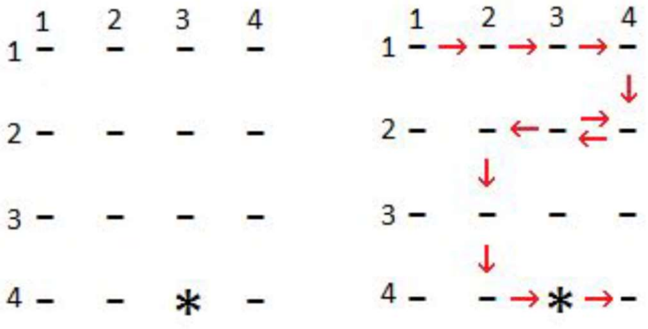
Morsican Navigation Problem

One day, while you're sitting at your desk working on your latest programming project, you hear a strange whirring sound. You spin around quickly, looking for the source of the noise as it grows louder and see the TARDIS appear in your living room. You hold your breath as the door opens and the Doctor leans out. "Well, what are you waiting for?", he says, "Get in! We haven't got all day!" You jump up, leaving your project behind without a thought, and enter the time ands pace traveling box.

On your first journey, you travel to the Morsican System, home of the Morsicus race. The Morsicans are threatening an invasion of Earth to feast on the hopes and dreams of aspiring programmers, which is their main source of food. Unfortunately, the TARDIS′ navigational system is down… again. You successfully made it to the edge of the planetary system, but there′s no good way to get to the Morsican home planet while the navigation system is still down. Luckily for you, this system contains mostly empty space and the only object of any legitimate size is the Morsican planet. You simply need to get close so that you can see it. The Doctor goes below the console to try and repair the system, but you decide to take matters into your own hands and write a program to navigate to the planet yourself.

First, ask the user for the row and column of the planet. Then, read in a path (a path is a string of characters symbolizing up (u), down (d), left (l), and right(r)). Starting in the upper left hand square of the map (which is the space (1,1)), you should test the path to see if it will succeed in getting you to the planet.

You may assume the path will never go off the edge of the map. If you end on a space marked by a *, your journey is successful and you will be able to meet the Morsicans (so you should print a statement letting you know)! Otherwise, your journey fails and you should print out that you didn′t make it to the planet. If you pass over the planet but don't finish on it, you do not find the planet.



The first number represents the row location of the planet
The second number represents the column location of the planet
The strings of directions are the attempts to get to the planet starting at (1,1) … top left corner

Input: (navigation.txt)

```
4
3
rrrdlrllddrr
ddrurdd
```

Output:

```
rrrdlrllddrr  -> You didn’t get to the planet
ddrurdd -> You got to the planet
```