Homework Assignment 4

Give written (and legible) answers to the following questions. Please give succinct and mathematically precise answers. Whenever you are asked to give an algorithm for a problem, you should present the following: the algorithm, a justification of its correctness, and a derivation of its running time. Write clear and convincing pseudo-code for your algorithms. All work must be done independently.

I. (30 points) We studied a randomized incremental algorithm for constructing the trapezoidal decomposition. Recall that the geometric structure of the final trapezoidal decomposition is always the same, regardless of the order in which the segments are inserted. However, there are instances of probabilistic constructions in which the final geometric structure depends on the particular order in which the objects are inserted. In this question, you will study one such structure called the Binary Space Partition (BSP).

You are given a set of \( n \) non intersecting line segments in the plane, and you build a subdivision recursively as follows: Initially, the subdivision contains no segments and only one face, the entire space. When the first segment is inserted, this face is partitioned into two faces by a splitting line that contains this segment. In general, as each segment is added, every face of the existing subdivision that intersects this segment is split by the line containing the segment. An example is shown in Figure 1. In (a), the order of insertion is \( s_1, s_2, s_3, s_4, s_5 \). In (b), the order of insertion is \( s_4, s_5, s_3, s_1, s_2 \). Observe that different orders of insertions of the segments result in different subdivisions.

When a splitting line is added, each future segment which intersects this splitting line is said to be cut by the line. (Therefore, if we say that a segment \( v \) is cut by a splitting line containing \( u \), then \( v \) must have been inserted after \( u \).) In Figure 1(a), segment \( s_3 \) is cut by the line extending segment \( s_1 \), and segments \( s_3 \) and \( s_5 \) are cut by the line extending \( s_2 \). Note that segment \( s_1 \) is not considered to be cut by \( s_2 \) (because \( s_2 \) was inserted after \( s_1 \)).

1. Given \( n \) segments in general position, prove that the number of faces of the final BSP subdivision is equal to \( n + 1 \) plus the total number of cuts (verify this in the figure above). \( \text{Hint: Use induction.} \)
2. Give an example of a set of \( n \) nonintersecting line segments such that for one insertion order, the resulting BSP has size \( O(n) \) and for another insertion order, the BSP has size \( \Omega(n^2) \).
3. Given two line segments \( u \) and \( v \), define \( \text{index}(u,v) \) as follows: If the line containing \( u \) does not intersect \( v \), then \( \text{index}(u,v) = \infty \). Otherwise, define the index to be the number of line segments that intersect this line and lie between \( v \) and the closest endpoint of \( u \). (In the figure below, \( \text{index}(u,v) = 3 \). In Figure 1(a), \( \text{index}(s_2,s_5) = 1 \), and \( \text{index}(s_1,s_3) = 0 \).) Prove that if the segments are inserted in random order, then the probability that \( u \) cuts \( v \) is at most \( \frac{1}{\text{index}(u,v)+2} \).
Figure 1: Two different binary space partitions for the same set of segments.

4. Assuming that segments are inserted in random order, use the results of parts (a) and (c) to prove that the expected number of faces in the subdivision is $O(n \log n)$. Hint: You need to show that the expected total number of cuts is $O(n \log n)$. To show this, you should first find the expected number of segments that the $i$-th segment (i.e., $i$-th in the random order) cuts.

II. (20 points) The following questions relate to Voronoi diagrams, and their properties. Let $P = \{p_1, p_2, \ldots, p_n\}$ be a set of $n$ sites. We say that sites $p_i$ and $p_j$ are *Voronoi neighbors* if their Voronoi regions share an edge $e$ in the Voronoi diagram. In other words, $e \subset \text{bis}(p_i, p_j)$ is a Voronoi edge.

1. Give an $O(n \log n)$ algorithm to find (a) for every site $p \in P$, the site that is closest to it, and (b) the closest pair of sites (i.e., the pair of sites that are closer to each other than any other pair).

