Final Exam Geometric Algorithms, April 17, 2012, 9.30–12.30

Read every question carefully, make sure you understand it, and be sure to answer the question. Read the question again after answering it, as a check whether you really answered the question. Answer the easier questions first, and then the harder ones. You may not use any algorithmic result from the book, unless it is explicitly stated that this is allowed in the question. Answer questions in sufficient but not too much detail. You may not use the textbook during the exam.

Be sure to put your name on every piece of paper you hand in. Also write down your studentnummer. If you write readable, unambiguous, and technically correct, you get one point for free (in particular, do not write "line" if you mean "line segment" and do not write "Step 1 takes $n \log n$ time" when you mean "Step 1 takes $O(n \log n)$ time"). The other nine points can be earned by answering the questions correctly. Good luck!

1. (2.5 points) What is the worst-case (or expected worst-case) running time of the algorithms for (use k for the output size whenever appropriate):

    (a.) computing the common intersection of a set of n half-planes in the plane explicitly?

    (b.) computing the smallest enclosing disk of n points?

    (c.) computing the arrangement of a set of n lines in the plane?

    (d.) computing the Voronoi diagram of a set of n disjoint line segments?

    (e.) performing a 3-dimensional range query on a kd-tree storing n points in 3D?

    (f.) performing a windowing query (reporting) with a rectangular query window in a set of n arbitrarily oriented, non-intersecting line segments in the plane?

    (g.) building a planar point location structure for the faces of an arrangement of n lines?

2. (1 point) Suppose we have a planar subdivision in doubly-connected edge list representation (DCEL), and the total number of inner components in all face objects is 1. Suppose the DCEL has 11 vertex objects and 38 half-edge objects, how many face objects does it have? Explain how you obtained your answer.

3. (1.5 point) In the plane sweep algorithm to compute the Voronoi diagram of a set of n point sites in the plane, there were site events and circle events. Some of the *potential* circle events that were inserted in the event list will not really take place. These were called *false alarms*.

    (a.) Explain the reasons why a circle event that at first seemed to take place at some point in the future, might not happen at all.

    (b.) The number of real circle events is $O(n)$, because at each circle event that takes place a new Voronoi vertex is discovered, and there are no more than $O(n)$ Voronoi vertices in a Voronoi diagram of n point sites. Give a clear argument that shows that the number of false alarms (circle events that are stored in the event list but will not happen) is also bounded by $O(n)$ throughout the whole sweep.

1

4. (1 point) In the plane, there are two line segments s_1 and s_2 that lie on the same line ℓ which has slope 1. s_1 and s_2 have no point in common and s_1 is to the left of s_2. There is a point p in the plane as well, and p is vertically above s_1. We are interested in all lines that contain p and intersect s_2.

Formulate the above paragraph in its dual form using the usual point-line duality. Then draw a possible situation with s_1^*, s_2^*, ℓ^*, and p^* in the dual plane, and indicate where the lines mentioned in the last sentence of the paragraph above can be found in the dual plane.

5. (1.5 points) Suppose we want to solve the following query problem: Given a set of n axis-parallel rectangles in the plane, store them in a data structure so that for any given query point q, all rectangles that contain q can be reported efficiently.

Describe a data structure that solves this query problem. You may be brief (as long as it is correct), like: "The structure has a main tree that is a AAA-tree with BBB in the leaves sorted on CCC-coordinate, and every internal node has an associated structure that is a DDD-tree on ...." Of course, you should substitute AAA, BBB, CCC, and DDD with the appropriate tree name or whatever is appropriate, and complete the sentence.

Your data structure should be as efficient as possible in storage requirements and query time. Give the storage requirements and query time of your solution, and briefly reason why you get these bounds.

6. (1.5 points) Let S be a set of n disjoint, non-horizontal line segments in the plane. Let H be a set of m horizontal line segments in the plane; assume that no two of them have the same y-coordinate.

Give an efficient plane sweep algorithm that sweeps a horizontal line from top to bottom over the plane, and reports all line segments of S that are intersected by at least one of the horizontal line segments in H. Describe the type of events and what happens precisely. Also analyze the running time of the algorithm, and express the time bound in n and m.