

Frogger

Philip J. Frog just wanted to go for a mid-afternoon swim, but in typical frog fashion he's ended up in the middle of a busy street. Help Phil figure out how long he'll be hopping on hot asphalt before he finds his way to the nice cool water.

Phil may hop one square horizontally or vertically per second. He may only hop onto road, grass, or water. Additionally, he cannot occupy any square occupied by a car. Phil and the cars move at the same time, meaning Phil can “hop over” an oncoming car. Phil can also remain in the same square if he wishes. All horizontal movement wraps (e.g., a rightward hop from the rightmost column places Phil in the leftmost column). Cars move horizontally in the direction indicated on the map ('<' means leftward, '>' means rightward) at a rate of one square per second and never collide with anything.

Input

Input begins with a single integer specifying the number of test maps. Each map begins with two integers R and C ($0 < R$, $C \leq 30$) specifying the number of rows and columns, respectively, followed by R lines each C characters long, specifying the map. The possible map characters are:

Phil ('&') - Phil's starting location. Each map contains exactly one. Always indicates road underneath.

Tree ('T') - Impassable.

Grass ('.') - Phil can move freely in the grass.

Road ('-') - Hot!

Car ('<', '>') - Always indicates road underneath.

Water ('~') - Phil's goal.

Output

For each map, output a line containing the fewest number of seconds Phil must spend on the road in order to reach the water, or the string “Impassable”, if no path to water exists.

Sample Input

```
3
2 1
~
&
4 7
~TTTTTT
.-----
-->--<--
---&---
3 5
~~~~~
..T..
>>&<<
```

**Sample Output**

```
1
6
Impassable
```