

[All Domains](#) > [Artificial Intelligence](#) > [Bot Building](#) > Bot saves princess - 2

Points: 13.90 Rank: 20930

Bot saves princess - 2

by [dheeraj](#)

Problem

Submissions

Leaderboard

Discussions

In this version of "Bot saves princess", Princess Peach and bot's position are randomly set. Can you save the princess?

Task

Complete the function `nextMove` which takes in 4 parameters - an integer `N`, integers `r` and `c` indicating the row & column position of the bot and the character array `grid` - and outputs the **next** move the bot makes to rescue the princess.

Input Format

The first line of the input is `N` (<100), the size of the board (`NxN`). The second line of the input contains two space separated integers, which is the position of the bot.

Grid is indexed using [Matrix Convention](#)

The position of the princess is indicated by the character 'p' and the position of the bot is indicated by the character 'm' and each cell is denoted by '-' (ascii value: 45).

Output Format

Output only the **next** move you take to rescue the princess. Valid moves are LEFT, RIGHT, UP or DOWN

Sample Input

```
5
2 3
-----
-----
p--m-
-----
-----
```

Sample Output

```
LEFT
```

Resultant State

```
-----
-----
p-m--
-----
-----
```

Explanation

As you can see, bot is one step closer to the princess.

Scoring

Your score for every testcase would be $(N \times N - \text{number of moves made to rescue the princess}) / 10$ where `N` is the size of the grid (5x5 in the sample testcase). Maximum score is 17.5



Solved score: 17.50pts



Submissions: 14643

Max Score: 17

Difficulty: Easy

[More](#)

Current Buffer (saved locally, editable)  

Java 7  

```
8 import java.io.*;
9 import java.util.*;
10 import java.text.*;
11 import java.math.*;
12 import java.util.regex.*;
13
14 public class Solution {
15     static void nextMove(int n, int r, int c, String [] grid){
16     }
17     public static void main(String[] args) {
18         Scanner in = new Scanner(System.in);
19         int n,r,c;
20         n = in.nextInt();
21         r = in.nextInt();
22         c = in.nextInt();
23         in.useDelimiter("\n");
24         String grid[] = new String[n];
25
26
27         for(int i = 0; i < n; i++) {
28             grid[i] = in.next();
29         }
30
31         nextMove(n,r,c,grid);
32     }
33 }
34 }
35 }
```

Line: 1 Col: 1

 [Upload Code as File](#)

Run Code

Submit Code

Copyright © 2016 HackerRank. All Rights Reserved

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](#)