# CSCI 5010: Exercise List 13

**Problem 1.** Describe an algorithm to build a kd-tree on $n$ points in $\mathbb{R}^2$ in $O(n \log n)$ time.
(Hint 1: it is easy to do $O(n \log^2 n)$ time. For improvement to $O(n \log n)$, prepare two sorted lists of the points at the beginning: one on the x-coordinate and the other on the y-coordinate. Then think about how to maintain the two lists incrementally *without* sorting as you build the levels of the tree in a top-down fashion.)
(Hint 2: the problem is trivial if you know how to find the median in linear time without sorting.)

**Problem 2. (Range Searching on Rectangles).** Let $S$ be a set of $n$ axis-parallel rectangles in $\mathbb{R}^2$. Given an axis-parallel rectangle $q$, a query reports all the rectangles $r \in S$ such that $r \cap q \neq \emptyset$. Describe a data structure of $O(n)$ size that answers such a query in $O(n^{3/4} + k)$ time, where $k$ is the number of rectangles reported.

**Problem 3.** Same problem as above, but give a structure with space consumption $O(n \log^3 n)$ and query time $O(\log^4 n + k)$.

**Problem 4 (Constrained Top-1 Search).** Let $S$ be a set of $n$ points in $\mathbb{R}^2$. A *constrained top-1 search* query specifies:

- real numbers $c_1, c_2$, and
- an axis-parallel rectangle $q$.

It returns a point $(x, y) \in S \cap q$ that maximizes the function $c_1 x + c_2 y$. Describe a data structure of $O(n \log^2 n)$ space that is able to answer any such query in $O(\log^3 n)$ time.