

Problem G. Choreography

Input file: *standard input*
Output file: *standard output*
Time limit: 10 seconds
Memory limit: 512 mebibytes

To celebrate the sixteenth anniversary of the unification of Vertical Byteland and Horizontal Byteland, a great parade will pass through the streets of the capital. The court choreographer Byteleon has decided to honor this event with a special choreography of the new national dance: Byterek.

Byterek is danced by n^2 dancers who are arranged in a square formation of size $n \times n$. It consists of a sequence of *vertical* and *horizontal* phases. During the dance dancers freely swap places with each other, on one condition, that for the time of vertical phases swapping is allowed exclusively within dancer's own column and during horizontal phases – within one's own row.

The choreography will also carry an additional meaning which should please king Byteasar. Every dancer will have an assigned costume of a specific color, so that the formation creates an image seen from the royal balcony. Byteleon wants the image to initially look like the flag of Vertical Byteland, and at the end of the dance – the flag of Horizontal Byteland. Unfortunately this task seems to be too hard for him, especially that the dance should consist of as few phases as possible, not to bore Byteasar. Please, help Byteleon and write a program that generates phases of Byterek.

Input

In the first line one integer $Z \leq 100$ is given, denoting number of testcases described in following lines.

The first line of each test case contains one integer n , denoting the length of columns and rows of the formation. The next n lines, each containing n integers, describe the initial arrangement of the dancers. Every number in $[1, n^2]$ occurs only once in this description and represents the desired position of the dancer in the final arrangement. This means that the final arrangement is a table in which every row and column is sorted, just like in the sample tests.

$n \in [1, 500]$, sum of n over all test cases does not exceed 1000.

Output

For each test case, in the first line your program should output an integer k – the minimum number of phases of Byterek. Then it should output k descriptions of arrangements in subsequent phases. The description of one arrangement consists of n rows each consisting of n integers, where each integer in $[1, n^2]$ occurs exactly once. If $k > 0$, the first arrangement must be possible to reach from the one given in the input after one phase of Byterek. The next one should be possible to reach from the previous one etc. The last arrangement should be equal to a sorted table. The order of vertical and horizontal phases is arbitrary.

Example

standard input	standard output
3	2
2	2 1
2 3	4 3
4 1	1 2
3	3 4
9 2 7	3
8 1 4	2 7 9
6 5 3	8 1 4
2	5 6 3
1 2	2 1 3
3 4	5 6 4
	8 7 9
	1 2 3
	4 5 6
	7 8 9
	0