

---

## Groups of Interest

X51913\_en

We want to configure the tables for a wedding. Every participant provides a list of topics of his/her interest. We assign a table such that everybody in the table shares (at least) one common topic.



### Input

The input is a text (in the stdin) with the number of tables and a list of people and their topics of interest. For instance, the text:

```
2
Maria  politics SAT AI
John   art soccer maths
Peter   soccer art
Andrew  bascket politics AI
Sara    art SAT maths
Holly   politics bascket
```

### Output

The output is also a text (in the stdout) where, in every line, there is a list of components of one of the tables. In this example:

```
{ Sara John Peter }
{ Andrew Maria Holly }
```

Notice that the order of the lines and the order inside each line is not relevant.

If the problem has no solution, the program must print the text "No solution".

All tables have the same size (number of people). If the total number of people is not divisible by the number of tables, then all of them must have approximately the same number of participants. In other words, if  $n$  is the number of people and  $m$  is the number of tables, in every table we must seat either  $\lfloor n/m \rfloor$  or  $\lceil n/m \rceil$  people.

**Sample input 1**

```
1
Maria  politics SAT AI
John   art   soccer maths
Peter  soccer art
Andrew bascket politics AI
Sara   art   SAT maths
Holly  politics bascket
```

**Sample output 1**

```
No solution
```

**Sample input 2**

```
2
Maria  politics SAT AI
John   art   soccer maths
Peter  soccer art
Andrew bascket politics AI
Sara   art   SAT maths
Holly  politics bascket
```

**Sample output 2**

```
{ Maria Holly Andrew }
{ Peter John Sara }
```

**Sample input 3**

```
3
Maria  politics SAT AI
John   art   soccer maths
Peter  soccer art
Andrew bascket politics AI
Sara   art   SAT maths
Holly  politics bascket
```

**Sample output 3**

```
{ Peter John }
{ Andrew Holly }
{ Maria Sara }
```

**Sample input 4**

```
3
Maria  politics SAT
John   art   SAT
Peter  soccer maths
Andrew soccer AI
Sara   soccer AI
Holly  art   bascket
Eva    politics SAT
```

**Sample output 4**

```
{ John Holly }
{ Eva Maria }
{ Peter Andrew Sara }
```

## Scoring

Samples have been selected in order to ensure that there exists a unique solution up to people and tables permutations.

In the solution of this problem, you should use a function implemented for solving the "cardinality constraints" problem.

You will get 5 if you can solve the basic version of the problem where the number of people is divisible by the number of tables. You will get 5 additional points if you can also solve the case where the tables are not all of the same size (but their size only differs in one).

## Problem information

Author: Jordi Levy

Generation: 2026-01-25T16:44:47.997Z

© Jutge.org, 2006–2026.

<https://jutge.org>