The Convex Hull Prob
(Sorting Prob of CG)

Def \( A \subseteq \mathbb{R}^d \) convex if closed under convex combinations.

Def \( \text{ConvexClosure}(A) = \text{CC}(A) = \text{smallest convex set} \supseteq A \)

2 Defs of Convex Hull

Def 1 \( \text{CH}(A) = \bigcup \text{CC}(A) \)

Def 2 \( \text{CH}(A) = \text{CC}(A) \)

We will use Def 1

A finite set

Thus in 2D \( \text{CH}(A) \) is a simple closed polygon.
(say CCW)
We will use following characterization

\[ \text{Claim} \quad [a, b] \text{ is on } \text{CH}(A) \iff \quad a \neq b \]
1) \(a, b \in A\)
2) \(\forall a' \in A \text{ either } a' \text{ left of } [a, b]\)
   \(a' \in [a, b]\)

---

2D Convex Hull by divide-and-conquer

\[ A = \{P_1, \ldots, P_n\} \quad P_i = (x_i, y_i) \]

Preprocess: sort \(A\) by \(x\)-coordinate

2D-CH(A)

\[ \text{if } |A| = 1 \text{ return } P_i \]

\[ \text{else } \quad \text{CH}_L = \text{2D-CH}(P_1, \ldots, P_{n_1}) \]
\[ \text{CH}_R = \text{2D-CH}(P_{n_2}, \ldots, P_n) \]

\[ \text{STITCH}(\text{CH}_L, \text{CH}_R) \]
STITCH (L, R)

Lower bridge (L, R)
    a = rightmost (L)
    b = leftmost (R)

(Repeat x) **)
** x) While a Right (a, b) set a = a
**x) While b Right (a, b) set b = b

Upper bridge (L, R) = ?

---

Correctness
** x) generates triangles (a, b, a)
**x) " " " (a, b, b)

1) The Δ's are disjoint
   They are ordered by their intersection with vertical line L.
2) They are in CC(A).

Thus termination:
At termination a, a, b, b are all lift of (a, b)
Since $(9, 9), (9, a), (b, b), (b, e)$ are on $CH(L) \& CH(R)$ respectively.

Done

---

Timing: Preprocess $O(n \log n)$ to sort:

STITCH in $O(n)$

$T(n) = 2T(n/2) + cn$

$\therefore T(n) = O(n \log n)$
Lower bounds

Sorting reducible to CH

Input: \( x_1, \ldots, x_n \)

\( CH((x_1,x_1^2), \ldots, (x_n,x_n^2)) \)

The CH will be \( x_i \)'s in sorted order.

An important use for CH

\( p_1, \ldots, p_n \in \mathbb{R}^2 \)

\( \bar{p}_i = (p_x, p_y, p_x^2 + p_y^2) \)

\( CH(\bar{p}_1, \ldots, \bar{p}_n) \) = Triangulated surface

The Delaunay Triangulation
Quick Sort & Backwards Analysis

Consider
\( QS(M) \) (distinct keys)
1) Pick random \( a \in M \)
2) Split \( M : \ 5 < a < 7 \) (\( |M| - 1 \) comparisons)
3) Return \( QS(S) \times a \times QS(L) \)

Goal: Expect \# comparisons

Consider dart game:

Init: empty board

While non-empty square
pick random empty sq
Cost = \# empty sqs to left & right of dart.

Claim: Expect cost of dart game = Expect cost QS.
Backwards game:

Init: full board

While I dart, remove random dart.
Cost: # empty E's left & right.

Claim: \( \text{Expected cost} \ DG = \text{Expected cost} \ BW \ DG \)

Analysis backwards game

Assume i darts on board

\( T_i = \) Expected cost to remove random dart.

Total Cost = \( \sum_{i} \) cost of i

\[ E(DG) = \sum T_i \leq \sum \frac{2(n-i)}{i} \leq \frac{2n}{i} \]

\[ = 0/n, n \]
Random Incremental CH

Procedure Random Incremental CH (P)

0) Make \( \Delta = (p_1, p_2, p_3) \) pick C \( \in \) interior \( \Delta \)

1) Construct ray from C to each \( p_i \)

2) Partition \( p_i \) by edge of \( \Delta \) they cross.

3) Randomly permute \( p_0, \ldots, p_n \).

For \( i = 4 \) to \( n \)

let \( e \) be edge crossed by ray \( C \rightarrow p_i \)

BuildTent (\( p_i, e \))

Procedure BuildTent (\( p, e \))

1) Find edges of CH "visible" to \( p \) by searching out from \( e \)

2) Replace visible edges with 2 new edge

3) Assign rays to the new edges.
Correctness?

Timing

$O(n)$ work other than BuildTest.

Consider steps 1 & 2 in BuildTest.

1) At most an edge
2) Charging rule for line-side tests
   a) Not visible tests: we charge $p_i$ each visible test: we charge to the edge
   
   Total $2n + 2n$ on $4n$ tests.

Consider step 3 in BuildTest:

Ray-costs

Backward analysis

Let points to pick from say $p_j$

$\text{Cost}(p_j) = \begin{cases} 0 & \text{if } p_j \text{ not on hull} \\ \# \text{ray crossing to left & right} & \text{otherwise} \end{cases}$
\[ C_i = \text{cost} \]

\[ E(C_i) \leq \frac{2(n-i)}{i-3} \]

\[ C = \text{total cost} \]

\[ E(C) = \sum_{i=4}^{n} E(C_i) \leq \sum_{i=4}^{n} \frac{2(n-i)}{i-3} \leq 2n \sum_{i=1}^{n} \frac{1}{i} \]