Divide & Conquer

- Paradigm
- Convex Hull
- Median finding

Paradigm

Given a problem of size $n$

Divide it into $a$ subproblems of size $\frac{n}{b}$

where $a > 1$, $b > 1$

Solve each subproblem recursively

Combine solutions of subproblems to get overall solution

$$T(n) = a \cdot T\left(\frac{n}{b}\right) + \left[\text{work for merge}\right]$$
Convex Hull

Given $n$ points in plane [Ref § 33.3]

$$S = \{ (x_i, y_i) \mid i = 1, 2, \ldots, n \}$$

Assume no two have same $x$ coord, no two have same $y$ coord, and no three in a line for convenience.

(Convex Hull: smallest polygon containing all points in $S$)

If points are nails, then $CH(S)$ is shape of rubber band around all the nails.

$CH(S)$ represented by the sequence of points on the boundary in order clockwise.

$$p \leftrightarrow q \leftrightarrow r \leftrightarrow s \leftrightarrow t$$
Brute force for Convex Hull

$n$ points

Test each line segment to see if it makes up an edge of the convex hull:

- If the rest of the points are on one side of the segment, the segment is on the convex hull — above

- else the segment is not — above

$O(n^2)$ edges, $O(n)$ tests $\Rightarrow O(n^3)$ complexity

Can we do better?
D&C for Convex Hull

Sort points by x coord (once & for all, O(n log n))
For input set S of points:
- Divide into left-half A & right half B
  - by x coords
- Compute CH(A) & CH(B)
- Combine CH's of two halves (merge step)

**HOW TO MERGE?**

A

\[ \begin{array}{c}
  a_4 \\
  a_5 \\
  a_1 \\
  a_2 \\
  a_3 \\
\end{array}\] 

B

\[ \begin{array}{c}
  b_2 \\
  b_1 \\
  b_3 \\
\end{array}\]

Find upper tangent \((ai, bj)\)
Find lower tangent \((ak, bm)\)

Cut & paste in time \(\Theta(n)\)
First Link \(ai \to bj\), go down b list till you see \(bm\) and link \(bm\) to \(ak\)
Continue along the a list until you return to \(ai\)
FINDING TANGENTS

Assume $a_i$ maximizes $x$ within $CH(A) (a_1, a_2, \ldots, a_p)$

$b_j$ minimizes $x$ within $CH(B) (b_1, b_2, \ldots, b_q)$

$L$ is the vertical line separating $A$ & $B$

Define $y(i,j)$ as $y$-coordinate of pt of intersection

between $L$ & segment $(a_i, b_j)$

CLAIM: $(a_i, b_j)$ is upper tangent iff it maximizes $y(i,j)$

If $y(i,j)$ is not maximum, there will be points on

both sides of $(a_i, b_j)$ and it can't be a tangent.

Algorithm: Obvious $O(n^2)$ algorithm looks at all

$a_i, b_j$ pairs $T(n) = 2T(n/2) + \Theta(n^2)

= \Theta(n^2)$

\[
\begin{align*}
\text{\#} & \quad i = 1 \\
\text{\#} & \quad j = 1 \\
\text{while} & \quad (y(i, j+1) > y(i, j) \quad \text{or} \quad y(i-1, j) > y(i, j)): \\
\text{if} & \quad y(i, j+1) > y(i, j): \quad \text{move right finger} \\
& \quad j = j+1 \quad \text{(mod q)} \\
\text{else} & \quad i = i-1 \quad \text{(mod p)} \quad \text{move left finger} \\
\text{return} & \quad (a_i, b_j) \quad \text{as upper tangent} \\
\text{Similarly for lower tangent} \\
T(n) = 2T(n/2) + \Theta(n) \quad \text{Master Theorem gives } \Theta(n \log n)
\end{align*}
\]
Intuition for why Merge works

$\text{a}_1, b_1$ are right most & leftmost points.

We move anti-clockwise from $\text{a}_1$, clockwise from $b_1$.

As, $\ldots \text{aq}$ is a convex hull, as is $b_1, b_2, \ldots b_q$.

If $\text{ai}, b_j$ is such that moving from either $\text{ai}$ or $b_j$ decreases $y(i,j)$ there are no points above the $(\text{ai}, b_j)$ line.

The formal proof is quite involved and won't be covered.
Median Finding

[Ref: § 9.3]

given set of n numbers, define rank(x) as
number of numbers in the set that are ≤ x

Find element of rank \( \left\lfloor \frac{n+1}{2} \right\rfloor \): lower median
(or element of rank i)

\( \left\lceil \frac{n+1}{2} \right\rceil \): upper median

Clearly sorting works in time \( \Theta(n \log n) \)

Can we do better?

Select \((S, i)\)

- Pick \( x \in S \) (cleverly)
- Compute \( k = \text{rank}(x) \)

\( B = \{ y \in S \mid y < x \} \)
\( C = \{ y \in S \mid y > x \} \)

\[ \begin{array}{c}
  \leftarrow B \\
  x \\
  \rightarrow C
\end{array} \]

- if \( k = i \): return \( x \)
- else if \( k > i \): return Select(\( B_i \))
- else if \( k < i \): return Select(\( C, i-k \))
PICKING X CLEVERLY

Need to pick $x$ so $\text{rank}(x)$ is not extreme.

- Arrange $S$ into columns of size 5 ($\frac{n}{5}$ cols)
- Sort each column (big elements on top) (linear time)
- Find "median of medians" as $x$

How many elements are guaranteed to be $> x$?

Half of the $\frac{n}{5}$ groups contribute at least 3 elements $> x$ except for 1 group with less than 5 elements & 1 group that contains $x$

At least $3(\frac{n}{10}7 - 2)$ elements are $> x$

Recurrence: $T(n) = \begin{cases} 
O(1) & \text{for } n \leq 140 \\
T\left(\frac{\lfloor n/5 \rfloor}{7}\right) + T\left(\frac{7n}{10} + 6\right) + \Theta(n) & \text{sorting each column}
\end{cases}$
Solving the Recurrence

Master theorem does not apply

Prove $T(n) \leq c \cdot n$ by induction, for some large enough $c$

- True for $n \leq 140$ by choosing large $c$
- $T(n) \leq c \cdot \lceil \frac{n}{5} \rceil + c \cdot \left( \frac{7n + 6}{10} \right) + a \cdot n$
  - ($a$ needs to be large enough to cover $O(n)$ term)

\[
\leq \frac{cn}{5} + c + \frac{7nc}{10} + 6c + an
\]

\[
= cn + c + \left( \frac{-cn + 7c + an}{10} \right)
\]

If this is $\leq 0$, we are done

$C \geq \frac{70c + 10a}{n}$

OK for $n \geq 140$ & $C \geq 20a$
Example

\[ a_3, b_1 \text{ is upper tangent} \]
\[ a_4 > a_3 \]
\[ b_2 > b_1 \]
\[ a_1, b_3 \text{ is lower tangent} \]
\[ a_2 < a_1 \]
\[ b_4 < b_3 \]

\[ a_i, b_j \text{ is an upper tangent. Does not mean that } a_i \text{ or } b_j \text{ is the highest point} \]