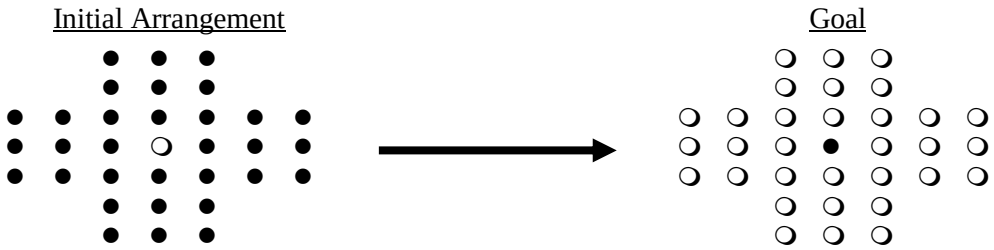
Problem 21: Hi-Q

Source filename: `hiq.cpp`
Input filename: `hiq.in`
Output filename: `hiq.out`

This puzzle-like game is played with 32 movable pegs that are initially placed on a board that has 33 holes, leaving one hole vacant. One such arrangement is shown below – with the vacant hole located in the middle of the board. A piece can move by jumping over its immediate neighbor horizontally or vertically into an empty hole on the opposite side of its neighbor; the jump removes the jumped-over neighbor from the board. The goal is to remove, by jumping, 31 pieces to finish with a single peg located in what was the original vacant hole.



The 33 hole positions are labeled using the digits `1` through `9` and the letters `A` through `Y` (skipping capital '`O`', since it is easily confused with the digit '`0`') as seen in the following figure.

```
    1 2 3
    4 5 6
7 8 9 A B C D
E F G H I J K
L M N P Q R S
    T U V
    W X Y
```

The input file (`hiq.in`) contains several test cases, one line per test case. Each line contains a single character, either a digit or a letter between `A` and `Y` (excluding the letter `O`) which represents the initial vacant hole for that test case. A single zero '`0`' digit marks the end of the input file.

For example, the input character, '`H`', would indicate that the initial vacant hole is located in the exact center of the board.

For each test case, the output file (`hiq.out`) should contain, on a single line, a list of 31 jumps, separated by single spaces, that corresponds to a solution. Each individual jump is represented by an ordered pair, `start->end`, where `start` is the position of a peg before it jumps to position `end` and the peg between `start` and `end` is removed. There should be no leading or trailing spaces on the line.

If there is more than one solution for a given test case, you are free to pick any one of the solutions.

If there is no solution for a given test case, the corresponding output line should contain the text: "`No Solution for hole at n`", where `n` indicates the location of the initially vacant hole.

Sample Input File

```
7
H
0
```

Sample Output File

```
9->7 1->9 3->1 6->4 A->8 1->9 8->A B->9 D->B G->4 E->G H->F I->6 K->I M->8 7->9 4->G Q->B 6->I S->Q
                                                P->R Y->Q I->V W->Y Y->Q R->P U->H H->F T->G G->E L->7
5->H 8->A 1->9 3->1 A->8 7->9 B->3 D->B G->4 1->9 E->G G->4 I->6 3->B K->I I->6 T->G L->N N->9 4->G
                                                G->I Q->B 6->I S->Q P->R Y->Q I->V W->Y Y->Q R->P U->H
```