Mock CCC '20 S2 - Flipper Redux

Time limit: 2.0s **Memory limit:** 512M

Recently, **Plasmatic** got into flipper puzzles! A flipper puzzle is an $N \times N$ grid with black and white squares. At any point, **Plasmatic** can flip all the squares in a single row or column, changing the respective squares from black to white and white to black. Furthermore, a flipper puzzle is considered solved when all the squares are black.

However, when **Plasmatic** began working on his flipper puzzle the other day, he found that no matter how he flipped the rows and columns he couldn't solve it. After raging on the puzzle for a few hours he came to the conclusion that the puzzle had been tampered with and threw it in the trash.

Now, seeing this as a perfect opportunity to have some fun, **ChrisT** took **Plasmatic**'s puzzle and began solving it. This time though, **ChrisT** chose to write a program to help him accomplish his task. But due to his subpar programming skills, he wasn't able to finish the program. Can you help him?

Input Specification

The first line of input contains the integer N.

The next N lines each contain N space separated integers, with the j^{th} integer on line i denoting the value of $a_{i,j}$, the j^{th} number on the i^{th} row on the final configuration.

Output Specification

If the puzzle is unsolvable, print `-1`.

Otherwise, first output an integer M, the number of moves needed to solve the puzzle. Next, output M more lines, denoting the operations needed to solve the puzzle. If there are multiple answers, output any of them.

Note that M doesn't have to be minimal, it just has to be ≤ 4000.

Each operation must be one of the following:

- `R x_i`: Flip all squares in row x_i
- `C x_i`: Flip all squares in column x_i

Note that the rows count from top to bottom and columns count from left to right.

Constraints

For all subtasks:

In your output, $0 \leq M \leq 4 \times 10^3$.

$1 \leq i \leq N \leq 2 \times 10^3$

$0 \leq a_{i,j} \leq 1$

For 1 out of 15 available marks, $1 \leq N \leq 2$.

For an additional 3 out of 15 additional marks, $1 \leq N \leq 10$.

Sample Input 1

```
3
1 0 0
1 0 0
0 1 1
```

Sample Output 1

```
2
R 3
C 1
```

Note that the rows count from top to bottom and columns count from left to right.

Sample Input 2

```
3
0 0 0
1 0 0
0 0 0
```

Sample Output 2

```
-1
```