Problem 1—Marching Up And Down The Square

There are 15 soldiers in a 4x4 square. (The 16th. would rather have been home with the wife and kids.) Every so often, the drill sergeant orders one specific soldier to "march up and down the square." Of course, with 15 soldiers in the grid, the only way the soldier can march is to the empty space. If he is adjacent to the empty space already, that's fine. If he isn't, then another soldier will have to move into the empty space, creating a different empty space closer to the soldier. Then another soldier will move into that empty space, and so forth until the soldier can move into an adjacent empty space.

The soldiers will not move around corners. If the soldiers cannot move strictly parallel to an edge of the square to fill the hole, they will announce that it can't be done.

Given the initial configuration of soldiers and the specific soldier that is told to move up and down the square, you are to print the square after the soldiers have moved (or refuse the order).

INPUT SPECIFICATION. Each case begins with the 4x4 grid. The grid consists of all the integers from 0 to 15 in some permutation. The 0 represents the empty space. Notice that the integers are 2-space right justified with one space between each pair. (Just look at the example.)

Following the grid is the number of the soldier given the order. This number is not right-justified and is followed by 2 **<EOLN>** characters.

<EOF> follows the last case.

OUTPUT SPECIFICATION. Each output case should appear in the same order as the corresponding input case. If there is a solution, the output should be "Case *c*:" followed by 2 **<EOLN>** characters and then the grid formatted as exactly as formatted in the input specification. An extra **<EOLN>** follows the grid.

If there is no solution, the output should be "Case *c*: I can't march up and down the square!" followed by 2 **<EOLN>** characters.

c is the case number in either situation.

SAMPLE INPUT.

```
·1··2··3··4<EOLN>
·5··6··7··8<EOLN>
·9·10·11·12<EOLN>
13·14·15··0<EOLN>
8<EOLN>
<EOLN>
·1··2··3··4<EOLN>
·5··6··7··8<EOLN>
·9·10·11·12<EOLN>
13·14·15··0<EOLN>
14<EOLN>
<EOLN>
·1··2··3··4<EOLN>
·5··6··7··8<EOLN>
·9·10·11·12<EOLN>
13·14·15··0<EOLN>
1<EOLN>
<EOLN>
<EOF>
```

SAMPLE OUTPUT.

```
Case·1:<EOLN>
<EOLN>
·1··2··3··4<EOLN>
·5··6··7··0<EOLN>
·9·10·11··8<EOLN>
13·14·15·12<EOLN>
<EOLN>
Case·2:<EOLN>
<EOLN>
·1··2··3··4<EOLN>
·5··6··7··8<EOLN>
·9·10·11·12<EOLN>
13··0·14·15<EOLN>
<EOLN>
Case·3:·I·can't·march·up·and·down·the·s
quare!<EOLN>
<EOLN>
<EOF>
```