4. Assignment on

Computational Geometry

Panos Giannopoulos, Wolfgang Mulzer, Lena Schlipf SoSe 2013

Due on 6.May 2013 in the tutorial session

Problem 1 Doubly-connected edge list (DCEL) *10 points*

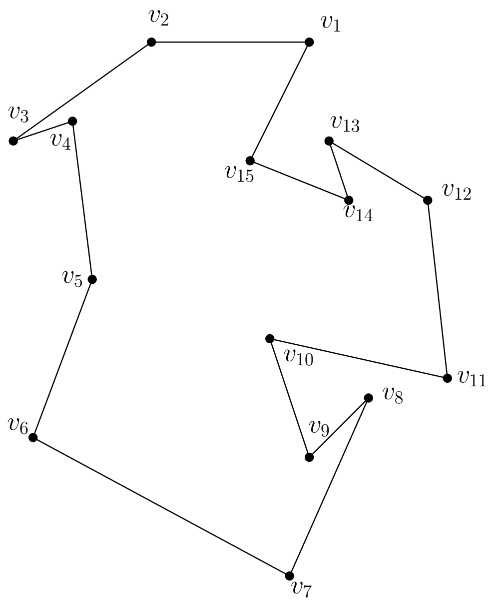
(a) Given a DCEL and one of its halfedges $\vec{e}$, which of the following are always true?

$Twin(Twin(\vec{e})) = \vec{e}$
$Next(Prev(\vec{e})) = \vec{e}$
$Twin(Prev(Twin(\vec{e}))) = Next(\vec{e})$
$IncidentFace(\vec{e}) = IncidentFace(Next(\vec{e}))$

(b) Give pseudocode for an algorithm that lists all vertices adjacent to a given vertex v in a doubly-connected edge list. Also, give pseudocode for an algorithm that list all edges that bound a face in a not necessarily connected subdivision.

Problem 2 Partitioning a polygon into monotone pieces *10 points*

Consider the following simple polygon:



Show how the polygon can be partitioned into y-monotone sub-polygons by using the algorithm we have seen in the lecture. *Note:* You should use the version of the algorithm in de Berg et al. [1, Section 3.2] and you should report all steps, all necessary information in every step (type of vertex, contents of status, etc.), and of course the diagonals inserted.

Problem 3 Point Set Triangulation *10 points*

Let P be a set of n points in the plane such that no two points in P have the same x-coordinate and such that no three points in P are on collinear. A *triangulation* T of P is a maximal planar straight-line graph with vertex set P. This means that the edge set of T is a maximal set of line segments with endpoints in P such that no two segments cross.

(a) Let T be a triangulation of P. Show that all the bounded faces of T are triangles and that the unbounded face is the exterior of the convex hull of P.

(b) Give an algorithm that finds a triangulation of P in $O(n \log n)$ time.

Hint: For example, you could adapt the polygon triangulation algorithm you saw in class. There is also a very simple incremental algorithm that builds the triangulation from left to right (or top to bottom etc.).

Literatur

[1] M. de Berg, O. Cheong, M. van Kreveld, and M. Overmars. *Computational Geometry Algorithms and Applications*. Springer, 1997, 2000, 2008.