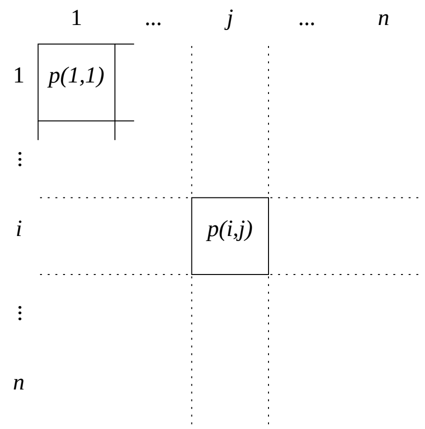
Task: PUZ
Puzzle

Day 2. Source file `puz.*` *July 15, 2004*

Available memory: 32 MB. Maximum running time: 0.3 s.

The king of Byteland has received a gift, a jigsaw puzzle. The puzzle consists of a board of size $n \times n$. The field in the i-th row and j-th column ($1 \leq i, j \leq n$) has coordinates (i, j) and contains a piece with the number $p(i, j)$, $1 \leq p(i, j) \leq n^2$. Each of the numbers $1 \ldots n^2$ appears on exactly one of the pieces.



Coordinates of the fields

To solve the puzzle, you have to put the pieces in order, so that for each $1 \leq i, j \leq n$ the field (i, j) contains the piece with number $j + (i - 1) * n$.

The following moves are allowed to solve the puzzle:

- a cyclic shift of all pieces in a row a certain number of fields to the right,
- a cyclic shift of all pieces in a column a certain number of fields down.

The king of Byteland managed to solve his puzzle, but he is not sure if he would be able to solve it starting from a different initial configuration. Help him to solve this problem.

Task

Write a program that:

- Reads from the standard input the description of the initial configuration of the puzzle.
- Finds out if the pieces on the board can be put in order using only the moves given above. If the solution is possible, the program should find the moves putting the pieces in order.
- Writes the result to the standard output.

Input

In the first line of the standard input there is one integer n — the size of the board side ($2 \le n \le 200$). The following n lines contain the description of the initial configuration. The line $i+1$ contains n integers $p(i,1), p(i,2), \dots p(i,n)$ separated by single spaces.

Output

If there is no solution, the program should write to the standard output only one line containing only one word `NO`.

If a solution exists, the first line should contain one integer m — the number of moves leading to the solution of the puzzle. The number of moves in your solution must not exceed 400 000. The following m lines should contain the descriptions of the moves, one move per line.

Each such line should consist of a letter `R` (for shifting a row to the right) or `C` (for shifting a column down), a space, and two integers: k and l separated by a space; $1 \le k \le n$, $1 \le l \le n-1$. A line containing `R` k l describes a cyclic shift of the k-th row l fields to the right. Such a move leads to the following board configuration:

$$p'(i,j) = \begin{cases} p(i, j+n-l) & \text{if } i = k \text{ and } j \le l \\ p(i, j-l) & \text{if } i = k \text{ and } j > l \\ p(i,j) & \text{if } i \ne k \end{cases}$$

Similarly, a line containing `C` k l describes a cyclic shift of the k-th column l fields down.

If there are several possible solutions, your program should output anyone of them.

Example

For the input data:

```
4
4 6 2 3
5 10 7 8
9 14 11 12
13 1 15 16
```

the correct result is:

```
2
C 2 1
R 1 3
```

The above sequence of moves gives the following sequence of configurations:

4	6	2	3
5	10	7	8
9	14	11	12
13	1	15	16

4	1	2	3
5	6	7	8
9	10	11	12
13	14	15	16

1	2	3	4
5	6	7	8
9	10	11	12
13	14	15	16