

7817 IOI 2017 Logo

Iran is the host of the International Olympiad in Informatics (IOI) 2017. In order to design the IOI 2017 logo, the organizing committee of the IOI 2017 decided to publicly call for logos. It was not surprising that many logos were received in a short time as the young generation in Iran is actively taking part in any national event. In the first round, logos were judged by some professional graphic designers, and the best logos being artistically capable to be the IOI 2017 logo were selected to be judged in the second round.

The selected logos are now presented to the organizing-committee members for voting. The voting system is a little bit complicated: each member can vote for at most three different logos in some order. The first, second and third choices of each member are awarded 3, 2 and 1 points, respectively. The score of a logo is the total points the logo receives from all members. The logo with the highest score is the winner. In the case of ties, the winner is the logo with higher number of first votes. Again, if some logos have the same score and first votes, the logo with more second votes is the winner. If we still have ties, all of them would be winners. Given the voting information, your job is to identify the winner logo (or logos).



Input

There are multiple test cases in the input. The first line of each test case contains a positive integer n denoting the number of voters ($1 \leq n \leq 100$). Each of the next n lines starts with an integer d_i , representing the number of logos chosen by the i -th voter ($1 \leq d_i \leq 3$), followed by d_i different logo IDs showing the choices of that voter (from left to right). Each logo ID is a positive integer not exceeding 10^6 . All integers in a line are separated with a single space.

The input terminates with a line containing '0' which should not be processed.

Output

For each test case, output a line containing the winner logos in the increasing order of their IDs. Logo IDs in a line must be separated with a single space.

Sample Input

```
4
3 5 2 1
3 12 5 2
2 1 2
3 2 1 5
2
3 3 2 1
3 2 3 1
0
```

Sample Output

2

2 3