CS 598TC, Spring 2017
Timothy Chan

Assignment 2 (due March 8 Wednesday 1pm (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. [*10 marks*] Prof. X has discovered a new method for 2D point location that has $O((\log n)^{1/4})$ query time and $O(n^4)$ space in the word RAM model. Describe how to further improve Prof. X's method to achieve $O((\log n)^{1/4})$ query time and $O(n)$ space, by using the random sampling approach from class.

2. [*15 marks*] Consider the following problem: store a set S of n *disjoint*, axis-aligned rectangular boxes in 3D so that for a given query point q, we can quickly find the box that contains q or report that none exists. (This can be viewed as an extension of 2D orthogonal point location.)

   (a) [*3 marks*] First show that in the special case when all boxes intersect a common vertical plane $x = m$, the problem can be solved in $O(n)$ space and $O(\log \log U)$ time, assuming that coordinates are integers bounded by U, by results from class.

   (b) [*12 marks*] Show that the general problem can be solved with $O(n)$ space and $O(\log n \log \log n)$ time.

   (Partial marks may be given for solutions with weaker bounds. It is open whether the general problem could be solved in near $O(\log \log U)$ time with near linear space.)

3. [*25 marks*] Let S be a set of n *possibly intersecting* line segments in 2D. We want to design a data structure for *vertical ray shooting*: given a query point q, find the line segment of S immediately above q.

   (a) [*5 marks*] First consider the following easier problem: Let S be a set of n lines (not line segments) in 2D. Design a data structure so that given a query vertical line segment q, we can report the lines of S intersecting q. Aim for $O(n)$ space and $O(n^\beta + k)$ query time (or better), where $\beta = \log_4 3$ and k is the output size.

   Hint: use duality and Willard's partition tree.

   (b) [*10 marks*] Now solve the vertical ray shooting problem for a set S of n lines in 2D, using $O(n)$ space and $O(n^\beta)$ query time (or better) by a randomized solution which is correct with good probability (say, at least 0.99).

   Hint: Recall that for a set Y of n numbers in 1D, a random sample $R \subset Y$ of size r divides the real line into $r + 1$ intervals each containing at most $O((n/r) \log r)$ elements, with good probability (at least 0.99).

   Bonus: Give an alternative deterministic solution that avoids randomization (and also avoids advanced techniques not covered in class).

(c) [*10 marks*] Finally solve the original vertical ray shooting problem for a set S of n line segments, using $O(n)$ space and $O(n^{\beta+\varepsilon})$ query time for an arbitrarily small constant $\varepsilon > 0$.

Hint: use (b) as a black box, in conjunction with a higher-degree segment tree.