COCI '14 Contest 2 #1 Mobitel

Time limit: 1.0s **Memory limit:** 32M

Grasshopper Marko was jumping happily all over the meadow. He wasn't being careful and his Nokia 3310 fell into a puddle. His mobile phone is now acting funny! The contacts got wet and the keyboard works in a completely unpredictable manner! All the numerical keys broke down. When we press one of them, the mobile phone acts as if we pressed **another key**. Luckily, there are **no two keys** that are **acting the same** so Marko can still **write all the letters**.

Grasshopper Marko was experimenting a bit and found out how each key acts. Now he wants to write a message to his girlfriend. Since he is just a grasshopper, you will do that for him.

To all those who don't remember how mobile phones with keys work, here is a short description.

1	2 abc	3 def
4 ghi	5 jkl	6 mno
7 pqrs	8 tuv	9 wxyz
*	0	#

Keyboard on a very old mobile phone.

The image shows keys with letters that we can get by pressing that key (on a working mobile phone that didn't fall into a puddle). For example, if we want to get letter `a` we will press key `2` once, and if we want letter `b` we will press the key `2` twice. If we want to write two letters from the same key consecutively, we have to press the pound key (`#`) **exactly once**. For example, if we want to write the string `klor` we will press the keys in the following order: `55#555666777`

Input Specification

The first line of input contains 9 integers. The first integer marks the behaviour of key `1`, the second the behaviour of key `2`, the third the behaviour of key `3`, and so on.

Marko is not using keys `*` and `0` because he is a grasshopper. Key `#` can't get broken.

The second line of input contains a string consisting of only lowercase letters of the English alphabet. The length of the word won't exceed 100 characters.

Output Specification

The first and only line of output must contain the sequence of keys you need to press in order to write Marko's message.

Sample Input 1

```
2 3 4 5 6 7 8 9 1
klor
```

Sample Output 1

```
44#444555666
```

Explanation of Output for Sample Input 1

All of the keys are shifted one place to the right so the output differs a little bit from the example in the task statement.

Sample Input 2

```
7 8 9 1 2 3 6 5 4
djevojka
```

Sample Output 2

```
68662227778#885
```

Sample Input 3

```
9 8 7 6 5 4 3 2 1
skakavac
```

Sample Output 3

```
33335585582228#888
```