

Minesweeper

Have you ever played the game [minesweeper](#)? It comes with an operating system whose name slips our mind. Well, the aim of the game is to discover where all the mines are hidden in a rectangular $m \times n$ playing field. To help you, numbers are displayed in the boxes of the playing field. These indicate how many mines are located in adjacent boxes. This includes either horizontally, vertically or diagonally adjacent boxes. For example, suppose there are two mines hidden in a 4×4 playing field (their location is indicated by an asterisk):

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

If we indicate how many mines are located in adjacent boxes in the empty boxes (indicated by dots), we get the following representation of the court:

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

As you might already have noticed, each box has eight adjacent boxes at most.

Input

The input consists of t test cases ($t \leq 100$). The first line of the input contains an integer t . Then follows the description of the t test cases. The first line of the description of a test case contains two integers m and n ($0 < m, n \leq 100$), which respectively indicate the number of rows and number of columns of the table. This is followed by m lines each containing exactly n characters that indicate whether the corresponding grid contains a mine (*) or not (.).

Output

For each test case, print the playing field in the same format as in the input, but where each box that does not contain a mine is now represented by a number indicating how many adjacent boxes do contain a mine. Make sure there is a blank line between successive playing fields.

Example

Input:

```
2
4 4
* ..
...
.*.
....
3 5
** ..
.....
```

*...

Output:

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

```
**100
33200
1*100
```

Heb je ooit het spelletje [mijnenveger](#) gespeeld? Het wordt meegeleverd met een besturingssysteem waarvan de naam ons nu niet direct te binnen schiet. Nu ja, de bedoeling van het spel is te ontdekken waar alle mijnen verborgen liggen in een rechthoekig $m \times n$ speelveld. Om je te helpen, worden getallen in de hokjes van het speelveld weergegeven. Deze geven aan hoeveel mijnen er in aangrenzende hokjes liggen. Hierbij worden zowel horizontaal, verticaal als diagonaal aangrenzende hokjes in rekening gebracht. Veronderstel bijvoorbeeld dat er in een 4×4 speelveld twee mijnen verborgen liggen (hun locatie wordt aangegeven met een sterretje):

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

Indien we nu in de lege hokjes (aangegeven met een puntje) aangeven hoeveel mijnen er in aangrenzende hokjes liggen, dan krijgen we de volgende voorstelling van het speelveld:

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

Zoals je al zou kunnen opgemerkt hebben, heeft elk hokje ten hoogste 8 aangrenzende hokjes.

Invoer

De invoer bestaat uit t testgevallen ($t \leq 100$). De eerste regel van de invoer bevat een natuurlijk getal t . Daarna volgt de omschrijving van de t testgevallen. De eerste regel van de omschrijving van een testgeval bestaat uit twee natuurlijke getallen m en n ($0 < m, n \leq 100$), die respectievelijk het aantal rijen en het aantal kolommen van het speelveld aangeven. Daarna volgen m regels die elk exact n lettertekens bevatten die aangeven of het corresponderende rooster een mijn bevat (*) of niet (.).

Uitvoer

Schrijf voor elk testgeval het speelveld uit in hetzelfde formaat als in de invoer, maar waarbij elk hokje dat geen mijn bevat nu wordt voorgesteld door een cijfer dat aangeeft hoeveel aangrenzende hokjes er een mijn bevatten. Zorg ervoor dat er een lege regel staat tussen opeenvolgende speelvelden.

Voorbeeld

Invoer:

2
4 4
*...
....
. * ..
....
3 5
** ...
.....
. *

Uitvoer:

*100
2210
1*10
1110

**100
33200
1*100