Have you ever played Minesweeper? It's a cute little game which comes within a certain Operating System which name we can't really remember. Well, the goal of the game is to find where are all the mines within a $M \times N$ field. To help you, the game shows a number in a square which tells you how many mines there are adjacent to that square. For instance, supose the following 4×4 field with 2 mines (which are represented by an '*' character):

```
*...
....
.*..
....
```

If we would represent the same field placing the hint numbers described above, we would end up with:

```
*100
2210
1*10
1110
```

As you may have already noticed, each square may have at most 8 adjacent squares.

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 stands for the number of lines and columns of the field respectively. The next n lines contains exactly m characters and represent the field.

Each safe square is represented by an '.' character (without the quotes) and each mine square is represented by an '*' character (also 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, you must print the following message in a line alone:

`Field #`x`:`

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 adjacent mines 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
```