2. Prove or disprove: Suppose $p_i$ and $p_j$ are Voronoi neighbors. Let $e$ be the Voronoi edge between them. Then the midpoint of $p_i$ and $p_j$ must belong to $e$.

3. Let $R$ be a set of red points and $B$ be a set of blue points, and assume that $R$ and $B$ are $x$-separated (that is, assume all points in $R$ lie to the left of all points in $B$). Let $r' \in R$ and $b' \in B$ be the pair of points such that $\text{dist}(r', b') = \min\{\text{dist}(r, b) | r \in R, b \in B\}$. That is, $(r', b')$ is a closest red-blue pair of points. Prove or disprove: $r'$ and $b'$ are Voronoi neighbors in $\text{Vor}(R \cup B)$ (the Voronoi diagram of $R \cup B$).

III. (15 points) Computing geometric properties of a union of spheres is important to many
applications in computational biology. The following problem is a 2-dimensional simplification of one of these problems.

You are given a set $P$ of atoms forming a protein, which for our purposes will be represented by a collection of circles in the plane, all of equal radius $r_a$. Such a protein lives in a solution of water. We will model a molecule of water by a circle of radius $r_b > r_a$.

We say that an atom $a \in P$ is solvent-accessible if there exists a placement of a water molecule that is tangent to $a$, and the water molecule does not intersect any other atoms in $P$. In the figure below, all atoms are solvent-accessible except for three (shaded). Given a protein molecule $P$ of $n$ atoms, devise an $O(n \log n)$ time algorithm to determine all solvent-inaccessible atoms of $P$.

IV. (25 points) In this problem we will investigate the properties of a variant of the Voronoi diagram, in which we replace the notion of “nearest” with “farthest”. Given a set $P = \{p_1, p_2, \ldots, p_n\}$ of point sites and site $p_i \in P$, recall that we defined Vor($p_i$) to be the set of all points in the plane that are strictly closer to $p_i$ than to any other site of $P$. Define Far($p_i$) to be the set of all points in the plane that are strictly farther from $p_i$ than from any other site of $P$. As we did with Voronoi diagrams, we define the farthest-point Voronoi diagram to be the union of the boundaries of Far($p_i$) for all points in $P$. Let NVD($P$) denote the nearest-point Voronoi diagram of $P$ and let FVD($P$) denote the farthest-point Voronoi diagram of $P$.

1. Show that Far($p_i$) is a convex polygon (possibly unbounded).
2. Unlike the nearest-point Voronoi diagram, a site may not contribute a region to the farthest point diagram. Show that Far($p_i$) is nonempty if and only if $p_i$ is a vertex of the convex hull of $P$. (Hint: Recall that a site $p_i$ is on the boundary of $P$’s convex hull if and only if there is a halfplane containing $P$ whose boundary line passes through $p_i$.)
3. Draw a picture of three points in the plane. Show (in different colors) NVD($P$) and FVD($P$). Label each region according to which sites are closest and farthest.
4. Repeat (c), but this time with four points, such that only three of the four points are on the convex hull.
5. Repeat (d), but this time with all four points on the convex hull. (Try to avoid degeneracies, such as four co-circular points or parallel sides.)

V. (10 points) The Euclidean metric is but one way to measure distances (this is the metric that we used when discussing Voronoi diagrams). In this question, we will examine another distance metric, called the $L_\infty$ metric, which is defined as follows
\[
\text{dist}_\infty(p,q) = \max(|p_x - q_x|, |p_y - q_y|).
\]

1. Given a point \(q\), describe the set of points that are at \(L_\infty\) distance \(w\) from \(q\).

2. Given two distinct points \(p\) and \(q\), describe the set of points that are equidistant from \(p\) and \(q\) in the \(L_\infty\) distance. \textit{Hint}: The bisector is a polygonal curve. As a function of the coordinates of \(p\) and \(q\), indicate how many vertices this curve has, and where these vertices are located. I would like the exact formula, and not simply a high-level description.

Please upload your homework assignment on Sakai by 11:55pm on April 8, 2017.