We have recently acquired a robot to help us move heavy boxes around in a factory. To move a box toward a direction, the robot first needs to reach the box from behind, and then push it forward in the desired direction.

There are several obstacles inside the factory. We represent the factory by an $m \times n$ grid, and mark obstacles as "blocked" cells inside the grid. Moreover, we represent the robot and the boxes as 1×1 squares, each occupying a single cell. An example is illustrated in the figure on the right. In this figure, blocked cells are shown in gray, and the positions of the robot and a box are marked by r and s, respectively.

We call a grid cell "free" if it is not blocked, nor it is occupied by a box. At each step, the robot can move from its current position to one of its four neighboring cells, located at the left, right, top, and bottom of the current cell, provided that the neighboring cell is free. If the neighboring cell is occupied by a box, the robot can push the box forward in the same direction that the robot is moving, provided that the cell to which the box enters is free. Obviously, the robot and the boxes can never cross the boundary of the grid.

Suppose that there is only one box inside the factory, located at a start position s. Assume that we want to move the box to a target position t using the robot. We say that t is "reachable" from s if there is a sequence of moves for the robot to push the box from position s to position t. For example, in the figure above, the leftmost cell marked by t is reachable from s, but the other cell at position t' is not reachable. Given a grid containing a box at position s and a robot at position r, your task is to write a program to check how many of the grid cells are reachable from s using the robot. Note that the cell s itself is always counted as a reachable cell, even if the robot can never reach the box.
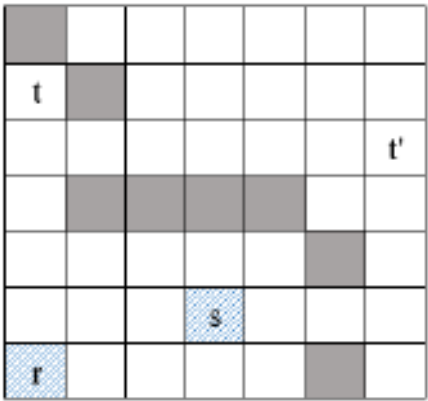
Input

There are multiple test cases in the input. The first line of each test case contains two positive integers m and n ($1 \le m, n \le 1000$), which indicate the size of the test case grid, $m \times n$. Each of the next m lines contains a string of length n, where the j-th character of the i-th line represents the cell (i, j) of the grid. Obstacles are represented by character 'o'. The position of the robot and the start position of the box are represented by characters 'r' and 's', respectively. Other cells are filled by – characters. The input terminates with a line containing '0 0' which should not be processed.

Output

For each test case, output a line containing the number of grid cells reachable from the start position of the robot and box.

Sample Input

```
7 7
o-----
-o----
------
-oooo-
-----o
---s--
r----o
3 4
---o
-os-
---r
0 0
```

Sample Output

```
21
6
```