A. World Football Cup

Everyone knows that 2010 FIFA World Cup is being held in South Africa now. By the decision of BFA (Berland's Football Association) next World Cup will be held in Berland. BFA took the decision to change some World Cup regulations:

- the final tournament features n teams (n is always even)
- the first $n / 2$ teams (according to the standings) come through to the knockout stage
- the standings are made on the following principle: for a victory a team gets 3 points, for a draw — 1 point, for a defeat — 0 points. In the first place, teams are ordered in the standings in decreasing order of their points; in the second place — in decreasing order of the difference between scored and missed goals; in the third place — in the decreasing order of scored goals
- it's written in Berland's Constitution that the previous regulation helps to order the teams without ambiguity.

You are asked to write a program that, by the given list of the competing teams and the results of all the matches, will find the list of teams that managed to get through to the knockout stage.

Input

The first input line contains the only integer n ($1 \le n \le 50$) — amount of the teams, taking part in the final tournament of World Cup. The following n lines contain the names of these teams, a name is a string of lower-case and upper-case Latin letters, its length doesn't exceed 30 characters. The following $n \cdot (n - 1) / 2$ lines describe the held matches in the format `name1-name2 num1:num2`, where $name1$, $name2$ — names of the teams; $num1$, $num2$ ($0 \le num1, num2 \le 100$) — amount of the goals, scored by the corresponding teams. Accuracy of the descriptions is guaranteed: there are no two team names coinciding accurate to the letters' case; there is no match, where a team plays with itself; each match is met in the descriptions only once.

Output

Output $n / 2$ lines — names of the teams, which managed to get through to the knockout stage in lexicographical order. Output each name in a separate line. No odd characters (including spaces) are allowed. It's guaranteed that the described regulations help to order the teams without ambiguity.

Examples

input

```
4
A
B
C
D
A-B 1:1
A-C 2:2
A-D 1:0
B-C 1:0
B-D 0:3
C-D 0:3
```

output

```
A
D
```

input

```
2
a
```

```
A
a-A 2:1
```

output

```
a
```