Week 6: Assignment 6 - Question 3

Recursive Path Finding

Given an $n \times n$ binary Matrix A , where each entry is 0 or 1.
A has a unique path of 1's from $A[0][0]$ to $A[n\text{-}1][n\text{-}1]$.
The path can go Right (R) Left (R) Down (D) or Up (U).

Write a C Program.to print the directions of this path.

Note: You can assume that there is exactly one correct path.
All 1's in A need not be in this unique path, there can be dead ends.

Input
The first line contains the dimension of the matrix n. Assume n < 100.
The second line contains the contents of the matrix A, each row per line.

Output
The path of 1's in the Matrix.

Example

Input

```
6
1 1 1 1 0 0
0 1 0 0 0 0
0 1 0 1 1 1
1 1 1 1 0 1
0 0 0 0 0 1
0 0 0 0 0 1
```

Output

```
RDDDRRURRDDD
```

Explanation

The correct path of 1's from $A[0][0]$ to $A[5][5]$ is

A[0][0] Right --> A[0][1] Down --> A[1][1] Down --> A[2][1] Down --> A[3][1] Right --> A[3][2] Right --> A[3][3] Up --> A[2][3] Right --> A[2][4] Right --> A[2][5] Down --> A[3][5] Down --> A[4][5] Down --> A[5][5].

1 **1** 1 1 0 0
0 **1** 0 0 0 0
0 **1** 0 **1** **1** **1**
1 **1** **1** **1** 0 **1**
0 0 0 0 0 **1**
0 0 0 0 0 **1**

Note: The code for reading inputs etc is given to you, complete the code of the function

*void **findPath**(int matrix[100][100], int n, int x, int y, char* path, int pathIndex);*

Hint

Try all the paths LRDU one by one recursively [except the opposite of last direction taken].
If any of the recursive calls succeed, the function succeeds, return '1' immediately.
If all of the recursive calls fail, the function fails, return 0.