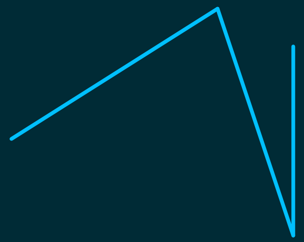
1. Suppose that in Graham's scan, pairs of points (not three or more) may have the same x-coordinates. How can you modify the algorithm to handle this degeneracy?

While computing the upper hull, change the sorting order to:

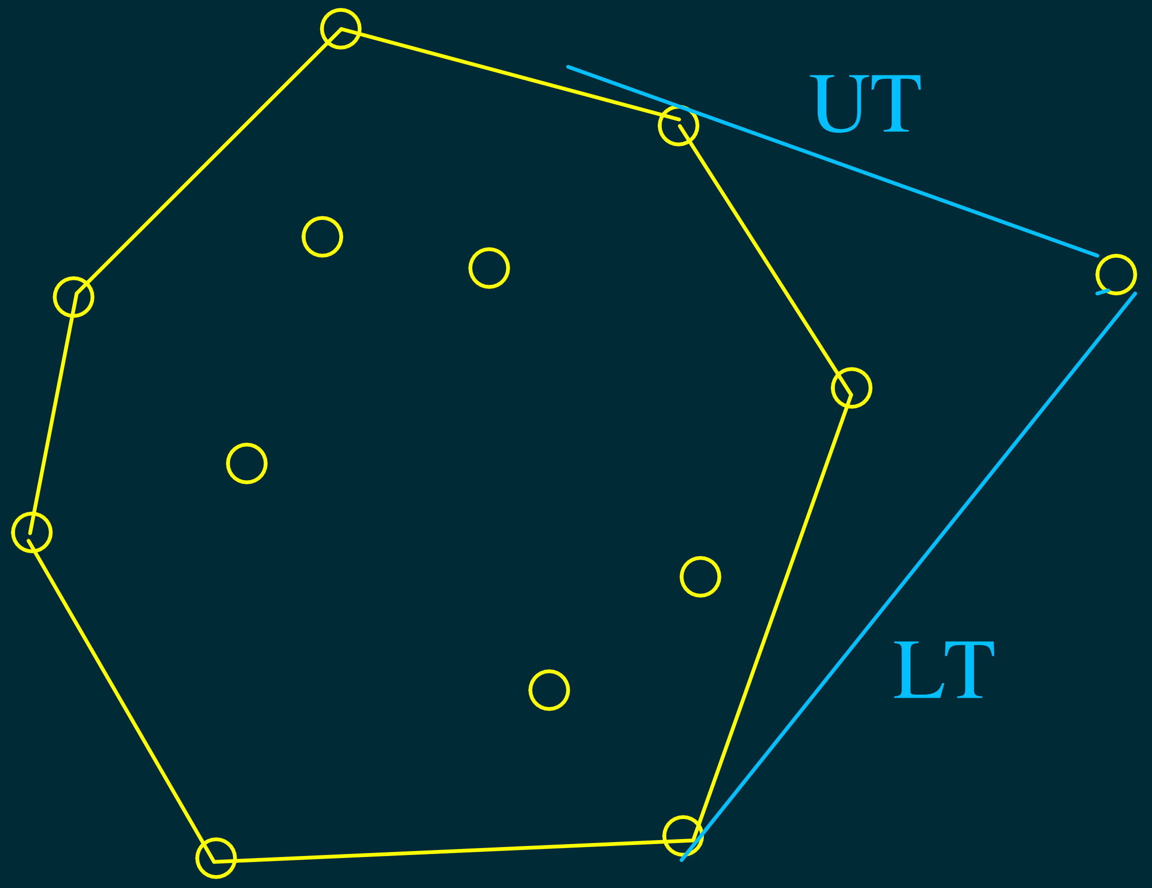
1. First in increasing order of x-coordinate.
2. Second (in case of ties) with respect to decreasing order of y-coordinates.

Right turn



Wrong turn

2. Suppose that the points are coming in increasing x-coordinates. How can you convert a hull of n points to a hull of n+1 points? Running time?



