AMS 545/CSE 555 (Fall, 2007) Joe Mitchell

COMPUTATIONAL GEOMETRY
Homework Set # 1

Due at the beginning of class on Thursday, September 20, 2007 *Reminder: Show your reasoning!*

Recommended Reading: BKOS: Chapter 1; O'Rourke Chapter 3.

DO ANY 4 OF THE FOLLOWING 5 PROBLEMS.

(1). [25 points] Problem 1.1, page 15, BKOS.

(2). [25 points] Problem 1.10, parts (a)-(d), page 17, BKOS. (Part (e) is not too tricky, if you want to think about it for extra credit.)

(3). [25 points] We defined a set Q to be convex if for any two points p and q in Q the line segment $\overline{pq}$ also lies in Q.

Let us define the notion of "rectangle-convex" as follows: Q is *rectangle-convex* if for any $p, q \in Q$, the axis-aligned bounding box, $BB(\{p, q\})$, of p and q also lies in Q. Given a set $S = \{p_1, \ldots, p_n\}$ of n points in the plane, give an efficient algorithm to compute the rectangle-convex hull, $RCH(S)$, of S. What is the running time of your algorithm (in big-Oh notation)?

[Optional, extra credit]: Similarly, let us define the notion of "circle-convex" as follows: Q is *circle-convex* if for any $p, q \in Q$, the minimum enclosing ball (circular disk), $MEB(\{p, q\})$, of p and q also lies in Q. Given a set $S = \{p_1, \ldots, p_n\}$ of n points in the plane, give an efficient algorithm to compute the circle-convex hull, $CCH(S)$, of S. What is the running time of your algorithm (in big-Oh notation)?

(4). [25 points] In class, we discussed the divide-and-conquer approach to computing the convex hull of a set of n points in the plane. (See also problem 1.8, Chapter 1.) Suppose now that the method of dividing is *not* to split according to the x-median, but, rather, is an arbitrary split of the input points S into two sets, S_1 and S_2, of sizes $\lfloor n/2 \rfloor$ and $\lceil n/2 \rceil$.

(a). What is different in the merge step of the algorithm? Draw a picture.

(b). Argue that the merge can still be done in time $O(n)$, so that the overall algorithm is $O(n \log n)$. (Note: It may be helpful to realize that two sorted lists, each of length $O(n)$, can be merged in time $O(n)$ (simply walk through each of the two lists in "lock step").)

(5). [25 points] A property of a convex region X in the plane is that any line intersects X in a single connected component (a line segment).

(a). Argue equivalence: Set X is convex if and only if the intersection of any line ℓ with X is a connected set (a line segment on ℓ), possibly the empty set.

(b). Now suppose we restrict ourselves to the set L of all horizontal or vertical lines: We say that X is L-convex if the intersection of X with any $\ell \in L$ is a connected set (line segment), or empty. Draw an example of an L-convex set.

(c). Define the notion of an "L-convex hull" of n points in the plane and draw an example.

(d). Describe an algorithm to compute the L-convex hull in time $O(n \log n)$ using ideas from computing convex hulls efficiently.

(C1). [optional – can be turned in after the due date] *CHALLENGE PROBLEM*: We can think of the convex hull of a point set S as the region bounded by a shortest "rubber band" that surrounds *all* of the points S. Suppose now that we want the shortest rubber band that surrounds *all but* k of the n points of S. (Think of the k omitted points as "outliers".) How efficiently can you solve this? Consider both "small" values of k and also situations in which k may be "large" (close to n).

Practical question: How would you *really* solve this question in practice? (You may be able to devise a project based on a thorough investigation/implementation.)