# 10632   Pyramid

In a classical computer game, a creature jumps around on the cells in a pyramid much like the one in the picture below. The creature can jump to the cells directly above him or below him. Jumping outside the pyramid is of course not allowed.

When the creature lands on a cell, it changes color: from red to green, green to blue or blue to red. Given how the cells are initially colored, determine a jumping sequence containing no more than 5000 jumps that turns all the cells into blue. You are allowed to start from any cell in the pyramid you want (this guarantees that there will always be a solution). The first cell to change color is the one reached after jumping from the start cell.
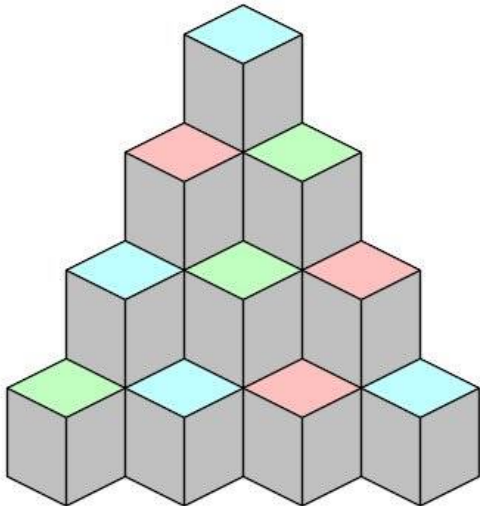
Input

The input consists of several test cases (at most 50). Each test case begins with a line containing a single integer n ($2 \leq n \leq 40$), the height of the pyramid. The next n lines describe the start configuration of the pyramid using the upper case letters 'R', 'G' and 'B'. The pyramid is described in row major order, each row from left to right; see the sample input for the exact format (the first sample input corresponds to the pyramid in the picture). At least one cell in the pyramid is not blue in the start configuration.

The input ends with a case where n is 0, which should not be processed.

Output

For each test case, output two lines. The first line should contain two integers, the creatures start position in the pyramid. The first integer is the row (1 being the top row) and the second integer is the cell in that row (1 being the leftmost cell). The next line should contain a string describing the jumps. The length of this string should be at most 5000 characters. Each character should either be a '7' (jumping up-left), a '9' (up-right), a '1' (down-left) or a '3' (down-right). Any solution that has no more than 5000 jumps and which turns all cells into blue will be accepted.

Sample Input

```
4
B
RG
BGR
GBRB
2
R
GB
0
```

Sample Output

```
3 1
193919193919373737717191991919373737
2 1
919
```