CS 598TC, Spring 2017
Timothy Chan

Assignment 3 (due April 7 Friday 2pm (in class))

Note: Acknowledge any discussions you have with other students (and any references you have consulted). Solutions must be written entirely *in your own words*.

1. [*18 marks*] Given n points in 2D, where each point is assigned a *color*, we want a static data structure so that we can quickly report the colors of the points inside a query halfplane. Aim for an $O(n \log^c n)$-space structure with $O(\log^{c'} n + k)$ query time for some constants c and c', where k is the number of *distinct* colors in the query halfplane (not the number of points in the halfplane).

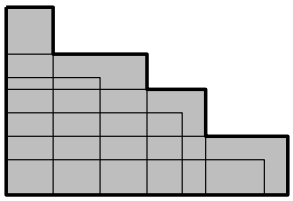
   [Hint: Work in dual space. Reduce to a problem about (uncolored!) line segments...]

   [Research Problem: what about 3D?]

2. [*14 marks*] We want a data structure to maintain a set S of n circular disks (possibly of different radii) in 2D under insertions and deletions so that at any moment, we can decide whether the disks in S are disjoint (i.e., nonoverlapping). Insertions could change the answer from "yes" to "no", and deletions could change the answer from "no" to "yes". Show that this problem can be solved with (amortized) update and query time $O(n^\alpha)$ for some constant $\alpha < 1$.

   [Hint: Sometimes it's easier to solve a more general problem (in this case, counting)... Reduce (static) disk intersection searching to halfspace range searching in some higher dimension by linearization.]

3. [*18 marks*] Given a set S of n rectangles in 2D all of which have the origin as their lower-left corners, the union of S forms a "staircase" polygon $U(S)$. We want to maintain the area of $U(S)$.

   Describe a fully dynamic data structure that supports insertions and deletions of rectangles in R in $O(\log^2 n)$ time, and can output the area of $U(S)$ in $O(1)$ time. Moreover, show how to output the area of the portion of $U(S)$ above any query horizontal line in $O(\log n)$ time.

   [Hint: Use an approach similar to Overmars and van Leeuwen's dynamic hull tree structure: divide by a near-median vertical line ℓ, and recurse on the left and the right. At each node, store the largest y-coordinate, the area of the portion of $U(S)$ left of ℓ and the area of the portion of $U(S)$ right of ℓ...]

   [Research Problem: what about 3D?]