2003/2004 SOUTHERN CALIFORNIA REGIONAL
ACM INTERNATIONAL COLLEGIATE PROGRAMMING CONTEST

Problem 5
Playfair

Swamp County Publishing is producing a book on the history of codes and ciphers. You have been hired to produce the CD. Today's job is the Playfair cipher. Here are the author's preliminary notes for it.

The Playfair cipher is an example of polygram encipherment: it replaces pairs of characters with other pairs. It is based on a rectangular array usually determined by an agreed upon passphrase or standard text. For this example we will use a 6 by 6 array of the alphanumeric characters. Here is the example we will use in the following discussion:

```
G7UMWS
LBDNPA
YTHE4R
2OVCI3
91Z0KF
865QXJ
```

The basic scheme is to locate a pair in the array, say WZ and consider them to be at the diagonally opposite corners of a rectangular sub-array:

```
UMW
DNP
HE4
VCI
ZOK
```

For the enciphered pair, take the pair at the other corners. For this example, for each character, choose the one on the same row. Thus WZ becomes UK. However, there are several other cases to take care of:

- If the characters are in the same column, choose the character below each character: B1 will become T6. If a character is in the last row, use the character in the first row: MQ will become NM.
- Similarly, if the characters are in the same row, choose the character to the right of each: T4 will become HR. If a character is in the last column, use the character in the first column: PA will become AL.
- Finally, if a letter is doubled, use the character to its right: HH will become EE. As before, if it is in the last column, use the one in the first column: FF will become 99.

There are many variations possible, each of the rules may go up, down, right or left. Use the above rules for the CD example. Deciphering is the obvious inverse of these operations.

Here is the scheme for constructing the array from the passphrase, in this case:

```
swamp run: AM437.
```

Copy the passphrase, keeping only alphanumeric characters, converting lower case to upper case, and keeping only the first occurrence of a character (ignoring case).

```
SWAMPRUN437
```

Of course a real passphrase would be longer. But, in general, we may need to add more characters to include all the alphanumeric characters. Write down the missing characters, in the order A..Z0..9, in columns under the converted passphrase, using more than one line if necessary:

```
SWAMPRUN437
BCDEFGHIJKL
OQTVXYZ0125
689
```

Next sort the columns by the contents of the first row in the order A..Z0..9:

```
AMNPRSUW347
DEIFGBHCKJL
TV0XYOZQ215
9    6 8
```

Now read off the all but the first row left to right, skipping spaces and append the characters to the converted passphrase:

```
SWAMPRUN437DEIFGBHCKJLTV0XYOZQ215968
```

Finally, we need to convert this to the array. It is best to avoid a straight row by row or column by column copy. We will do it diagonally, top left to bottom right, starting in the top right corner. This gives the array at the start of this example.

Problem 5
Playfair (continued)

What remains is how to choose the pairs in the plaintext. It is not very strong to take the characters 2 by 2, instead the text is usually scrambled in some way. We will just split the text in half. Here is an example text:

```
In skew binary, the digits include 2.
```

Copy the text, skipping any non-alphanumeric characters and converting lower to upper case.

```
INSKEWBINARYTHEDIGITSINCLUDE2
```

Now split the message into equal parts, appending an X if it contains an odd number of characters. Write the second half below the first half:

```
INSKEWBINARYTHE
DIGITSINCLUDE2X
```

The pairs are the columns. The first pair is ID which becomes VP, the first two characters of the enciphered text. The second pair is NI which becomes PC. The enciphered text is:

```
VPPCG7XK4HSGPOCPE0LBHSHLH4YV4Q
```

Deciphering is just the reverse of the above procedure.

Input will consist of one or more test cases, terminated by end-of-file. Each test case will start with a passphrase starting with an alphanumeric character in column 1. It will be followed by one or more lines to be enciphered or deciphered. A line to be enciphered will have a '+' in column 1 while one to be deciphered will have a '-' in column 1. The text to be enciphered or deciphered will start in column 2. All lines will be at most 80 characters in length. Each passphrase and data to be enciphered will contain at least 1 alphanumeric character. Data to be deciphered will contain at least 2 alphanumeric characters. If data to be deciphered contains an odd number of characters, discard the last character.

Output will be the enciphered or deciphered text, with no leading or trailing spaces.

Sample Input

```
swamp run: AM437.
+In skew binary, the digits include 2.
-VPPCG7XK4HSGPOCPE0LBHSHLH4YV4Q
-4OYUV47R3PMAR74U44YYR
MY NAME IS DUADUA EDWIN AND TOGETHER WITH MY COLLEAGUES [PARTNERS] WHO ARE
-ISQTU7YDATDHM4ZIDKLMOP90NU3MVS1S76VCZQVWPO
+the attack will use plan 42
-5MUHTDXMOYUOOODUSDM3NEEDSAN5UNQN2BPOXMCYAHDN4EDD1CSS7UN77ML7
abcdefghijklmnopqrstuvwxyz0123456789
+this is just a test
-YNEYFBNYNINNNO6J
k
+this is just a test
-YNNZNKNYOINNA56I
```

Output for the Sample Input

```
VPPCG7XK4HSGPOCPE0LBHSHLH4YV4Q
INSKEWBINARYTHEDIGITSINCLUDE2X
THISISTHERIGHTANSWER
PLEASETRANSFER3200FROMCHECKINGTOVISATHANKS
L4JEUHYAYLS4TDTYWDJIPZ
WEAREPLEASEDTOINFORMYOUOFTHERESULTSELGORDOSWEEPSTAKELOTTERYX
YNEYFBNYNINNNO6J
THISISJUSTATESTX
YNNZNKNYOINNA56I
THISISJUSTATESTX
```