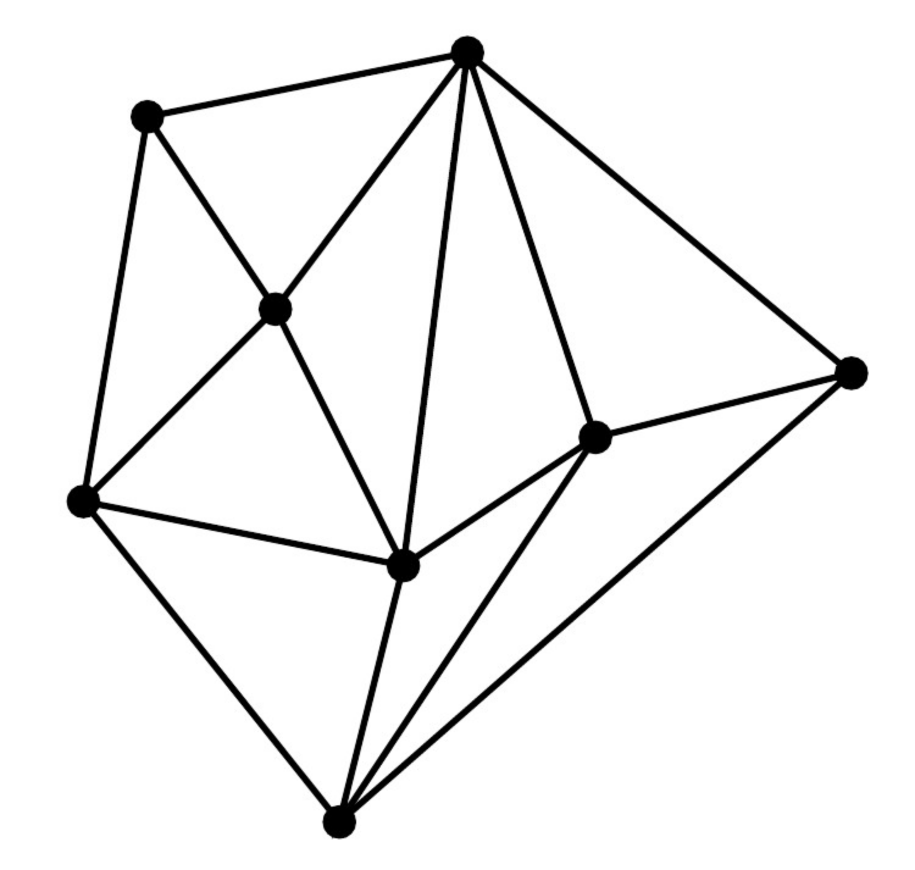
Compute the upper and the lower tangents starting two marches from the rightmost point. Discard the points between the points of tangency.

O(n) because these many vertices may be deleted.

3. [Incremental Convex-Hull construction] In the construction of CH(S), we first sort the points in increasing x-coordinates. Then we apply the previous construction by introducing one point at a time. What will be the overall running time?

The initial sorting takes O(n log n).
The remaining construction takes O(n) amortized time.

4. (a) Prove that the number e of edges in any triangulation of CH(S) with |S| = n satisfies $2n - 3 \leq e \leq 3n - 6$.



Let f be the number of triangles, and h the number of edges of CH(S).

n - e + (f + 1) = 2 [Euler's fomula]
3f + h = 2e.
Eliminate f.
Finally use 3 <= h <= n.

(b) How can you use the incremental convex-hull construction to triangulate CH(S)? Running time?

Join the new point with
(1) the points of tangency, and
(2) all points discarded.

O(n) extra effort to produce the triangulation by Part (a).

5. Suppose that in the line sweep algorithm for the line-segment intersection problem, some lines are allowed to be vertical. Explain how you can handle a "vertical segment" event.

