CS 473: Algorithms, Fall 2009
HBS 7

1. Recall the problem of finding a set of undominated points. Given an input set of points $P = \{p_1, p_2, \ldots, p_n\}$ in the plane (with distinct coordinates), a point p is undominated if there are no other points above and to the right of p. Describe a method to maintain the set of undominated points as new points are added to the set. Specifically, describe and analyze a data structure that stores the undominated points of P, and an algorithm $\textsc{Insert}(x, y)$ that adds the point $p = (x, y)$ to P and returns $\textsc{True}$ or $\textsc{False}$ to indicate whether the set of undominated points has changed. Your data structure should use $O(n)$ space, and your $\textsc{Insert}$ algorithm should run in $O(\log n)$ amortized time.

2. Given a string of letters $Y = y_1 y_2 \ldots y_n$, a segmentation of Y is a partition of its letters into contiguous blocks of letters (also called words). Each word has a quality that can be computed by a given oracle (*e.g.* you can call $\textsc{quality}$("meet") to get the quality of the word "meet"). The quality of a segmentation is equal to the sum over the qualities of its words. Each call to the oracle takes linear time in terms of the argument; that is $\textsc{quality}(S)$ takes $O(|S|)$ time.

   Using the given oracle, give an algorithm that takes a string Y and computes a segmentation of maximum total quality.

3. Consider two distinct horizontal lines l_1 and l_2 in the plane. There are n points on l_1 with distinct x-coordinates $A = a_1, a_2, \ldots, a_n$ and there are n points on l_2 with distinct x-coordinates $B = b_1, b_2, \ldots, b_n$. Design an algorithm to compute, given A and B, a largest set S of non-intersecting line segments subject to the following restrictions:

   (a) Any segment in S connects a_i to b_i for some i $(1 \leq i \leq n)$.

   (b) Any two segments in S do not intersect.