Pac-man meets Popeye

You are working on a twist on the pac-man game where there are no ghosts on the rectangular board, but pac-man must travel from a cell labeled A to a cell labeled B as fast as possible, or in other words reaching B from A in the fewest number of moves possible (cell A and B are different).

At every move, pac-man can go right, left, up or down one cell unless there is a barrier marked by an X blocking it. Pacman cannot move diagonally, but the board wraps around horizontally and vertically, meaning, for example, if pac-man is on the first row and it goes up it will emerge on the same column but on the bottom row, but this happens only if there is no barrier marked by an X on the cell where it would emerge. Same applies to all other border rows and columns, so, for instance if pac-man is on the last column, it could go right and could emerge on the first column and the same row unless there is an X in the cell where it would emerge. **Basically pac-man can never occupy a cell that has an X.**

This special pac-man game has one more twist. Some cells, labeled S have spinach in them that will give pac-man extra energy such that in addition to the existing moves he can also:

1. move diagonally one cell unless the cell where it would emerge does have an X and the move is valid unless a wrap-around is not required (in other words the diagonal moves cannot wrap around)
2. move horizontally and vertically two spaces as soon as the cell where it would emerge does not have an X, but it can jump over a cell that has an X and it can wrap around!

Calculate the fewest number of moves needed to get from cell A to cell B.

Input

This first line of input contains two integers m and n (1 < =n, m <= 100), where n is the number of rows in the board representation, and m is the number of columns.

Each of the next m lines contains a string of length exactly n, consisting only of the following characters: ‘’.’ – denoting an empty cell (we use the dot instead of the space to make the I/O easier), ‘A’ the starting cell (there is exactly one A cell in the board), ‘B’ the end cell (there is exactly one A cell in the board), ‘X’ – denoting a wall and ‘S’ a cell that has spinach.

Output

An integer denoting the fewest number of moves from A to B or -1 if no path exists.

Sample input 1

```
4 4
A...
XXX.
....
...B
```

Sample output 1

```
2
```

Sample input 2

```
1 4
AXXB
```

Sample output 2

```
1
```

Sample input 3

```
4 4
AS..
XXX.
.B..
....
```

Sample output 3

```
2
```

Sample input 4

```
4 4
....
XXXB
```

```
.ASX
....
```

Sample output 4

```
2
```

Sample input 5

```
4 4
..BX
XXXX
..AS
.XXX
```

Sample output 5

```
6
```

Sample input 6

```
4 5
ASSSS
SSSSS
SSSSS
SSSBS
```

Sample output 6

```
2
```

Sample input 7

```
4 5
.X...
XAX..
.X...
...BS
```

Sample output 7

-1