

### 1.6.2 Minesweeper

**PC/UVa IDs:** 110102/10189, **Popularity:** A, **Success rate:** high **Level:** 1

Have you ever played Minesweeper? This cute little game comes with a certain operating system whose name we can't remember. The goal of the game is to find where all the mines are located within a  $M \times N$  field.

The game shows a number in a square which tells you how many mines there are adjacent to that square. Each square has at most eight adjacent squares. The  $4 \times 4$  field on the left contains two mines, each represented by a “\*” character. If we represent the same field by the hint numbers described above, we end up with the field on the right:

*...	*100
....	2210
.*..	1*10
....	1110

#### Input

The input will consist of an arbitrary number of fields. The first line of each field contains two integers  $n$  and  $m$  ( $0 < n, m \leq 100$ ) which stand for the number of lines and columns of the field, respectively. Each of the next  $n$  lines contains exactly  $m$  characters, representing the field.

Safe squares are denoted by “.” and mine squares by “\*,” both without the quotes. The first field line where  $n = m = 0$  represents the end of input and should not be processed.

#### Output

For each field, print the message **Field #x:** on a line alone, where  $x$  stands for the number of the field starting from 1. The next  $n$  lines should contain the field with the “.” characters replaced by the number of mines adjacent to that square. There must be an empty line between field outputs.

#### Sample Input

```
4 4
*...
....
.*..
....
3 5
**...
.....
.*...
0 0
```

#### Sample Output

```
Field #1:
*100
2210
1*10
1110

Field #2:
**100
33200
1*100
```