Problem Jumpy

Input file	jumpy.in
Output file	jumpy.out

Little Square and Little Triangle play a game of Jumpy. In this game they alternately control Jumpy the carpenter as he runs around a rectangular board formed out of $N \times M$ cells. Each cell is either a wall or an empty cell which is colored in either blue, red or green. Jumpy will start from an empty cell. Little Square and Little Triangle take turns alternately and are allowed to play only in a very particular way: Little Square can only jump horizontally (left and right), and Little Triangle can only jump vertically (up and down). While doing a jump, they can't travel through a wall. They don't necessarily have to jump on an adjacent cell: it is allowed to jump over several empty cells, just not over any walls.

Initially, all cells are blue, but they can change colors after each move. If Jumpy lands on a red cell it means a loss for the player controlling Jumpy; while landing on a green one will result in a win for the player controlling Jumpy. If the player lands on a blue cell, the cell from which he took off becomes green and the cell he lands on becomes red, in this order. These changes happen after landing, and cannot lead to or prevent an immediate win or lose. **The ordering of color changes matters if the starting and ending cell coincide**.

We want to know, for every starting position and starting player, who would win if both Little Square and Little Triangle play optimally.

Input file

The first line of the input file will contain N and M, the dimensions of the board.

The next N lines will contain M characters, either . (which represents an empty cell) or # (which represents a wall). These represent the playing board.

Output file

The output file will contain N lines each with M characters, representing the answers for each cell.

If a cell contains a wall, output #. If neither player would win if they moved first starting in a cell, output N. If both players would win if they moved first starting in a cell, output B. If only Little Square would win if they moved first starting in a cell, output S. If only Little Triangle would win if they moved first starting in a cell, output T.

Constraints

- $1 \le N, M \le 500$

Subtask 1 (31 points)

- $1 \le N \cdot M \le 20$

Subtask 2 (13 points)

- There are no walls.

Subtask 3 (7 points)

- There is at most 1 wall.

Subtask 4 (22 points)

- There is no path from a cell to itself that passes through any cell at most once.

Subtask 5 (17 points)

- $1 \leq N, M \leq 300$

Subtask 6 (10 points)

- No additional constraints.

Examples

jumpy.in	jumpy.out
9 9 ....#..#. .####..#. ....#. .#..#.##. .#..#.... ########. ....#.... ##.##.##. ....#.	BSSS#TT#T T####TT#T BSBBSBB#T T#BB#T##T T#BB#BSSB ########T SSBS#BSSB ##B##B##T SSBSSBS#T
5 5 ...#. .###. ...#. ##.#.	BSS#T B###T BBB#T ##B#T SSBSB