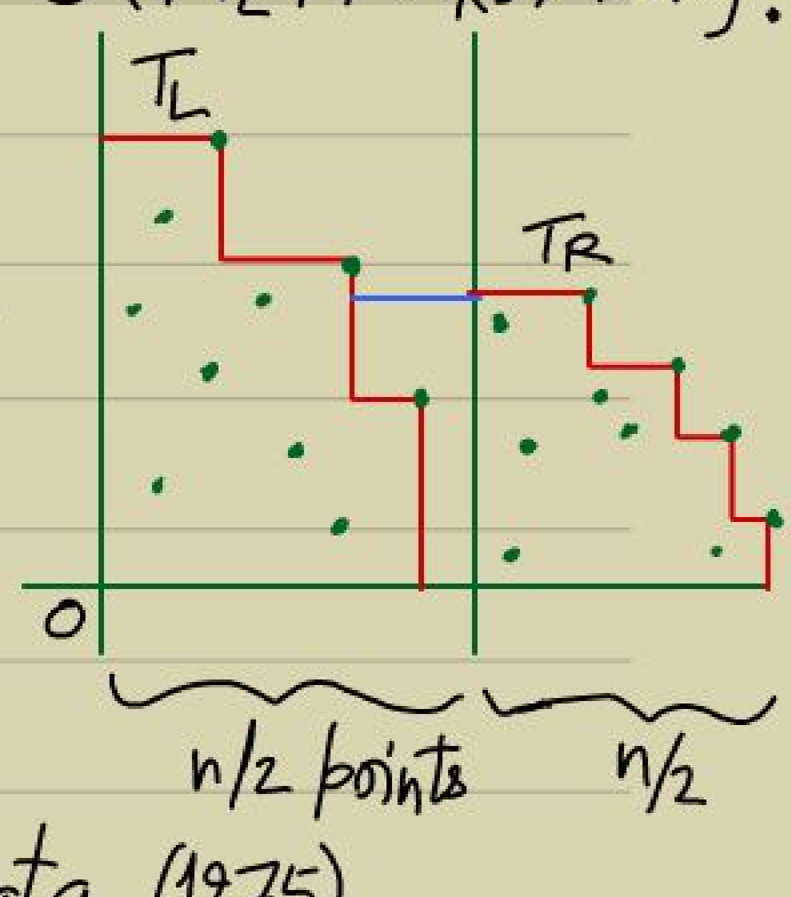
▷ If h = # non-dominated points in S, then the time complexity is $O(nh)$.

output-sensitive algorithm

- *Idea 2*: Divide S into two halves. Solve each & Merge the two staircases.
  - Let the two staircases be T_L & T_R.
  - During the merge step scan T_R to find the max-y-coordinate point $p \in T_R$.

    Scan T_L & delete those points q s.t. $y(q) < y(p)$.

    Take the union of T_L & T_R that remain.
  - Merge steps are $O(|T_L| + |T_R|)$ many.



Theorem: Non-dominated points in S can be computed in $O(|S| \lg |S|)$ time.

- by Kung, Luccio, Preparata (1975).