Problem M. Minegraphed

Time limit: 2 seconds

Marika is developing a game called "Minegraphed" that involves moving around a 3D rectangular world.

Each cell of a parallelepiped game field is either an empty cell or an obstacle cell. You are always standing inside an empty cell, located either in the bottommost layer or on top of an obstacle cell. Each move can be made in one of four directions: north, east, south, or west. You make a move according to these rules:

- If you try to move outside of the parallelepiped, then you can't make this move.
- Otherwise, if the cell in front of you is empty, then you move one cell forward and then fall towards the bottom until you reach the bottommost layer or an obstacle cell.
- Otherwise, if you are in non-topmost layer, the cell in front of you is an obstacle, the cell above you and the cell above that obstacle are both empty, then you move (climb up) on top of that obstacle.
- Otherwise, you can't make this move.

Marika prepared a directed graph with n vertices. Now she wants to lay out the field and label n different possible standing positions with numbers from 1 to n, so that it is possible to get from the cell labeled i to the cell labeled j by making valid moves if and only if in Marika's graph it is possible to get from vertex i to vertex j along the edges of the graph. Help Marika design a field satisfying this property.

Input

The first line contains a single integer n ($1 \le n \le 9$) — the number of vertices. Each of the next n lines contains n integers equal to 0 or 1. The j-th number in the i-th line is 1 if there is an edge from vertex i to vertex j and 0 otherwise. The i-th number in the i-th line is always zero.

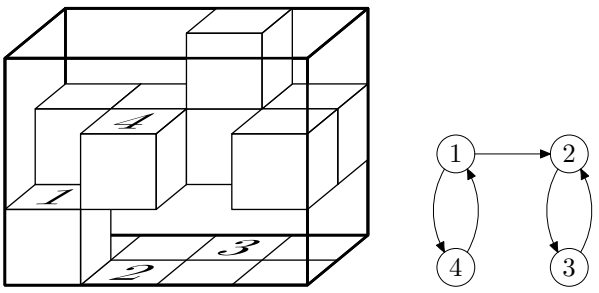
Output

Output a layer-by-layer description of a field that has the same reachability as the given graph.

The first line should contain three positive integers x, y, and z, the west-east, north-south, and top-bottom sizes of the designed parallelepiped. Blocks describing z layers should follow, from the topmost layer to the bottommost layer, separated by single empty lines. Each block should contain y lines of length x, consisting of dots ('.'), hashes ('#'), and digits from 1 to n. A hash denotes an obstacle cell. A dot denotes an unlabeled empty cell. A digit denotes a labeled empty cell. Each digit from 1 to n should appear exactly once. Each digit should be located either in the bottommost layer or on top of an obstacle cell. It is okay for obstacles to be "hanging in the air" (there is no gravity for them).

The volume of the field $x \times y \times z$ should not exceed 10^6. It is guaranteed that there is a correct field fitting into this limit for any possible graph in the input.

Example

standard input	standard output	Illustration
4 0 1 0 1 0 0 1 0 0 1 0 0 1 0 0 0	4 2 3 ..#. .4.. #### 1#.# ..3. #2..	

Note

Note that you can climb up from cell 1 to cell 4, but you cannot climb up from cell 2 to cell 1 because of a "ceiling" obstacle.