

Alice and Bob use a key, a code, to communicate in privacy. This code is just an ordered sequence of m different characters out of a set S of n characters. For example, if $S = \{1, 2, 3, 4, 5, 6, 7, 8, 9\}$ and $m = 4$, then one possible key is $(3, 8, 1, 5)$.

No one knows the key except Alice and Bob. However, Steve the Spy knows the set S and length m of the key, and he will try to deduce the key by posing different guesses of the code. To each guess, Alice or Bob will reply with an answer of the form (c, w) , where c is the number of correctly placed digits in Steve's guess, and w is the number of digits in the guess which are present in the code but in the wrong position, i.e., they are misplaced.

For instance, these are guesses and answers for the aforementioned code.

```
(1,1,1,1) -> (1,0)
(2,2,2,2) -> (0,0)
(9,8,3,1) -> (1,2)
```

Note that a correct guess will produce the answer $(m, 0)$. You will be given S , m and a list (of arbitrary length) of trials and answers, and your program has to take either one of the following courses of action:

1. claim that the given information is not enough to find out the correct code, or
2. give the correct code, i.e., the code for which the answer is $(m, 0)$.

Input

The input file contains several test cases, each of them as described below. Note that n is less than or equal to 10, and m is less than or equal to n .

First line of input has the set S with its elements separated by blank spaces (without brackets). Second line has the value of m . The following lines are the guesses, in the form (g_1, \dots, g_m) with $g_i \in S$ for all $1 \leq i \leq m$, the '-' symbol and the answer, in the form (c, w) . Each case ends with a line with a single '%' character.

Output

For each test case, output either the solution in the form (s_1, \dots, s_m) with $s_i \in S$ for all $1 \leq i \leq m$ or the character 'N' to denote the impossibility of finding a solution, on a line by itself.

Sample Input

```
1 2 3 4 5 6 7 8 9
3
(1,1,1) -> (0,0)
(2,2,2) -> (0,0)
(3,3,3) -> (0,0)
(4,4,4) -> (1,0)
(5,5,5) -> (0,0)
(6,6,6) -> (0,0)
(7,7,7) -> (1,0)
(8,8,8) -> (1,0)
(4,7,8) -> (1,2)
(4,8,7) -> (0,3)
(8,7,4) -> (0,3)
(7,8,4) -> (1,2)
(8,4,7) -> (1,2)
%
1 2 3 4 5 6 7 8 9
3
(1,1,1) -> (0,0)
(2,2,2) -> (0,0)
(3,3,3) -> (0,0)
(4,4,4) -> (1,0)
(5,5,5) -> (0,0)
(6,6,6) -> (0,0)
(7,7,7) -> (1,0)
(8,8,8) -> (1,0)
(9,9,9) -> (0,0)
%
```

Sample Output

```
(7,4,8)
N
```