Smallest Range Covering Elements from K Lists [LeetCode](https://leetcode.com/problems/smallest-range-covering-elements-from-k-lists/description/)

You have k lists of sorted integers in **non-decreasing order**. Find the **smallest** range that includes at least one number from each of the k lists.

Example: [[4,10,15,24,26],[0,9,12,20],[5,18,22,30]]

Output: [20, 24]

**Approach 1: Function to find the smallest range covering elements from multiple arrays**

* **Functionality:**
  + Finds the smallest range covering elements from multiple arrays using a min heap.
* **Explanation:**
  + Initializes a min heap (**priority\_queue**) to keep track of the smallest element from each array.
  + Initializes **mini** and **maxi** to **INT\_MAX** and **INT\_MIN** respectively.
  + Pushes the first element from each array into the min heap, updating **mini** and **maxi** accordingly.
  + Continues processing until the min heap is not empty:
    - * Pops the minimum element from the heap (**temp**).
      * Updates **mini** with the popped element's value.
      * Checks if the range (**maxi - mini**) is smaller than the current smallest range (**end - start**). If so, updates **start** and **end**.
      * Checks if there are more elements in the same array. If yes, pushes the next element from the same array into the heap.
  + Returns the smallest range as a pair (**start**, **end**).
* **Time Complexity:**
  + **Overall Time Complexity: O(N log K), where N is the total number of elements across all arrays, and K is the number of arrays.**
* **Space Complexity:**
  + **O(K) - The min heap stores at most one element from each array.**