Problem D. Solar Panel Grid Optimization

In the not-so-distant future, the world is managed by advanced robots capable of performing complex tasks to maintain global systems. One such system is responsible for managing a massive grid of solar panels that generate renewable energy for an entire city. These panels are arranged in an $N \times N$ matrix, where each cell in the matrix can either be in an active (1) or inactive (0) state.

Figure 1: By AleSpa - Own work, CC BY-SA 3.0

To optimize energy production, the robots need to transform the current state matrix A of the solar panel grid into a desired state matrix B. The transformation process involves two types of operations:

- Shift an entire row to the left, i.e., select a positive integer i between 1 and N and transform $(A_{i,1}, A_{i,2}, \ldots, A_{i,N})$ into $(A_{i,2}, A_{i,3}, \ldots, A_{i,N}, A_{i,1})$.
- Shift an entire column downwards, i.e., select a positive integer i between 1 and N and transform $(A_{1,i}, A_{2,i}, \ldots, A_{N,i})$ into $(A_{N,i}, A_{1,i}, A_{2,i}, \ldots, A_{N-1,i})$, and then flip the state of the top cell, making the new $A_{1,i} \leftarrow 1 - A_{1,i}$.

You are tasked with finding a sequence of operations that can transform the grid from its current state A to the desired state B by using no more than 10^3 operations.

Input

The first line of the input contains a single integer N ($3 \le N \le 20$).

The next N lines describe the initial matrix A. The j-th character on the i-th line of these lines will be either 0 or 1, representing the value of $A_{i,j}$.

After these N lines, there will be an empty line.

The next N lines describe the target matrix B. The j-th character on the i-th line of these lines will be either 0 or 1, representing the value of $B_{i,j}$.

Output

The first line of the output should be an integer T ($0 \le T \le 10^3$), representing the number of the operations you used.

The next T lines of the output describe the operations. The i-th line of these lines should be one of the following:

- `row` i: Perform an operation on the i-th row.
- `column` i: Perform an operation on the i-th column.

If there are multiple possible solutions, you may print any of them.

Example

standard input	standard output
4 1011 1100 0100 1011 1001 0110 0110 1001	4 row 2 row 2 row 2 column 3