**Q. Transformation of Word**

We introduce a function, denoted by P such that P: {1,2,3...n} -----> {1,2,3...n}. Foe ex, if P (1) = 2, P (2) = 3, ... P(n-1) =n, p(n)=1; then P is represented by a sequence: 2,3,4...1. 2 that appears in the first position indicates that, the image of 1 is 2, 1 appears in the last position informs that the image of n is 1, and so on. For a set with n elements, n! functions are possible. We can apply this function P on a word, to transform the word.

Given a word abcd, and a function P: 2,3,4,1 The function transforms the word as follows:

Since the image of 1 is 2, the symbols in the first position moves to the second position. Since the image of 2 is 3. the symbol in the second position moves to the third position. Since the image of 4 is 1, the symbol in the fourth position moves to the first position. Hence P(abcd) = dabc, where P = 2,3,4,1.

If P= 2,1,3, P(abd) = bad. The symbol in the first position moves to the second position, the symbol in the second position moves to the first position and the symbol in the third position remains in the third position. A function P that acts upon a word of length n, should have a sequence of n numbers, separated by a comma. If we apply the function: 2,1,3 to bad, we get the original word abd.

Given a word w and a function P, write a code to compute P(w) and to compute the function which will transform P(w) to w.

Input Format:

Enter the word of length n

Enter the first digit of the function

Enter the second digit of the function.

...

Enter the n-th digit of the function

Output Format:

Print the word P(w)

The next n-lines should print the digits of the function that transforms P(w) to w.

Illustration:

Input:

abc

2

3

1

Output:

cab

3

1

2

**Python Program**

s = input()  
s\_new = **""**l = []  
dic = {}  
for counter in range(len(s)):  
 d = int(input())  
 l.append([s[counter], d])  
 dic[s[counter]] = counter+1  
for i in range(len(l)):  
 small = i  
 for j in range(i + 1, len(l)):  
 if l[small][1] > l[j][1]:  
 small = j  
 l[i], l[small] = l[small], l[i]  
for counter in range(len(l)):  
 s\_new += l[counter][0]  
print(s\_new)  
for counter in range(len(s\_new)):  
 print(dic[s\_new[counter]])