$O((h+n)\log n)$ time $\quad O(n)$ space

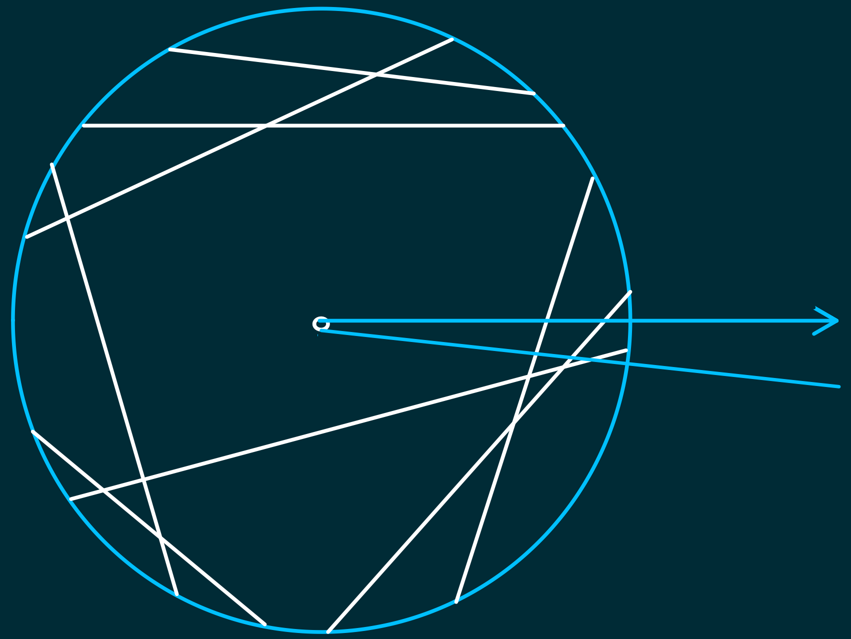
Given a BST T and two values x and y.

Locate all keys k in the BST st $x < k < y$.

Or keep on finding successors

$O(\underbrace{h(T)}_{O(\log n)} + \#\text{ of keys})$

6. You are given a set of chords in a circle. Propose an algorithm to find which chords are visible fully or partially from the center.



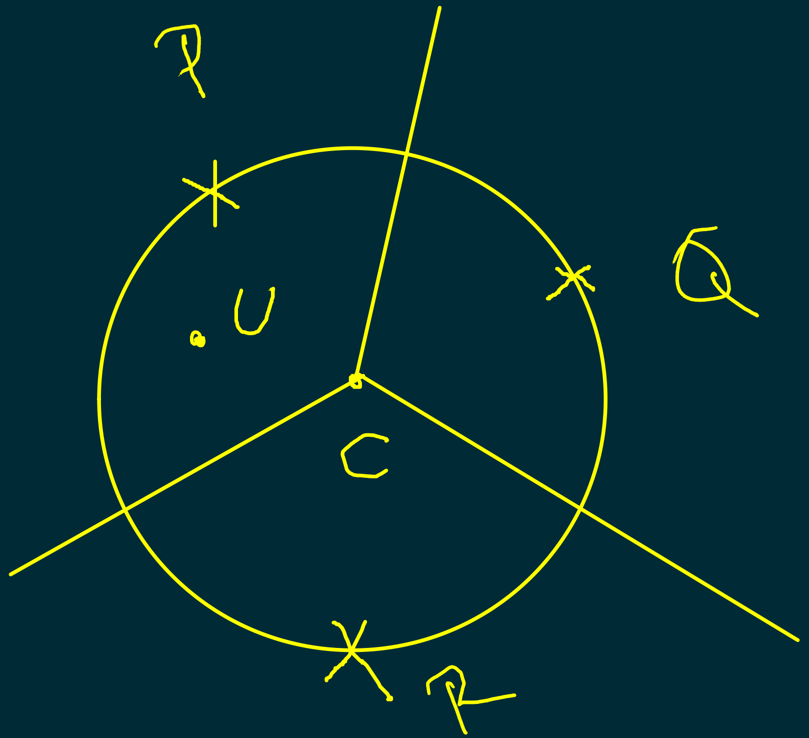
Ray sweep algorithm
Events: Enter chord, leave chord, chord intersection.

Sweep-ray information: Active chords (from closest to farthest from center). Maintain as balanced BST.

Event information: Also a balanced BST, because arbitrary deletion is needed. We maintain intersection events for consecutive chords only.

