11074 Draw Grid

It is very easy to draw grids with ASCII characters. For example look at the picture on the right. It shows a (4×4) grid, where each smallest square is of size 3 and the thickness of drawing line is 2.

   In this problem your job is very simple: Given the size of the grid, size of smallest square and thickness of drawing line you will just have to draw the grid.

Input

The input file contains at most 101 lines of inputs. Each line contains three integers S, T and N $(0 < S, T, N < 21)$. Here S is the size of smallest squares, T is the thickness of drawing line and N is the size of the grid. Input is terminated by a set where the value of S, T and N is zero. This set should not be processed.

Output

For each set of input first produce the serial of output. In next several lines draw an $(N \times N)$ sized grid where each smallest square is of size $(S \times S)$ and the thickness of drawing line is T. Print a blank line after the output of each case. Note that line pixels are denoted with '*' (asterisk) and blank pixels are denoted with '.'.

Sample Input

```
3 3 3
2 3 4
0 0 0
```

Sample Output

```
Case 1:
********************
********************
********************
***...***...***...***
***...***...***...***
***...***...***...***
********************
********************
********************
***...***...***...***
***...***...***...***
***...***...***...***
********************
********************
********************
```

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

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

Case 2:
**********************
**********************
**********************
***..***..***..***..***
***..***..***..***..***
**********************
**********************
**********************
***..***..***..***..***
***..***..***..***..***
**********************
**********************
**********************
***..***..***..***..***
***..***..***..***..***
**********************
**********************
**********************
***..***..***..***..***
***..***..***..***..***
**********************
**********************
**********************
```