

The Playfair cipher is a manual symmetric encryption technique and was the first digraph substitution cipher. The scheme was invented in 1854 by Charles Wheatstone, but bears the name of Lord Playfair who promoted the use of the cipher.

The Playfair cipher uses a 5 by 5 table containing each letter in the English alphabet exactly once (except 'Q' which is missing). The table constitutes the encryption key. To more easily remember the table, it is typically generated from a key phrase. First fill in the spaces in an empty table with the letters of the key phrase (dropping spaces and duplicate letters), then fill the remaining spaces with the rest of the letters of the alphabet in order. The key phrase is written in the top rows of the table, from left to right. For instance, if the key phrase is "playfair example", the encryption key becomes

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

To encrypt a message, one would remove all spaces and then break the message into digraphs (groups of 2 letters) such that, for example, "Hello World" becomes "HE LL OW OR LD". Then map them out on the key table, and apply the rule below that matches the letter combination:

- If both letters are the same (or only one letter is left), add an 'X' after the first letter. Encrypt the new pair and continue (note that this changes all the remaining digraphs).
- If the letters appear on the same row of your table, replace them with the letters to their immediate right respectively (wrapping around to the left side of the row if a letter in the original pair was on the right side of the row). With the table above, the digraph 'CH' would be encrypted 'DB'.
- If the letters appear on the same column of your table, replace them with the letters immediately below respectively (wrapping around to the top side of the column if a letter in the original pair was on the bottom side of the column). With the table above, the digraph 'VA' would be encrypted 'AE'.
- If the letters are not on the same row or column, replace them with the letters on the same row respectively but at the other pair of corners of the rectangle defined by the original pair. The order is important – the first letter of the encrypted pair is the one that lies on the same row as the first letter of the plaintext pair. With the table above, the digraph 'KM' would be encrypted 'SR'.

Write a program that reads a key phrase and a plaintext to encrypt, and outputs the encrypted text.

The text to encrypt will not contain two 'x's following each other, or an 'x' as the last character, as this might cause the first rule above to repeat itself indefinitely.

Input

The first line of the input file contains an integer N ($N < 25$) which denotes the total number of test cases. The description of each test case is given below:

The input for each test case is given in two lines. The first line contains the key phrase. The second line contains the text to encrypt. Each line will contain between 1 and 1000 characters, inclusive. Each character will be a lower case English letter, 'a' - 'z' (except 'q'), or a space character. Neither line will start or end with a space.

Output

For each test case the output should be a single line containing the encrypted text, in upper case. There should be no spaces in the output. Look at the output for sample input for details.

Sample Input

```
2
playfair example
hide the gold in the tree stump
the magic key
i love programming competition
```

Sample Output

```
BMNDZBXDKYBEJVDMUIXMMNUVIF
YDVHCWSPKNTAHKUBIPERMHGHDVRU
```