CSULA PROGFEST 2014
Problem 3
Lazy Days

Eddie is a sophomore studying computer science, and is failing his discrete mathematics course. It's not that he doesn't understand the concepts: it's just that he's too lazy, or too smart(so he claims), to do homework. Staring at the chapter homework for functions and relations, he realized that he could write a program to do the homework for him. He learned the following definitions in class:

Definition 1. *A **relation** from a set A to a set B is a set of ordered pairs (a, b) where a is an element of A and b is an element of B. i.e.* $\{(-4, 1), (-2, 3), (0, -5)\}$*, set A is* $\{-4, -2, 0\}$ *and set B is* $\{1, 3, -5\}$ *The set A is the **domain** of the relation and set B is its **codomain**.*

Definition 2. *A **function** is a relation that satisfies:*

- *for each element a in the domain, there is an element b in its codomain such that (a, b) is in the relation, and*

- *if (a, b) and (a, c) are in the relation, then b = c.*

Definition 3. ***one-to-one**: A function is said to be one-to-one, if and only if each element in the codomain corresponds to no more than one element in the domain.*

Definition 4. ***onto**: A function is said to be onto, if and only if each element of the codomain of the function has a corresponding element in the domain.*

Definition 5. ***bijective**: A function is said to be bijective if and only if it is both one-to-one and onto.*

Your job is to design a program that, given a domain and codomain and a list of relations using them, determine what type of functions the relations are, if they are functions at all.

Input

The first line will contain a list of D integers separated by commas, denoting the domain for all relations to be tested, $1 \le D \le 100$. The second line will contain a list of C integers separated by commas, denoting the codomain for all relations to be tested, $1 \le C \le 100$. The third line will contain a single integer, denoting the number of relations to be tested. Each relation consists of one line, denoting the relation in set notation.

Output

For each relation, print, on a single line whether it's "not a function" or "function". If it is a function, print a single comma after "function", followed by either "one-to-one", "onto", "bijective", or nothing at all if it doesn't fall into any of the categories.

Sample Input

```
1,2,6,7,8,9
1,2,3,4,5
4
{(1 1),(1 2),(2 3),(6 5),(7 4)}
{(1 1),(2 2),(6 3),(7 4),(8 5)}
{(1 4),(2 4),(6 4),(7 4),(8 4)}
{(1 1),(2 2),(6 3),(7 5),(8 5),(9 4)}
```

Sample Output

```
not a function
function,bijective
function
function,onto
```