

12605 Ripple Effect

Check if a solution to a Ripple Effect puzzle is valid.

Ripple Effect is played on a rectangular grid divided into polyominoes. A polyomino is a plane geometric figure formed by joining one or more equal squares edge to edge.

The solver must place one positive integer into each cell of the grid — some of which may be given in advance — according to these rules:

- Every integer from 1 to the quantity of cells in a polyomino must appear exactly once in that polyomino.
- If two identical numbers appear in the same row or column, at least that many cells with other numbers must separate them.

For example, two cells both containing ‘1’ may not be orthogonally adjacent, but must have at least one cell between them with a different number. Two cells marked ‘3’ in the same row or column must have at least three cells with other numbers between them in that row or column, and so on.

3	4	1	2	3	5	4	1	3	2
1	2	3	1	2	1	3	2	5	4
2	1	4	3	5	2	1	4	2	3
1	3	2	4	1	3	2	5	4	1
3	2	1	5	2	4	1	3	1	2
1	5	3	1	4	2	3	1	2	1
2	1	5	2	3	1	4	2	3	5
4	3	1	6	1	3	5	4	1	2
1	2	4	3	5	1	2	1	6	4
3	1	2	4	1	2	6	5	4	3

In this problem polyominoes will be of size at most 8.

Input

The input starts with a line containing an integer T ($T \leq 100$), the number of test cases.

Each test case starts with two integers on a line, R and C ($4 \leq R, C \leq 15$).

R lines follow, each containing a string of C digits d_i ($1 \leq d_i \leq 8$), the description of the solution.

Next R lines will contain C integers ($R \times C$ table, we will call it “*descr*”), each describing the corresponding cell in the puzzle ($0 \leq descr(r, c) \leq 15$).

The value of $descr(r, c)$ is determined by the values of connections with neighbouring cells, in the following manner:

```
descr(r, c) = 0
if(connected((r, c), (r - 1, c)) descr(r, c) += 1; (UP)
if(connected((r, c), (r, c + 1)) descr(r, c) += 2; (RIGHT)
if(connected((r, c), (r + 1, c)) descr(r, c) += 4; (DOWN)
if(connected((r, c), (r, c - 1)) descr(r, c) += 8; (LEFT)
```

For example, a polyomino of size 1 (single square not connected to others) will have the value of 0 and a square that is connected only to the squares above and below will have the value of 5.

See sample input for clarification.

Output

For each test case print either ‘valid’ or ‘invalid’ on a single line.

Sample Input

```
2
6 6
241321
312432
131243
423121
214312
141231
2 12 4 4 6 8
4 5 1 5 5 4
5 1 0 5 1 5
3 8 4 1 4 1
2 10 9 2 9 4
0 2 10 10 8 1
6 6
421321
312432
131243
423121
214312
141231
2 12 4 4 6 8
4 5 1 5 5 4
5 1 0 5 1 5
3 8 4 1 4 1
2 10 9 2 9 4
0 2 10 10 8 1
```

Sample Output

```
valid
invalid
```