Due Thursday November 27, 23:59, by e-mail (pdf)

Requirements: Assignments have to be typeset in English. They have to be handed in by each student separately, except for exercises marked with (G), which should be solved and handed in by each (project) group. Please include your name on top of the first sheet.

How to describe algorithms: Whenever you are asked to describe an algorithm, you should present three things: the algorithm, a proof of its correctness, and a derivation of its running time.

A careful geometric proof can be quite cumbersome at times, so you are highly encouraged to use intuition and give illustrations whenever appropriate. However, beware of “proof by picture”: a poorly drawn figure might make you ignore special cases that are not apparent in your drawing. Try to keep your answers concise and to the point. Never give detailed code. Only give sufficient detail to convince us that your method and your running time analysis are complete and correct.

Unless stated otherwise, you can assume that objects are in general position, that is, no three points are collinear and no four points are cocircular. You may also assume that any geometric primitive that involves only a constant number of objects which are each of constant complexity can be computed in $O(1)$ time.

Exercise 1: [5 points] Given a simple polygon $P$ with $n$ vertices, show how to compute a diagonal that splits $P$ into two polygons with at most $\lfloor 2n/3 \rfloor + 2$ vertices. Your algorithm should run in at most $O(n \log n)$ time. Hint: Use the dual graph of the triangulation.

Exercise 2: [5 points] Given a trapezoidal map of a simple polygon $P$ with $n$ vertices, show how to compute a triangulation of $P$ in $O(n)$ time. Hint: Use as part of the solution a linear-time algorithm from the lectures.

Exercise 3 (G): [2+1+1+0* points] We are given a simple polygon $P$ with $n$ vertices and want to construct the trapezoidal map (of the edges) of $P$ using a randomized incremental construction

(a) We assume that we are using a randomized incremental construction (as given in the lecture) to construct the trapezoidal map of the edges of $P$. Let $D_j$ be the point location data structure after adding $j$ edges. Fix any point $q$ in the plane as a query point. Assume we have located $q$ in $D_j$ for some $j$. Show that for $j < k \leq n$ we can now locate $q$ in $D_k$ in expected time $O(\log(k/j))$.

(b) Still in the setting of a randomized incremental construction let $T_j$ be the trapezoidal map after adding $j$ edges. Prove that the expected number of proper intersections between $T_j$ and $P$ is $O(n)$. Hint: Consider a random edge of $P$. What do you know about the number of intersections with $T_j$?
(c) Combining (a) and (b) we can try to speed up the randomized incremental construction of the trapezoidal map. Let $e_1, \ldots, e_n$ be the edges of $P$ in random order. Assume we have just constructed $T_j$ and $D_j$. Explain how to use (b) to locate all end points of edges of $P$ in $O(n)$ time, thus in particular the end points of $e_{j+1}, \ldots, e_k$ (for some given $k$). Now we can construct $T_k$ and $D_k$ (for $k > j$) fast using (a). When $O(\log(k/j))$ gets too big we switch back to using (b) (and then (a) again and so on). Give (brief) pseudo-code for the algorithm. Analyze the algorithm for the case that (b) is used for $j = 2^i$, $i = 1, \ldots, \lfloor \log n \rfloor$.

(d) Try to further speed up the algorithm by balancing the terms better (that is, use (b) less frequent). (Note: If you solve (d), your score on exercise 3 is $\min(4, \text{score on (a)} + (b) + (c) + 1)$).