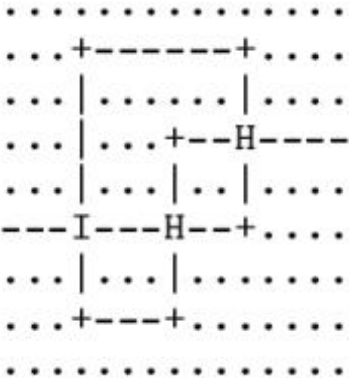


11767 World of Knots

Today, we will be unraveling the mysteries of one of the most twisted and curly beasts in all existence — knots! The problem here is simple: given the picture of a string laid out curled on a table, you are to determine if it will curl up into a knot when the ends are pulled out.

The input will consist of a diagram of one single continuous string described in a two dimensional grid of characters such as shown above. The ‘-’ and ‘|’ characters represent a horizontal and vertical segment of the string, respectively. ‘+’ characters represent corners where the string turns at right angles on the table. ‘I’ or ‘H’ characters represent locations where parts of the strings cross:



- ‘H’ represents locations where the horizontal string passes over the vertical string
- ‘I’ represents locations where the horizontal string passes under the vertical string

The dot character, ‘.’, obviously, represents empty spaces of the table unoccupied by the string. Two horizontally adjacent nonempty spaces of the table are connected by the string iff neither of them are ‘|’ characters. Similarly, vertically adjacent nonempty spaces are connected by the string iff neither of them are ‘-’ characters. Inputs will be such that every ‘+’ character will represent a proper corner where the string turns at a unambiguously at a right angle: formally, every ‘+’ character will be connected to exactly one vertically adjacent and exactly one horizontally adjacent space.

Moreover, to further simplify matters, you can assume that the only characters along the border of the given diagram, other than dots, will represent endpoints of the string. In short, the input will unambiguously specify a valid piece of string starting and ending at the border of the input diagram. Finally, assume that the string has negligible width and is made of a very smooth material, so that parts of the string can easily slide over each other with negligible friction.

Input

The input will consist of at most 80 test cases. Each test case will start with a single line containing two integers, *r* and *c*, indicating that the size of the diagram for that case is of *r* rows and *c* columns. This line will be followed by *r* more lines, each with exactly *c* characters, with characters in each line representing a row of the diagram. You may assume that $2 \leq r, c \leq 120$.

Input is terminated by a line containing two zeroes. This line should not be processed.

Output

The output should consist of exactly one line for each test case in the format ‘Case *c*: *Response*’, where *c* is the test case serial number, starting from 1, and *Response* is either the string ‘straightened’ or ‘knotted’ (without the quotes) depending on whether the string will straighten out or coil up into a knot, respectively.

See the sample for further clarifications.

Sample Input

```

9 15
.....
...+-----+...
...|.....|...
...|...+--H---
...|...|..|...
---I---H--+...
...|...|.....
...+---+.....
.....
9 15
.....
...+-----+...
...|.....|...
...|...+--I---
...|...|..|...
---I---H--+...
...|...|.....
...+---+.....
.....
0 0

```

Sample Output

```

Case 1: knotted
Case 2: straightened

```