 Practice   Compete   Jobs   Rank   Leaderboard

Dashboard > Algorithms > Dynamic Programming > Fairy Chess

Badge Progress (Details)

Points: 126   Rank: 270330

Fairy Chess 🔖

 by **HackerRank**

| Problem | Submissions | Leaderboard | Discussions | Editorial 🔒 |

Let's play *Fairy Chess*!

You have an $n \times n$ chessboard. An s-leaper is a chess piece which can move from some square (x_0, y_0) to some square (x_1, y_1) if $abs(x_0 - x_1) + abs(y_0 - y_1) \le s$; however, its movements are restricted to *up* (↑), *down* (↓), *left* (←), and *right* (→) within the confines of the chessboard, meaning that diagonal moves are not allowed. In addition, the leaper cannot leap to any square that is occupied by a *pawn*.

Given the layout of the chessboard, can you determine the number of ways a leaper can move m times within the chessboard?

Note: $abs(x)$ refers to the absolute value of some integer, x.

Input Format

The first line contains an integer, q, denoting the number of queries. Each query is described as follows:

1. The first line contains three space-separated integers denoting n, m, and s, respectively.

2. Each line i of the n subsequent lines contains n characters. The jth character in the ith line describes the contents of square (i, j) according to the following key:

   - . indicates the location is *empty*.

   - P indicates the location is occupied by a *pawn*.

   - L indicates the location of the *leaper*.

Constraints

- $1 \le q \le 10$

- $1 \le m \le 200$

- There will be exactly one L character on the chessboard.

- The s-leaper can move *up* (↑), *down* (↓), *left* (←), and *right* (→) within the confines of the chessboard. It *cannot* move diagonally.

Output Format

For each query, print the number of ways the leaper can make m moves on a new line. Because this value can be quite large, your answer must be modulo $10^9 + 7$.

Sample Input 0

```
3
4 1 1
....
.L..
.P..
....
3 2 1
...
...
..L
```
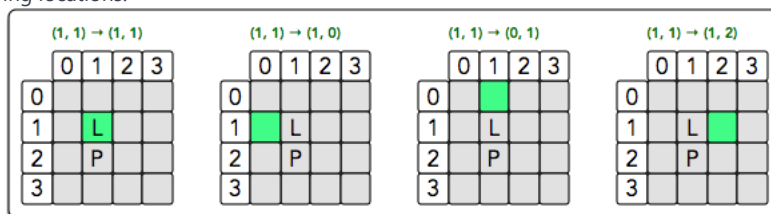
```
4 3 2
....
...L
..P.
P...
```

Sample Output 0

```
4
11
385
```

Explanation 0

You must perform two queries, outlined below. The *green* cells denote a cell that was leaped to by the leaper, and coordinates are defined as $(row, column)$.

1. The leaper can leap to the following locations:

Observe that the leaper cannot leap to the square directly underneath it because it's occupied by a pawn. Thus, there are **4** ways to make **1** move and we print **4** on a new line.

2. The leaper can leap to the following locations:
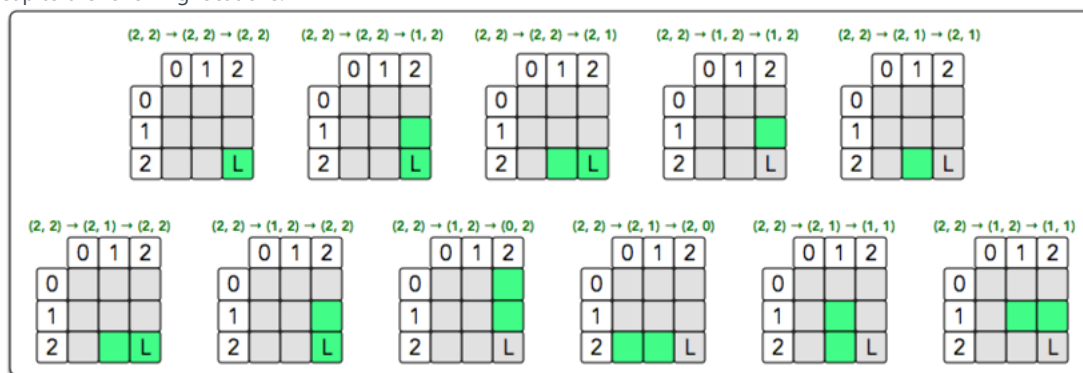
Thus, we print **11** on a new line.

Note: Don't forget that your answer must be modulo $10^9 + 7$.

Submissions:591

Max Score:80

Difficulty: Advanced

Rate This Challenge:

☆ ☆ ☆ ☆ ☆

More

| Current Buffer (saved locally, editable) ⑂ 🕘 | Java 7 |

```
1 ▾ import java.io.*;
2   import java.util.*;
3   import java.text.*;
4   import java.math.*;
5   import java.util.regex.*;
```

```
 6
 7 ▼ public class Solution {
 8
 9 ▼     public static void main(String[] args) {
10 ▼         /* Enter your code here. Read input from STDIN. Print output to STDOUT. Your class should be named Solution. */
11       }
12 }
```

Line: 1 Col: 1

⬆ Upload Code as File   ☐ Test against custom input   Run Code   Submit Code

Join us on IRC at #hackerrank on freenode for hugs or bugs.

Contest Calendar | Blog | Scoring | Environment | FAQ | About Us | Support | Careers | Terms Of Service | Privacy Policy | Request a Feature