Computational Geometry Exercise Set 2 WS06/07

URL: http://www.ti.inf.ethz.ch/ew/courses/CG06/

Exercise 1 (10 points)

We consider the mapping that assigns to a point \( p = (p_x, p_y) \in \mathbb{R}^2 \) the line \( p^* : y = p_x x - p_y \) and, vice versa, assigns to the line \( g : y = mx + b, m, b \in \mathbb{R} \) the point \( g^* = (m, -b) \).

1. Show that this mapping preserves incidences, i.e. for a point \( p \) and a line \( g \) it holds \( p \in g \iff g^* \in p^* \).

2. Show that this mapping preserves order, i.e. for a point \( p \) and a line \( g \) it holds: \( p \) is above \( g \iff g^* \) is above \( p^* \).

3. Describe the image of the following point sets under this mapping

   (a) a half plane
   (b) \( k \geq 3 \) colinear points
   (c) a line segment
   (d) the boundary points of the upper convex hull of a finite point set.

4. Consider the parabola \( \mathcal{P} : y = x^2/2 \). For \( p \in \mathcal{P} \) characterize \( p^* \) with respect to \( \mathcal{P} \).

Exercise 2 (10 points)

The lower envelope of a set \( G \) of non-vertical lines in \( \mathbb{R}^2 \) is defined to be the set of all points \( p \) such that

- \( p \) lies on (at least) one line in \( G \) and
- there is no line in \( G \), which is strictly below \( p \).

Describe an \( O(n \log n) \) algorithm which computes the lower envelope of a set of \( n \) non-vertical lines in \( \mathbb{R}^2 \).

Exercise 3 (10 points)

Consider a set \( M \subset \mathbb{R}^2 \) of \( n \) points. Describe an algorithm which decides in linear time if another point \( q \) lies in \( \text{conv}(M) \).
Exercise 4 (10 points)

For a sequence of \( n \) pairwise distinct numbers \( y_1, \ldots, y_n \) consider the sequence of pairs \((\min(y_1, \ldots, y_i), \max(y_1, \ldots, y_i))\) \(i = 0, 1, \ldots, n\) (\( \min \emptyset := +\infty, \max \emptyset := -\infty \)). How often do these pairs change in expectation if the sequence is permuted randomly, each permutation appearing with the same probability? Determine the expected value.

Due date: 13. November 2006, 13h15