Problem L: Lost Logic

Time limit: 1 s
Memory limit: 512 MiB

Gustav is reading about *2-satisfiability*, a well known problem of assigning truth values to boolean variables in order to satisfy a list of *constraints* — simple logical formulas involving two variables each.

We are using n variables $x_1, x_2, \ldots, x_n$ that can take on values 0 (false) and 1 (true). A constraint is a formula of the form $a \to b$ where both a and b are possibly negated variables. As usual, $\to$ denotes logical implication: $a \to b$ is 0 only when a is 1 and b is 0. The negation of variable a is denoted by $!a$.

Given an assignment of values to variables, we say that the constraint is *satisfied* if it evaluates to 1. Gustav has constructed a list of constraints and correctly concluded that there are *exactly three* different assignments of values to variables that satisfy all the constraints. Gustav has wrote down all three assignments but has, unfortunately, lost the list of constraints.

Given three assignments of n values to variables, find any list consisting of at most 500 constrains such that the three given assignments are the *only* assignments that satisfy all the constraints.

Input

The first line contains an integer n ($2 \leq n \leq 50$) — the number of variables. Three lines follow, each describing one assignment. The k-th line contains n space-separated integers $v_1^k, v_2^k, \ldots, v_n^k$, where each v_i^k is either 0 or 1 and denotes the value of the variable x_i in the k-th assignment. All three assignments will be different.

Output

If there is no solution output a single line containing the integer -1.

Otherwise, the first line should contain an integer m where $1 \leq m \leq 500$ — the number of constraints in your solution. The k-th of the following m lines should contain the k-th constraint. Each constraint should be a string constructed according to the following rules:

- A *variable* is a string of the form "xi" where i is an integer between 1 and n inclusive written without leading zeros.
- A *literal* is a string consisting of a variable possibly preceded by the "`!`" character.
- A *constraint* is a string of the form "a `->` b" where both a and b are literals. The implication sign consists of the "minus" character and the "greater-than" character and there is a single space character both before and after the implication sign.

Example

input	output
3 0 0 0 0 1 0 1 0 0	3 x1 -> !x2 x3 -> x1 x3 -> x2

input	output
4 0 0 1 0 1 0 0 0 1 0 1 1	-1