Algorithms and Data Structures
K. Mehlhorn and R. Seidel
Exercise 40

Summer 2008
Thu., Oct. 2nd
morning and afternoon

Motivation

1. Ask questions about the material presented in class.

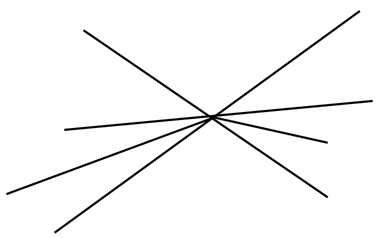
2. (trapezoidal decomposition) The trapezoidal decomposition induced by a set S of line segments is a decomposition of the plane into trapezoids. We assume for simplicity that no segment in S is vertical. From every endpoint of a segment in S and from any intersection point between two segments in S shoot a vertical ray upward and downward and extend it until its hits another segment.

   (a) Draw a small example.

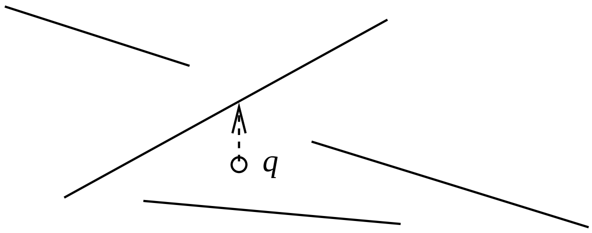
   (b) Show how to extend the sweep line algorithm for line segment intersection to an algorithm computing trapezoidal decompositions.

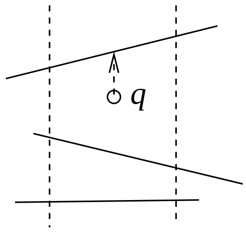
3. (line segment intersection, degenerate inputs) In class, we discussed the sweep line algorithm for line segment intersection under the assumption that no three segments intersect in a common point. Remove this assumption.

   Hint: do not distinguish three kinds of events (start point, endpoints, intersections), but have one one kind of event point. At an event point some segments end, some pass through, and some start. In the example below three segments pass through the event point, one ends and one starts.



4. Let S be a set of n *nonintersecting* line segments except for intersections at common endpoints. We want to derive a data structure that finds for a query point q, the segment immediately above q, i.e., the first segment intersected by an upward vertical ray emanating from q.





(a) Consider first the special case that all segments in S span the stripe between two vertices lines ℓ_L and ℓ_r and query points lie between these lines. Use binary search. Explain which comparison is made in every iteration and how it can be implemented. What is the running time of your method?

(b) Let $X = \{x_1, \ldots, x_m\}$ be the set of x-coordinates of the endpoints of the segments in S in sorted order. Any adjacent pair of coordinates defines an elementary interval $I_j = [x_{j-1}, x_j]$. Let T be a binary tree of depth $O(\log n)$ whose leaves correspond to the elementary intervals (of course in sorted order).

As in class, associate with every node v of T an interval $C(v)$. For a leaf v, $C(v)$ is the elementary interval associated with the leaf. For a non-leaf v, $C(v)$ is the union of the intervals associated with its children.

For a segment s, let I_s be the projection of s onto the x-axis and let

$$V_s = \{v \in T;\, C(v) \subseteq I_s \text{ and } C(parent(v)) \not\subseteq I_s \text{ or } v \text{ is the root}\}$$

be the node set of T supporting s.

We proved in class that $|V_s| = O(\log n)$. Here is an alternative argument. For this argument we assume that T is perfectly balanced, i.e., the number of leaves of T is a power of two and the depth is the logarithm of the number of leaves. Mark the leaves corresponding to elementary intervals contained in I_s. If both children of a node are marked, mark the node.

i. Argue that the marked nodes on any level of the tree are consecutive and that at most two of them have an unmarked parent.

ii. Conclude from this that $|V_s| = O(\log n)$.

iii. Can you extend the argument to trees that are not perfectly balanced? If you find a five line argument, sent it to KM.

Store any segment $s \in S$ at all nodes in V_s. Argue how to answer queries.

Hint: consider all nodes $v \in T$ with $q_x \in C(v)$.

5. Let S be a set of points in the plane. Design a data structure that finds for a query point q, the closest point in S. You may assume that you have an algorithm computing Voronoi diagrams.

6. Assume that you are given the Delaunay triangulation of a point set S in the form of a graph G, i.e., G has vertex set S and the edges of G are the edges of the triangles in the triangulation. Can you construct the Voronoi diagram of S from G?

7. Let S be a set of points in the plane which we call *sites*. For an arbitrary point $p \in \mathbb{R}^2$, let $furthest(p)$ be the set of sites in S that have largest distance from p, i.e.,

$$furthest(p) = \{\, s \in S;\ dist(p,s) \geq dist(p,t) \text{ for all } t \in S \,\} \ .$$

The furthest site Voronoi diagram $FVD(S)$ consists of all points p with $|furthest(p)| \geq 2$. We call the points with $|furthest(p)| \geq 3$ vertices and the maximal connected components of points with $|furthest(p)| = 2$ edges of the diagram.

(a) Draw a small example.

(b) Show that the vertices of FVD are centers of circles that contain all points in S and have three points in S on their boundary.

(c) Interpret the regions of FVD. For every region, there is a unique point $s \in S$ such that $furthest(p) = s$ for all p in the region. Which points in S have nonempty regions associated with them?

(d) Show that FVD has no bounded regions and hence is a tree.

(e) Let C be a circle of smallest radius containing all points in S. Show that the center of C lies on $FVD(S)$. Design an algorithm for finding C.

(f) In your example, draw a triangle for each vertex v of FVD. The vertices of the triangle are the sites in S defining v.

(g) Consider the lifting map $(x,y) \mapsto (x,y,x^2+y^2)$ that maps in $\mathbb{R}^2$ onto the paraboloid $z = x^2+y^2$. Let $L(S)$ be the set of lifted points. Is there a connection between the triangles in (f) and the upper convex hull of $L(S)$?

Have fun with the solutions.