Final Exam Geometric Algorithms, March 20, 2009, 9–12

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 by quoting it only, 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 “collegekaartnummer”. If you write readable and unambiguous enough, you get one point for free. The other nine points can be earned by answering the questions correctly. Good luck!

1. (1 point) Let a doubly-connected edge list D of a planar subdivision be given.

   Give the pseudo-code for a method that puts a new edge as a separate component in a face. The method MakeEdgeInFace(f, p, q) gets a face object f from D and two points p and q (not yet in D), and changes D such that $\overline{pq}$ is represented as an edge in D.

   Use the standard names of the doubly-connected edge list stucture, like *Origin*, *Twin*, etc.

2. (2 points) We consider randomized incremental construction to solve the planar point location problem (Chapter 6 of the book). Suppose that the structure was built and now we have a search graph that can report the face containing any query point efficiently.

   (a.) Explain why the expected size of the search structure is $O(n)$.

   (b.) Give an example with n line segments and a insertion order that leads to a quadratic size search structure.

3. (1 point) The edges and triangles of the Delaunay triangulation have the *empty circle property*. Explain what this means for both edges and triangles.

4. (1 point) Prove that every simple polygon has a triangulation.

5. (2 points) Let S be a set of non-intersecting line segments in the plane. They also do not share endpoints. A line segment $s \in S$ is *topmost* if there exists an x-coordinate c such that the line $y = c$ intersects s, and s is highest of all line segments intersecting this line.

   Give a plane sweep algorithm to report all line segments of S that are topmost. Describe the events and event handling, the status and status structure, and outline the algorithm. Also, analyse the efficiency of the algorithm.

6. (1 point) Give the storage requirements and the query time for a query (point containment query; reporting all intervals that contain the query point) in an interval tree, assuming that it stores a set of n intervals.

   Also, describe how such a query is performed.

7. (1 point) In the plane, there is a point p that lies vertically below a line ℓ and above a line segment s. The line ℓ and the line segment s intersect to the right of p (that is, the intersection point has a larger x-coordinate than p).

   Describe the situation in the dual plane, without missing any statement from the primal plane.