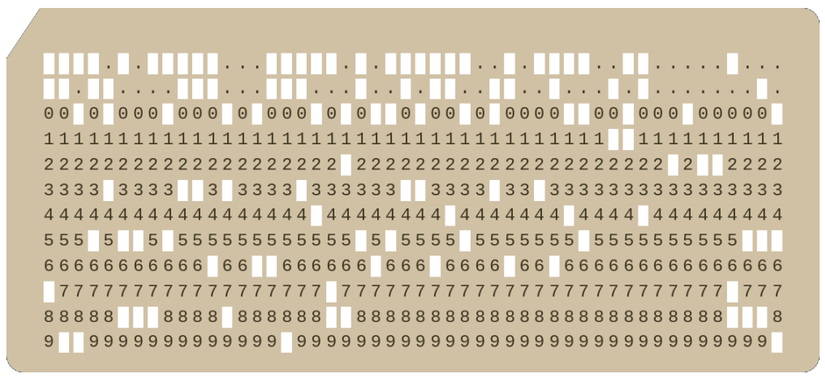
Qualification Round 2022 - Code Jam 2022

Punched Cards

Problem

A secret team of programmers is plotting to disrupt the programming language landscape and bring punched cards back by introducing a new language called *Punched Card Python* that lets people code in Python using punched cards! Like good disrupters, they are going to launch a viral campaign to promote their new language before even having the design for a prototype. For the campaign, they want to draw punched cards of different sizes in ASCII art.



The ASCII art of a punched card they want to draw is similar to an $\mathbf{R} \times \mathbf{C}$ matrix without the top-left cell. That means, it has $(\mathbf{R} \cdot \mathbf{C}) - 1$ cells in total. Each cell is drawn in ASCII art as a period (`.`) surrounded by dashes (–) above and below, pipes (|) to the left and right, and plus signs (+) for each corner. Adjacent cells share the common characters in the border. Periods (`.`) are used to align the cells in the top row.

For example, the following is a punched card with $\mathbf{R} = 3$ rows and $\mathbf{C} = 4$ columns:

```
..+-+-+-+
..|.|.|.|
+-+-+-+-+
|.|.|.|.|
+-+-+-+-+
|.|.|.|.|
+-+-+-+-+
```

There are more examples with other sizes in the samples below. Given the integers $\mathbf{R}$ and $\mathbf{C}$ describing the size of a punched card, print the ASCII art drawing of it as described above.

Input

The first line of the input gives the number of test cases, $\mathbf{T}$. $\mathbf{T}$ lines follow, each describing a different test case with two integers $\mathbf{R}$ and $\mathbf{C}$: the number of rows and columns of the punched card that must be drawn.

Output

For each test case, output one line containing `Case #`x`:`, where x is the test case number (starting from 1). Then, output $(2 \cdot \mathbf{R}) + 1$ additional lines with the ASCII art drawing of a punched card with $\mathbf{R}$ rows and $\mathbf{C}$ columns.

Limits

Time limit: 5 seconds.
Memory limit: 1 GB.

Test Set 1 (Visible Verdict)

$1 \leq \mathbf{T} \leq 81$.
$2 \leq \mathbf{R} \leq 10$.
$2 \leq \mathbf{C} \leq 10$.

Sample

Sample Input

```
3
3 4
2 2
2 3
```

Sample Output

```
Case #1:
..+-+-+-+
..|.|.|.|
+-+-+-+-+
|.|.|.|.|
+-+-+-+-+
|.|.|.|.|
+-+-+-+-+
Case #2:
..+-+
..|.|
+-+-+
|.|.|
+-+-+
Case #3:
..+-+-+
..|.|.|
+-+-+-+
|.|.|.|
+-+-+-+
```

Sample Case #1 is the one described in the problem statement. Sample Cases #2 and #3 are additional examples. Notice that the output for each case contains exactly $\mathbf{R} \cdot \mathbf{C} + 3$ periods.