

Problem I

Lost Layers

Time limit: 2s

Memory limit: 256MB

Linh is restoring a video project whose file was damaged. The project has n clips, numbered from 1 to n . Every clip lies on its own layer, and the layers are stacked one above another: when several clips are on screen at the same frame, a clip on a higher layer covers the clips below it. Clip i is on screen at frames $l_i, l_i + 1, \dots, r_i$, including both l_i and r_i .

The frame ranges of the clips survived, but the order of the layers was lost. Linh still has the log of the preview renderer. Each line of the log is a snapshot taken at some frame x . It names the clips that are on screen at frame x , starting from the top layer and going down, but it stops after the first k of them. The value of k may differ from snapshot to snapshot. It can be 0, and it can be equal to the number of clips on screen, in which case all of them are named.

Help Linh find an order of all n layers, from top to bottom, that agrees with every snapshot. An order agrees with a snapshot $(x, k, s_1, \dots, s_k)$ if the following holds: take the clips i with $l_i \leq x \leq r_i$ and sort them from top to bottom according to the order; then the first k of them are $s_1, s_2, \dots, s_k$, in this order. In particular, if $k \geq 1$, every clip on screen at frame x that is not among $s_1, \dots, s_k$ must be below clip s_k . A snapshot with $k = 0$ places no constraint on the order.

Input

- The first line contains a single integer n ($1 \leq n \leq 3 \cdot 10^5$), the number of clips.
- The i -th of the next n lines contains two integers l_i and r_i ($0 \leq l_i < r_i \leq 10^9$), the first and the last frame of clip i .
- The next line contains a single integer m ($1 \leq m \leq 3 \cdot 10^5$), the number of snapshots.
- Each of the next m lines describes one snapshot. It starts with two integers x and k ($0 \leq x \leq 10^9$, $0 \leq k \leq n$), the frame of the snapshot and the number of clips it names. They are followed by k integers $s_1, s_2, \dots, s_k$ ($1 \leq s_j \leq n$), the named clips from top to bottom. The clips $s_1, \dots, s_k$ are pairwise distinct, and each of them is on screen at frame x . If $k = 0$, the line contains only x and k .

The snapshots are given in non-decreasing order of x ; several snapshots may be taken at the same frame. The sum of k over all snapshots does not exceed 10^6 .

It is guaranteed that at least one order agrees with all snapshots.

Output

Print n distinct integers $p_1, p_2, \dots, p_n$: the numbers of the clips from the top layer to the bottom layer. If several orders agree with all snapshots, print any of them.

Sample Input	Sample Output
5 2 6 0 4 4 9 6 8 1 10 4 1 1 2 4 1 3 6 1 4 9 2 3 5	4 3 2 5 1
4 0 5 5 10 3 7 20 30 4 0 0 5 1 3 5 2 3 2 12 0	3 2 4 1

Notes

In the first example, clips 2 and 5 are on screen at frame 1, so clip 2 is above clip 5. At frame 4, clips 1, 2, 3 and 5 are on screen (clip 2 ends and clip 3 starts at this frame), and clip 3 is the top one. At frame 6, clips 1, 3, 4 and 5 are on screen (clip 1 ends and clip 4 starts at this frame), and clip 4 is the top one. At frame 9, only clips 3 and 5 are on screen, and the snapshot names both of them. Besides the order in the example output, the orders 4 3 1 2 5 and 4 3 2 1 5 are also accepted.

In the second example, the snapshots at frames 0 and 12 name no clips, and the two snapshots at frame 5 see clips 1, 2 and 3. Together they say that clip 3 is above clip 2, and clip 2 is above clip 1. No snapshot sees clip 4, so it can be anywhere in the order.