Computational Geometry — Spring 2014 — Dan Halperin

Assignment no. 1

due: March 10th, 2014

Most of the exercises will be taken from the main textbook of the course *Computational Geometry Algorithms and Applications*[1] (CGAA). Some exercises are similar but not identical to exercises in the book. When a figure in the book could be helpful, the exercise number in the book is given.

Exercise 1.1 Devise an algorithm to compute the convex hull of a set of n points in the plane based on the divide-and-conquer paradigm. Start by developing an $O(n)$ time algorithm to compute the convex hull of two disjoint convex polygons having n vertices in total.

Exercise 1.2 In many situations we need to compute convex hulls of objects other than points. Let S be a set of (possibly intersecting) unit circles in the plane. We want to compute the convex hull of S.
(a) Show that the boundary of the convex hull of S consists of straight line segments and pieces of circles in S.
(b) Show that each circle can occur at most once on the boundary of the convex hull.
(c) Let S' be the set of points that are the centers of the circles in S. Show that a circle in S appears on the boundary of the convex hull if and only if the center of the circle lies on the convex hull of S'.
(d) Give an $O(n \log n)$ algorithm for computing the convex hull of S.

Exercise 1.3 Describe in detail a "gift-wrapping" algorithm for computing the convex hull of a finite set of points in three-dimensional space and analyze its running time. You may assume that the input points are in general position, which means, in particular, that no four points lie on a common plane, that no three points lie on a common line, etc.

Exercise 1.4 Let S be a set of n disjoint triangles in the plane. We want to find a set of $n-1$ segments with the following properties:

- Each segment connects a point on the boundary of one triangle to a point on the boundary of another triangle.
- The interiors of segments are pairwise disjoint and they are disjoint from the triangles
- Together they connect all the triangles to each other, that is, by walking along the segments and the triangle boundaries it must be possible to walk from a triangle to any other triangle.

Develop a plane sweep algorithm for this problem that runs in $O(n \log n)$ time. State the events and the data structures that you use explicitly, and describe the cases that arise and the actions required for each of them. Also state the sweep invariant. (CGAA Ex. 2.13)

Exercise 1.5 Let S be a set of n disjoint line segments in the plane, and let p be a point not on any of the segments in S. We wish to determine all the line segments of S that p can see, namely, all the line segments of S that contain some point q so that the open segment $\overline{pq}$ does not intersect any line segment of S. Give an $O(n \log n)$ time algorithm to solve this problem. (CGAA Ex. 2.14)

[1] M. de Berg, M. van Kreveld, M. Overmars, and O. Schwarzkopf, *Computational Geometry: Algorithms and Applications*, 3rd Edition, Springer, 2008.