

You are given a square map as a matrix of integer strings. Each cell of the map has a value denoting its depth. We will call a cell of the map a *cavity* if and only if this cell is not on the border of the map and each cell adjacent to it has *strictly smaller depth*. Two cells are adjacent if they have a common side, or *edge*.

Find all the cavities on the map and replace their depths with the uppercase character **X**.

For example, given a matrix:

```
989
191
111
```

You should return:

```
989
1X1
111
```

The center cell was deeper than those on its edges:  $[8,1,1,1]$ . The deep cells in the top two corners don't share an edge with the center cell.

### Function Description

Complete the *cavityMap* function in the editor below. It should return an array of strings, each representing a line of the completed map.

*cavityMap* has the following parameter(s):

- *grid*: an array of strings, each representing a row of the grid

### Input Format

The first line contains an integer  $n$ , the number of rows and columns in the map.

Each of the following  $n$  lines (*rows*) contains  $n$  positive digits without spaces (*columns*) representing depth at *map[ $row$ ,  $column$ ]*.

### Constraints

$$1 \leq n \leq 100$$

### Output Format

Output  $n$  lines, denoting the resulting map. Each cavity should be replaced with the character x.

### Sample Input

```
4
1112
1912
1892
1234
```

### Sample Output

```
1112
1X12
18X2
1234
```

### Explanation

The two cells with the depth of 9 are not on the border and are surrounded on all sides by shallower cells. Their values have been replaced by X.

