Objective

Preparing cocktails (without alcohol) is quite an art. An art in which you do not do so badly, but for your talent to be recognized at its true value and to get the prestigious title of "Best Bartender of France" you still have to train.

You ask yourself how to automate the preparation of your drinks a little and you realize that you always waste a lot of time thinking about in which order to mix the ingredients. If you were able to learn once and for all the chaining of ingredients, for sure you would be more efficient (and a more effective bartender is a better bartender).

You know a little programming (you're not the best either), and you want to generate the list of ingredients needed to prepare a cocktail in an order that respects the basic rules of a drink, namely that certain ingredients should be added to the preparation before others (e.g. sugar before tonic).

And hop, all you have to do is shake the shaker vigorously for about ten seconds.

Data format

Input

Row 1: two integers separated by a space, the first, **N,** between *1* and *149* describing the number of ingredients at most for this preparation, the second, **M,** between *1* and **3N** describing the number of constraints in the cocktail recipe.

Rows 2 to **M** + 1: two chains comprising between *1* and *128* characters in lowercase **A** and **B** that represent the name of two ingredients. A row means that

A must be mixed with the mixture before **B.** Ingredient names may also include character -.

Output

A list of ingredients separated by ' > '(with a space before and after the sign >) describing an order in which you have to mix the ingredients to make the wonderful beverage, if such an order does not exist, return *KO*.

Note: There are several solutions, you can return any.

Example

Input

```
5 4
club-mate pamplemousse
pamplemousse grenadine
mojito club-mate
fraise club-mate
```

Output

```
mojito < fraise < club-mate < pamplemousse < grenadine
```

Input

```
3 3
citron avocat
courgette citron
avocat courgette
```

Output

```
KO
```