Due date: September 23, 2025

Problem 1: Let $\mathcal{R}$ be a set of n (possibly intersecting) orthogonal rectangles in $\mathbb{R}^2$. Describe an $O(n \log n)$ time algorithm to compute the area of the union of rectangles in $\mathcal{R}$.

Problem 2: Let $\mathcal{I}$ be a set of n intervals in $\mathbb{R}^1$. Given an arbitrary interval $\delta \subset \mathbb{R}^1$, an interval $I \in \mathcal{I}$ intersecting δ is called *short* (resp. *long*) if δ contains an endpoint of I (resp. $\delta \subset I$). Show that $\mathbb{R}^1$ can be partitioned, in $O(n \log n)$ time, into a family $\Delta = \{\delta_1, \ldots, \delta_m\}$ of intervals so that for each $i \leq m$, $||L_i| - |S_i|| \leq 1$, where L_i (resp. S_i) if the subset of intervals of $\mathcal{I}$ that are long (resp. short) in δ_i. Use this partition to build a linear-size data structure that for a query point $x \in \mathbb{R}^1$, returns the subset of all k intervals of $\mathcal{I}$ containing x in $O(\log n + k)$ time. Analyze the size, query time, and preprocessing time of your data structure.

Problem 3:

(i) Suppose that a data structure is needed that can answer triangular range queries, but only for triangles whose edges are horizontal, vertical, or have slope -1. Develop a linear-size data structure that answers such range queries in $O(n^{2/3} + k)$ time, where k is the number of points reported.

(ii) Describe an $O(n \log^2 n)$ size data structure for the above problem that answers a query in $O(\log^2 n + k)$ time.

Problem 4: Preprocess a set S of n horizontal segments in the plane so that the segments of S that intersect a vertical ray running from a point (q_x, q_y) vertically upwards towards infinity can be reported quickly. Describe a data structure for this problem that uses $O(n)$ storage and has a query time of $O(\log n + k)$, where k is the number of reported answers. (**Hint:** *Use persistent data structure.*)

Problem 5: Let $P = \{p_1, \ldots, p_n\}$ be a set of n points in $\mathbb{R}$. Each point $p_i \in P$ is assigned a color $\chi(p_i) \in \{1, \ldots, m\}$. Preprocess P in $O(n \log n)$ time into a data structure of $O(n)$ size so that for a query interval I, the colors of points in $I \cap P$ can be reported in $O(\log n + t)$ time, where t is the number of colors of these points. Note that $|I \cap P|$ can be much larger than t. (**Hint:** *Reduce to Problem 4.*)