

H Hidden Code

It's time to put your hacking skills to the test! You've been called upon to help crack enemy codes in the current war on... something or another. Anyway, the point is that you have discovered the encryption technique used by the enemy; it is quite simple, and proceeds as follows. Note that all strings contain only uppercase letters of the alphabet.

1. We are given a key K and a plaintext P which is encrypted character-by-character to produce a ciphertext C of the same length.
2. If $|K|$ is the length of the key K , then the first $|K|$ characters of C are obtained by adding the first $|K|$ characters of P to the characters of K , where adding two letters means interpreting them as numbers ($A = 0$, $B = 1$, and so on) and taking the sum modulo 26. That is, $C_i = (P_i + K_i) \bmod 26$ for $i = 1, \dots, |K|$. If $|K| > |P|$, then the extra characters in K are ignored.
3. The remaining characters of P , i.e. P_i for $i > |K|$, are encrypted using the previous ciphertext characters by $C_i = (P_i + C_{i-|K|}) \bmod 26$ for $i = |K| + 1, \dots, |P|$.

As an example, consider the encryption of the string "STANFORD" using the key "ACM":

```
STA  NFORD
+ ACM SVMFA
-----
SVM  FAAWD
```

Knowing this, you are well on your way to being able to read the enemy's communications. Luckily, you also have several pairs of plaintexts and ciphertexts which your team recovered, all of which are known to be encrypted with the same key. Help find the key that the enemy is using.

H.1 Input

The input consists of multiple test cases. Each test case begins with a line containing a single integer N , $1 \leq N \leq 100$, the number of plaintext and ciphertext pairs you will receive. The next N lines each contain two strings P and C , the plaintext and ciphertext, respectively. P and C will contain only uppercase letters (A-Z) and have the same length (at most 100 characters). The input terminates with a line with $N = 0$. For example:

```
1
A B
3
STANFORD SVMFAAWD
AVOWIENR AXAWFEJW
VAMRI VCYMK
3
ABCDEFGHIJKLMNPOQRSTUVWXYZ AAAAAAAAAAAAAAAAAAAAAA
Y Y
Z Z
2
A B
B A
0
```

H.2 Output

For each test case, print a single line that contains the shortest possible key or “Impossible” (quotes added for clarity) if no possible key could have produced all of the encryptions. For example, the correct output for the sample input above would be:

```
B
ACM
AZYXWVUTSRQPONMLKJIHGFEDCB
Impossible
```