

Wandering Queen

There is a checkmates board with **n** rows and **m** columns. Some of the cells of the board are occupied. There is a queen standing on a certain cell. It wants to move to another cell of this board. Help it do this making the least possible moves. The queen can go any number of cells in any of eight directions in a single move, but it can't pass through or stand on the occupied cells and leave the board.

Input

The first line of the input contains number **t** – the amount of tests. Then **t** test descriptions follow. The first line of each test consists of two numbers **n** and **m** separated with a space. Then **n** lines follow each containing **m** characters describing the board. Character '.' means a free cell, character 'X' – an occupied cell, character 'S' – the starting cell of the queen, character 'F' – the cell where the queen wants to go. It is guaranteed that there will be exactly one character 'S' and one character 'F' on each board.

Constraints

$1 \leq t \leq 30$

$2 \leq n, m \leq 1000$

Output

For each test case print the minimum number of moves the queen has to do to reach the desired cell. Print '-1' if the queen can't reach the cell.

Example

Input:

```
3
3 3
S..
...
..F
3 3
S..
XX.
F..
3 3
S..
XXX
..F
```

Output:

```
1
3
-1
```