Abhinav and Lists

Assignment 5
Data Structures & Algorithms

Problem Statement: Abhinav has N lists of integers of varying lengths containing numbers from 1 to K. He wants to find out that whether the sequence in which the lists are given to him follow the given rule
$list_i <= list_{i+1}$
where lists are compared on the basis of the priority of the corresponding elements, if an element is same in both lists, we move on to compare the next corresponding elements.

For example, consider the lists [3, 1, 2] and [2, 3, 1], in this case if we keep our priority order as (3, 1, 2) or (3, 2, 1), it would satisfy our answer, as in both cases 3 is in lower priority than 2. So if we would compare the first element of both lists, $list_1 < list_2$.

As Abhinav is very busy with his research, help him find out such a priority order that would satisfy the above conditions.

Input
First line contains 2 integers: N and K.
Then $2N$ lines follow, the first containing an integer l_i denoting the length of the array and the subsequent line contains the array.

Output
Output k integers the lexicographically smallest sequence in which the numbers should be considered in increasing order of their priority such that the above conditions hold. Output "-1"(without quotes) if not possible.

Constraints
$2 \leq N \leq 10^5$
$1 \leq K \leq 10^5$
$\sum_{i=1}^{i=N} l_i \leq 10^6$
Time Limit: 4 seconds
Memory Limit: 256 MB

Sample Test Case 1

Input	Output
4 6	2 3 4 5 1 6
3	
4 6 3	
3	
5 1 2	
2	
1 2	
2	
1 1	

Explanation As we can see from the output that 4 is lower in priority than 5, hence $list_1 < list_2$. Now, 5 is in lower priority than 1, so $list_2 < list_3$. For $list_3$ and $list_4$, the first element is same, so we move on to the second element and as the priority of 2 is less than 1, hence $list_3 < list_4$.

So, this priority order satisfies our above constraints and also is the lexicographically smallest.
Sample Test Case 2

Input	Output
3 5	-1
3	
1 2 3	
3	
2 3 4	
3	
1 4 5	

Explanation So, for $list_1$ to be lesser than $list_2$, we need to have 1 in lower priority than 2 but for $list_2$ to be lesser than $list_3$ we need 2 to be in lower priority than 1. Both of the conditions are not possible simultaneously, hence output is -1.