

Problem D - Delicious Pizza

For the next ACM practice, Daniel is ordering pizza from Paul’s Pizza, a newly opened pizzeria that serves novel square pizzas. The novelty of the pizzas (besides being square) is that when you view the pizza as a $n \times n$ grid, each cell of the grid has a specific height of crust ranging from 1 to n inches. The ACM club members were quite picky about the crust and refuse to have the pizza unless their following demands were met:

1. For each row and each column, the heights of each grid cell must be distinct.
2. A non-empty cell with a number indicates that there must be crust of that height in that cell.
3. If there is a number to the left of a row, then this represents the number of visible cells when viewing the row from the left side. A similar rule applies if there is a number to the right of a row.
4. Similarly, if there is a number above a column, then this represents the number of visible cells when viewing the column from above. A similar rule applies if there is a number to the bottom of a column.

Since the ACM club members are extremely picky, it’s possible that there is no such pizza that satisfies their demands. Can you help Daniel order the perfect pizza?

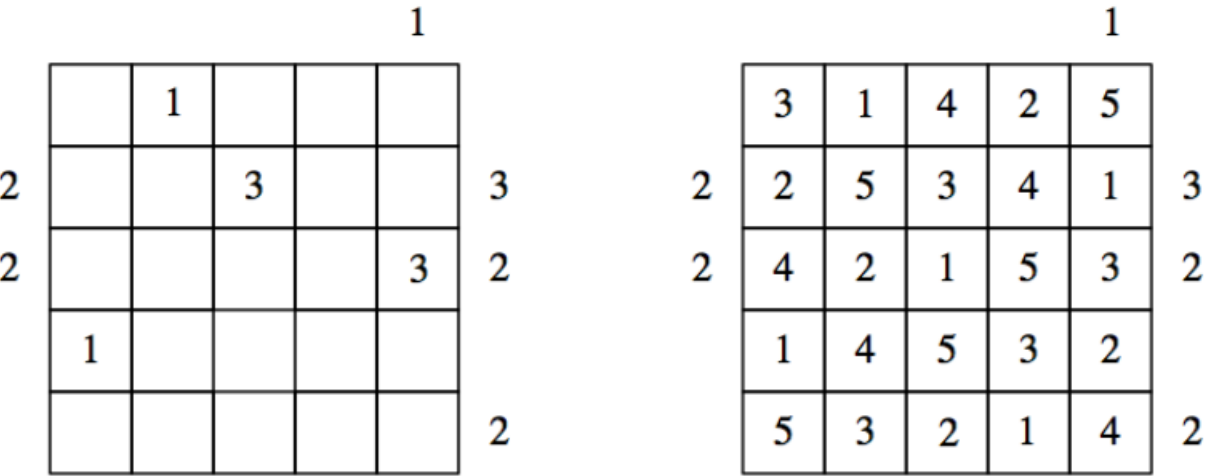


Figure 1: An example rule (left) and a satisfying configuration (right).

Input

The first line contains a single integer, T specifying the number of test cases.

Each test case begins with a single integer, $1 \leq n \leq 5$, on its own line, specifying the length of the square pizza.

$(n+2)$ lines of $(n+2)$ characters then follows, describing the configuration rule. The outer perimeter specifies the “visibility constraints” (rules of type 3 and 4), and the interior specifies cells where crust of a specific height must be used (rules of type 2). A ‘-’ denotes the absence of any constraint for the row / column (if it is on the perimeter) or for the cell (if it is in the interior).

The corners are guaranteed to be ‘-’ and every other character is either ‘-’ or a single digit between 1 and n .

Output

For each test case, if there is a configuration satisfying the rule, output the configuration (without the perimeter); otherwise, output the word “no” by itself. Output a blank line after each test case. If there are multiple solutions, output any of them.

Sample Input

```
2
5
-41223-
25----3
3-----2
2-----1
1-----5
3-----2
-23212-
5
-----1-
--1----
2--3--3
2----32
-1-----
-----2
-----
```

Sample Output

no

```
31425
25341
42153
14532
53214
```
