Problem 3.1 [Thinking Outside The Box]. The orthogonal range query data structure described in Lecture 4 supports the following operation:

- \texttt{inside}(p, q): return all \( k \) points inside the bounding box spanned by points \( p \) and \( q \).

Implement the following operation using \( O(d) \) calls to \texttt{inside}():

- \texttt{outside}(p, q): return all \( k \) points outside the bounding box spanned by points \( p \) and \( q \).

Other than the black-box calls to \texttt{inside}(), the running time of your algorithm should be \( O(d + k) \).

**Solution:** The idea is to subdivide the space outside the query range into \( O(d) \) disjoint ranges, query each of them, and combine the results.

Without loss of generality, \( p_i < q_i \) for all \( i \).

For each dimension, we can partition the space into 3 parts: \( L_i = \{ x \in \mathbb{R}^d \mid x_i < p_i \} \), \( C_i = \{ x \in \mathbb{R}^d \mid p_i \leq x_i \leq q_i \} \), and \( R_i = \{ x \in \mathbb{R}^d \mid x_i > q_i \} \). Then we can write the exterior as

\[
\mathbb{R}^d \setminus (C_1 \cap \cdots \cap C_d) = L_1 \sqcup R_1 \\
\sqcup (C_1 \cap L_2) \sqcup (C_1 \cap R_2) \\
\sqcup (C_1 \cap C_2 \cap L_3) \sqcup (C_1 \cap C_2 \cap R_3) \\
\vdots \\
\sqcup (C_1 \cap \cdots \cap C_{d-1} \cap L_d) \sqcup (C_1 \cap \cdots \cap C_{d-1} \cap R_d).
\]

Each of these \( 2d \) terms is an orthogonal range. For example, \( C_1 \cap L_2 = [p_1, q_1] \times (-\infty, p_2) \times \mathbb{R}^{d-2} \).

We call \texttt{inside}() on each these ranges in order and concatenate the results.

Creating the first range takes \( O(d) \) time. Each successive range differs from the previous range in only \( O(1) \) coordinates, so we can update it in \( O(1) \) time. Concatenating the results takes \( O(k) \) time. So the total running time outside the \( 2d \) calls to \texttt{inside}() is \( O(d + k) \).