CS 240: Data Structures and Data Management
Module 8 Study Guide

Taylor J. Smith — Spring 2017

Key Concepts

- **Multi-dimensional data** consists of a collection of d-dimensional items with d aspects/coordinates.
- For $d = 1$, performing a **range search** is straightforward.
  - 1D range search in an ordered array — $O(\log(n) + k)$
  - 1D range search in a binary search tree — $O(\log(n) + k)$
- For $d > 1$, however, these methods do not scale easily.
- Instead, we use new data structures (based on trees) to store and search higher-dimensional data.
- **Quadtrees** partition a region of 2D points into equally-sized quadrants.
  - Construction — $\Theta(nh)$
  - Range search — $\Theta(nh)$
- **kd-trees** partition a region of 2D points based on the median point of each partition.
  - Construction — $O(n \log(n))$
  - Range search — $O(n^{1-1/d} + k)$
- **Range trees** are a "tree of trees"; x- and y-coordinates of each 2D point are stored in their own tree
  - Construction — $O(n(\log(n))^{d-1})$
  - Range search — $O((\log(n))^d + k)$

Suggested Readings

- **Goodrich/Tamassia:** 12.1 (Range Trees), 12.3 (Quadtrees and k-D Trees)

Practice Questions

Goodrich/Tamassia

R-12.8. What would be the worst-case space usage of a range tree, if the primary structure were not required to have $O(\log(n))$ height?

R-12.12. What is the worst-case depth of a k-d tree defined on n points in the plane? What about in higher dimensions?

R-12.13. Suppose a set S contains n two-dimensional points whose coordinates are all integers in the range $[0..N]$. What is the worst-case depth of a quadtree defined on S?

R-12.14. Draw a quadtree for the following set of points, assuming a 16×16 bounding box:

$$\{(1,2),(4,10),(14,3),(6,6),(3,15),(2,2),(3,12),(9,4),(12,14)\}.$$

R-12.15. Construct a k-d tree for the point set of Exercise R-12.14.

C-12.2. Give a pseudocode description of an algorithm for constructing a range tree from a set of n points in the plane in $O(n\log(n))$ time.

C-12.4. Design a static data structure (which does not support insertions and deletions) that stores a two-dimensional set S of n points and can answer queries of the form $\textsc{CountAllInRange}(a,b,c,d)$ in $O(\log^2(n))$ time, which returns the number of points in S with x-coordinates in the range $[a..b]$ and y-coordinates in the range $[c..d]$. What is the space used by this structure?

C-12.5. Design a data structure for answering $\textsc{CountAllInRange}$ queries (as defined in the previous exercise) in $O(\log(n))$ time.

(*Hint:* Think of storing auxiliary structures at each node that are "linked" to the structures at neighbouring nodes.)

C-12.6. Show how to extend the two-dimensional range tree so as to answer d-dimensional range-searching queries in $O(\log^d(n))$ time for a set of d-dimensional points, where $d \geq 2$ is a constant.

(*Hint:* Design a recursive data structure that builds a d-dimensional structure using $(d-1)$-dimensional structures.)