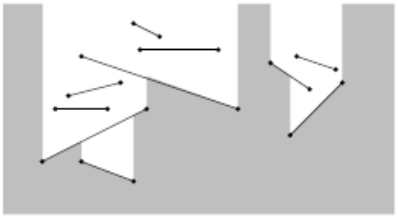
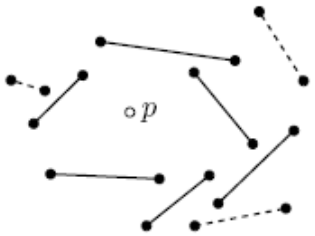
Computational Geometry Problems

1. Consider a set of n disjoint line segments in the plane $S = \{s_1, s_2, \ldots, s_n\}$, where each segment s_i is defined by its left and right endpoints p_i and q_i. A point that lies on one of these segments is said to lie on the lower envelope if an open vertical ray shot downward from this point does not intersect any other segment. (The portions of the segments touching the shaded region in the figure below form the lower envelope.) Give a plane sweep algorithm that, given S, outputs the sequence of segments (or sub-segments) that form the lower envelope. (If you like, you may assume that there is a special horizontal sentinel segment, s_0, of infinite length that lies well above all the segments.) Your algorithm should run in O(n log n) time. You may make whatever general-position assumptions you like.



2. Let S be a set of n disjoint line segments in the plane, and let p be a point not on any of the line segments of S. We wish to determine all line segments of S that p can see, that is, all line segments of S that contain some point q so that the open segment pq does not intersect any line segment of S. Give an O(n log n) time algorithm for this problem that uses a rotating half-line with its endpoint at p.



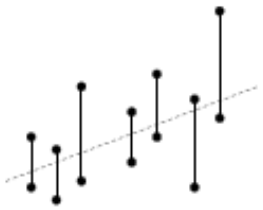
3. Let R be a set of n rectangles, $R_1, \ldots, R_n$, all them having their base on the x-axis. Each rectangle R_i is defined by its lower left vertex, $(x_i, 0)$, its width, a_i, and its height, h_i. The profile of R is the upper envelope of the rectangles and the x-axis. a) Prove that the profile of R has at most 2n vertical segments. b) Prove that it has at most 2n + 1 horizontal segments. c) Consider the profiles corresponding to two sets of rectangles, R´ and R´´, consisting of a total number of n´, respectively n´´ horizontal and vertical segments. Prove that the number of intersections of the two profiles is O(n´ + n´´) (you may suppose that there is no overlapping among the segments of the two profiles). d) Give an algorithm to compute the profile of a set R of n rectangles.

4. Construct an efficient algorithm, which given a set of vertical segments, reports all pairs of segments having nonempty intersection.

5. Show how to test in linear time if a point p is inside the convex hull of a set S of n other points. Only the set S is given, not its convex hull.

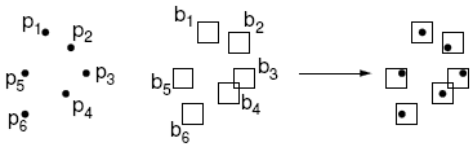
6. Let S be a set of n points in the plane. Show how to prepare a data structure that uses O(n) storage, so that, given any query line l, we can determine, in O(log n) time: a) whether l separates S (i.e., each of the halfplanes bounded by l contains points of S); b) the point of S farthest from l.

7. You are given a set of n vertical line segments in the plane. Present an O(n) time algorithm to determine whether there exists a line that intersects all of these segments. An example is shown in the figure below. Justify your algorithm's correctness and derive its running time.



8. The following problem arises in point pattern matching and image registration in computer vision. You are given a sequence of n points in the plane $P = \{p_1, p_2, \ldots, p_n\}$ and sequence of n closed, axis-aligned rectangles $B = \{B_1, B_2, \ldots, B_n\}$. Describe O(n) time algorithms to solve each of the following problems:

a) Does there exist a translation vector, $t = (t_x; t_y)$ such that for $1 <= i <= n$, the translated point $p_i + t$ lies within the corresponding rectangle B_i? (You may assume that the rectangles B_i are represented in any manner that is convenient to your solution.)



b) An affine transformation T is given by six coefficients, a_{11}, a_{12}, etc. and maps a point $p = (p_x, p_y)$ to the point: $T(p) = (a_{11}p_x + a_{12}p_y + a_{13}, a_{21}p_x + a_{22}p_y + a_{23})$.

(Note that a translation is just a special case where $a_{11} = a_{22} = 1$ and $a_{12} = a_{21} = 0$.)

Generalize your solution to a), to determine whether there exist an affine transformation that maps each point pi of P to lie within the corresponding rectangle Bi.

In both cases, it is important that the correspondences between points and boxes is given in advance. (Otherwise, the problem is much harder.)

9. You are given two sets of points A and B, which are separated by a vertical line (so that A lies to the left of B). The total number of points in these two sets is n. Compute in O(n) time, the non-vertical line that passes through one point of A and one point of B such that no point of $A \cup B$ lies above this line (upper tangent connecting the two sets).

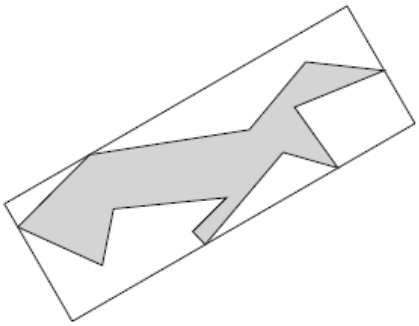
10. Let P and Q be two disjoint point sets in the plane. Let p in P and q in Q be two points from these sets that minimize the Euclidean distance d(p,q). Prove that the segment pq is an edge of DT(P $\cup$ Q). Give an O(n log n) algorithm for finding p and q.

11. Given a set S of n points in the plane, give a method of constructing an efficient data structure that allows one to answer quickly a query of the following form: Is there a point of S within distance r of point q?

12. Show how to do point-in-polygon queries in time O(log n), using O(n) preprocessing time and space, for the case of: a) a convex polygon; b) an x-monotone polygon.

13. We want to solve the following query problem: Given a set S of n disjoint line segments in the plane, determine the first segment that is stabbed by a vertical ray running from a point q vertically upwards to infinity. Describe briefly a data structure and a method for this problem. Try to be as efficient as possible in both space and query time.

14. Let P be a simple planar polygon with n vertices. A bounding box of P is a rectangle that completely contains P. This rectangle does not have to be axis-aligned but it can have any arbitrary orientation. Give an algorithm that computes in O(n log n) time a bounding box for P that has: a) the smallest perimeter among all possible bounding boxes; b) the smallest area among all possible bounding boxes.



15. Let P be an x-monotone polygonal line with n vertices, all with positive y coordinate, and let a and b respectively be the minimal and maximal x-coordinates of the points in P. We call terrain the portion of the plane enclosed by the x-axis, the vertical lines x = a and x = b, and the polygonal line P. We say that P is the profile of the terrain. a) Characterize the set S of all the points lying in the strip a <= x <= b and above P which can see all of P. Give an algorithm O(n) to find the point of S of minimum y-coordinate. b) Give an O(n log n) algorithm to find the point of the profile allowing to construct the shortest tower guarding the entire profile P.