Vigenère Cipher Encryption

`vigenere.c`, `vigenere.C`, `vigenere.java`

One of the oldest and most common encryption algorithms is *Vigenère Cipher*. It is quite an old thing — a similar encryption was first described in 1553 by Giovan Battista Bellaso and improved in 1586 by Blaise de Vigenère.

Vigenère encryption produces a single letter of ciphertext for each letter of plaintext, combining one plaintext letter with one single letter of a *key* on the corresponding position. If the key is shorter than the plaintext, it is simply repeated as needed, e.g. for a key of length 3 and plaintext of length 7, letters will be combined like this (K_i is the key letter, P_i is the plaintext letter, and C_i is the resulting ciphertext letter).

$$\begin{array}{ccccccc} K_1 & K_2 & K_3 & K_1 & K_2 & K_3 & K_1 \\ P_1 & P_2 & P_3 & P_4 & P_5 & P_6 & P_7 \\ \hline C_1 & C_2 & C_3 & C_4 & C_5 & C_6 & C_7 \end{array}$$

The letter of the key specifies how many positions should be the plaintext letter "shifted forward" in the alphabet. If a key letter is A, the correspoding plaintext letter will be shifted by one character, B means two positions, etc. The alphabet is considered circular, so if the last letter (Z) should be shifted, it becomes A again. Please note that A (key) combined with another A (plaintext) will result in B, which may be a little unusual for the common Vigenère cipher. The Vigenère square at the end of this problem statement gives an overview how letters of a plaintext get combined with letters of a key to produce the ciphertext.

Your task is to write a program that will encrypt messages using the Vigenère cipher with a given key.

Input Specification

The input contains several instances. Each instance consists of two lines, the first line is the encryption key and the second line is the plaintext. Both key and plaintext consist of uppercase letters of the English alphabet $\{A, B, C, ..., Z\}$. The length of the key will be between 1 and 1000, the length of the plaintext between 1 and 100 000, inclusive.

Input is terminated by a line containing one zero.

Output Specification

For each input instance, output the ciphertext — the encrypted version of the message.

Sample Input

```
ICPC
THISISSECRETMESSAGE
ACM
CENTRALEUROPEPROGRAMMINGCONTEST
LONGKEY
CERC
0
```

Output for Sample Input

```
CKYVRVIHLUUWVHIVJJU
DHAUUNMHHSRCFSEPJEBPZJQTDRAUHFU
OTFJ
```

	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z
A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A
B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B
C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C
D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D
E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E
F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F
G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G
H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H
I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I
J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J
K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K
L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L
M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M
N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N
O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O
P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P
Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q
R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R
S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S
T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T
U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U
V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V
W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W
X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X
Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y
Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z

Vigenère square:
Mapping a given plaintext letter (column) and a key letter (row) to the resulting ciphertext letter