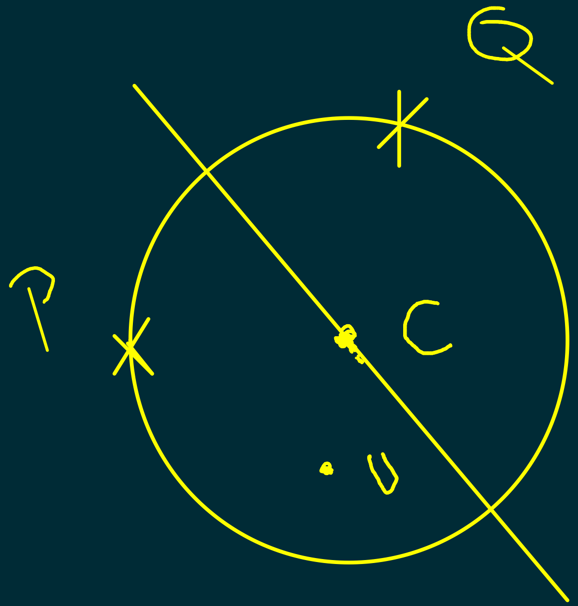
7. Let S be a set of n sites. Prove the following facts about Vor(S).

(a) Let C be a vertex of Vor(S). Then C is equidistant from three sites P,Q,R. Prove that the circle with center at C and passing through P,Q,R contains no other sites.



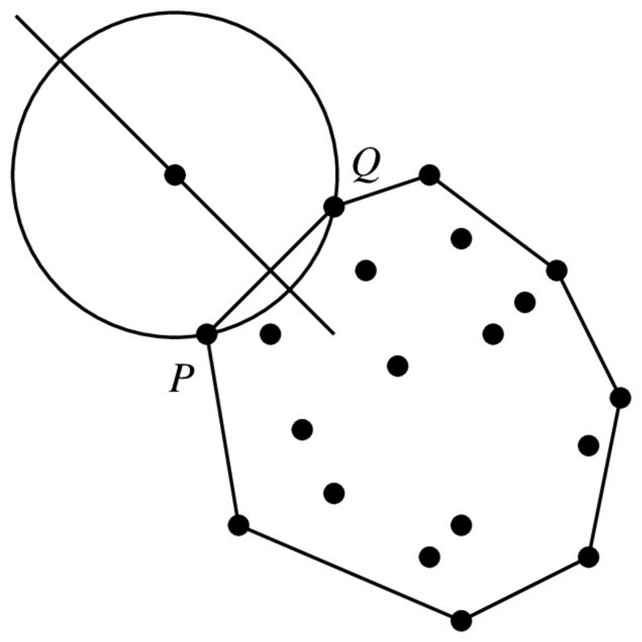
If a site U is inside the circle, then C and its immediate neighborhood is closer to U than to P,Q,R. So the Voronoi Cells of P,Q,R are wrong.

(b) Let P,Q be two sites. Prove that the Voronoi cells of P and Q share an edge if and only if there exists a circle passing through P and Q and containing no other site.



Similar to Part (a).

(c) Prove that two sites P and Q share a semi-infinite edge if and only if PQ is an edge of CH(S).



(c) First suppose that PQ is an edge of $\mathrm{CH}(S)$. Then, all other sites are on one side of PQ. Consider a circle passing through P and Q and having center on the other side and on the perpendicular bisector of PQ. Since no three points are collinear, the segment PQ cannot contain another site. This means that when the center of the circle is sufficiently far away from PQ, it does not contain any site other than P and Q. As the center moves farther away, the circle becomes flatter and flatter, and will never contain other sites. Therefore all these centers belong to the boundary between $\mathrm{VCell}(P)$ and $\mathrm{VCell}(Q)$ (see Part (c) of Figure 134).

Conversely, suppose that $\mathrm{VCell}(P)$ and $\mathrm{VCell}(Q)$ share a semi-infinite edge. A circle passing through P and Q and having its center on any point on the semi-infinite edge does not contain any other site. Since the edge is semi-infinite, we can take the center arbitrarily far away from PQ. In the limit, the circle approaches the line PQ. This implies that all sites are on one side of PQ, so PQ is an edge of $\mathrm{CH}(S)$. Indeed, if there is a site on the other side of PQ, the circle would eventually become flat enough to include that site, contradicting the hypothesis that $\mathrm{VCell}(P)$ and $\mathrm{VCell}(Q)$ share a semi-infinite edge.

8. Prove that any algorithm for computing the Voronoi diagram of n sites must take O(n log n) time in the worst case.

Use reduction CH <= VOR. Pass S for CH to S for VOR (O(n) time). Given Vor(S), CH(S) can be identified from the semi-infinite edges. Since the size of Vor(S) is linear in n, this can be done in O(n) time. Therefore an o(n log n) algorithm for Vor(S) gives a o(n log n)-time algorithm for CH(S).