**Trace Path**

Show Topic Tags   

There is the Rectangular path for a Train to travel consisting of n and m rows and columns respectively. The train will start from one of grid cells and it will be given a command in the form of String s. consisting of characters ‘L’, ‘R’, ‘U’, ‘D’.The train will follow the instructions of the command string, where 'L' corresponds moving to the left, 'R' towards the right, 'U' for moving up, and 'D' means down.  
You have already selected the command string s, and are wondering if it is possible to place the train in one of the grid cells initially and have it always stay entirely within the grid upon execution of the command string s. Output “1” if there is a starting cell for which the train doesn’t fall off the grid(track) on following command s, otherwise, output "0".

**Input:**  
The first line of input will contain an integer T, the number of test cases.  
Each test case will be on two lines.  
The first line will have two space separated integers n,m.  
The second line will have the command string s.

**Output:**  
For each test case, output "1" (without quotes) or "0" (without quotes) in a new line.

**Constraints:**  
1 <= T <= 1,00  
1 <= n,m <= 10  
1 <= |s| <= 10

**Example:  
Input:**  
2  
1 1  
R  
2 3  
LLRU  
**Output:**  
0  
1

\*\*For More Examples Use Expected Output\*\*

Contributor: Saksham Raj Seth

<http://practice.geeksforgeeks.org/problems/trace-path/0>

#include <iostream>

#include <stdio.h>

#include <vector>

using namespace std;

bool haycamino(int n, int m, string s) {

for(int fila = 1; fila <= n; fila++) {

for(int col =1 ; col <=m; col ++) {

int fi = fila, ci = col;

bool paso = true;

for(int i =0; i<s.size(); i++) {

//‘L’, ‘R’, ‘U’, ‘D’

if(s[i] == 'L') {

ci--;

if(ci < 1) {

paso = false;

break;

}

}

else if(s[i] == 'R') {

ci++;

if(ci > m) {

paso = false;

break;

}

}

else if(s[i] == 'U') {

fi--;

if(fi < 1) {

paso = false;

break;

}

}

else if(s[i] == 'D') {

fi++;

if(fi > n) {

paso = false;

break;

}

}

}

if(paso ) {

return true;

}

}

}

return false;

}

int main() {

//code

int t;

scanf("%d", &t);

while(t-- > 0) {

int n,m;

scanf("%d %d", &n, &m);

string s;

cin >> s;

cout << haycamino(n,m,s) << endl;

}

/\*

int n = 2, m = 3;

string s = "LLRU";

cout << haycamino(n,m,s) << endl;

system("pause");\*/

return 0;

}