COMPUTATIONAL GEOMETRY (0368-4211) – FINAL EXAM

Prof. Micha Sharir
Blavatnik School of Computer Science
Raymond and Beverly Sackler Faculty of Exact Sciences
Moed Alef, August 8, 2013

Answer four of the following six problems. All problems have equal weight (25 percent). You may use any written material.
The exam is 3 hours long.
You may assume general position of the input.
Good luck!!

Problem 1

Let P be a set of n points in the plane. For each pair of distinct points $a, b \in P$, let $V(a, b)$ denote the set of all points $q \in \mathbb{R}^2$ such that

$$d(q,a),\ d(q,b) \le d(q,c) \text{ for all } c \in P \setminus \{a,b\}.$$

That is, a and b are the two points of P nearest to q (we don't care which of a, b is nearer to q).

Let $\mathrm{Vor}_2(P)$ denote the partition of the plane into all the nonempty cells $V(a, b)$.

(a) What is the shape of each cell $V(a, b)$? Show that these cells cover the plane and have pairwise disjoint interiors.
(b) Show that $V(a, b)$ is nonempty if and only if a and b are neighbors in the standard Voronoi diagram $\mathrm{Vor}(P)$.
(c) What is the maximum possible number of nonempty cells of $\mathrm{Vor}_2(P)$? (Give a concrete upper bound, without the $O(\cdot)$ notation.)
(d) Show that $\mathrm{Vor}_2(P)$ can be computed in $O(n \log n)$ time.

Problem 2

(a) Let $\mathcal{D} = \{D_1, D_2, \ldots, D_n\}$ be a set of n unit disks in the plane. Describe an $O(n \log n)$-time algorithm for computing the convex hull of $\mathcal{D}$ (that is, the convex hull of the union of thee disks of $\mathcal{D}$). Describe also the geometric structure of the hull.
(b) Solve the same problem (both parts) for a set of n unit balls in three dimensions.

Problem 3

(a) Let $\mathcal{R}$ be a collection of n axis-parallel rectangles in the plane. Give an algorithm that finds, in $O(n \log n)$ time, a point q that lies in the maximum number of rectangles of $\mathcal{R}$. (**Hint:** Use a sweep.)
(b) Preprocess $\mathcal{R}$ into a data structure, so that, for any given query point q, we can find, in $O(\log n)$ time, the *depth* of q, which is the number of rectangles of $\mathcal{R}$ containing q. What are the storage and preprocessing costs of the structure?

Problem 4

Let $E = \{e_1, e_2, \ldots, e_n\}$ be a set of n line segments in the plane. Using duality, construct a data structure on E that can answer in $O(\log n)$ time queries of the form: Given an arbitrary (non-vertical) line ℓ, count the number of segments of E that ℓ intersects. How much storage and preprocessing does the data structure require?

Problem 5

Let R be a set of n rays in the plane.
(a) Suppose that the orientations of all the rays of R (when oriented from their endpoint outwards) lie in the first quadrant (they are all between 0 and $\pi/2$). How fast can you find a line that intersects all the rays, or report that no such line exists?
(b) Suppose that the orientation of each ray (when oriented from its endpoint outwards) lies either in the first or in the third quadrant (each such orientation is either between 0 and $\pi/2$ or between π and $3\pi/2$). How fast

can you find a line *of negative slope* that intersects all the rays, or report that no such line exists?

Problem 6

Let P be a set of n points in the plane. Let $\delta > 0$. The graph $G(\delta)$ is the graph whose vertices are the points of P, and whose edges connect all the pairs $p, q \in P$ whose distance is at most δ.
(a) Given P and δ, give an efficient algorithm for computing $G(\delta)$, which runs in time $O((n+k)\log n)$, where k is the number of edges of $G(\delta)$. (**Hint:** Transform each point into "something", so that (p, q) is an edge of $G(\delta)$ iff the "something" of p and the "something" of q intersect.)
(b) Suppose that $G(\delta)$ is not connected. Give an efficient algorithm for finding the smallest $\delta' > \delta$ such that $G(\delta')$ has fewer connected components than $G(\delta)$. That is, find a shortest segment pq that connects two points p, q in different connected components of $G(\delta)$. (**Hint:** Show that such a pair p, q must be neighbors in $\mathrm{Vor}(P)$.)