Problem 1—Magic Square

The handheld Merlin game was released by Parker Brothers in 1978. Merlin was famous for being able to play several different games. Magic Square was a game played on a 3x3 grid of lighted buttons. Pressing a button would cause certain buttons in the grid to toggle between off and on. If the grid is numbered

```
1 2 3
4 5 6
7 8 9
```

Pressing 1 would toggle 1, 2, 4, and 5, and all corners toggle similarly.
Pressing 2 would toggle 1, 2, and 3, and all edge squares toggle similarly.
Pressing 5 would toggle 2, 4, 5, 6, and 8.

The goal, given an initial configuration chosen at random, was to press the buttons to toggle the lights so that the 5 button is *not* illuminated, but all other buttons are (hence the name Magic Square).

A moment's thought will reveal that the solution to any Magic Square problem is permutation-independent. For example, pressing 5 8 3 will generate the same pattern as 3 5 8 or 8 3 5, etc.

Given an initial configuration, you are to compute the *smallest* sequence of button presses that will result in the Magic Square.

INPUT SPECIFICATION. Each input case is a 3x3 grid where 1 represents a lit square and 0 represents an unlit square. Each line of the grid is followed by **<EOLN>** and an extra **<EOLN>** follows each input case.

OUTPUT SPECIFICATION. The output cases are to appear in the same order as the corresponding input cases. Each output case is the minimal sequence of buttons, listed in numerical order, that will solve the configuration specified by the input case. A space is to follow each number in the input and **<EOLN>** is to follow each output case.

SAMPLE INPUT

```
100<EOLN>
000<EOLN>
000<EOLN>
<EOLN>
000<EOLN>
000<EOLN>
000<EOLN>
<EOLN>
<EOF>
```

SAMPLE OUTPUT

```
5·6·8·9·<EOLN>
1·2·3·4·6·7·8·9·<EOLN>
<EOF>
```