Rat in a Maze Problem

Consider a rat placed at **(0, 0)** in a square matrix of order **N * N**. It has to reach the destination at **(N - 1, N - 1)**. Find all possible paths that the rat can take to reach from source to destination. The directions in which the rat can move are **'U'(up)**, **'D'(down)**, **'L' (left)**, **'R' (right)**. Value 0 at a cell in the matrix represents that it is blocked and rat cannot move to it while value 1 at a cell in the matrix represents that rat can be travel through it.

Note: In a path, no cell can be visited more than one time. If the source cell is 0, the rat cannot move to any other cell.

Example 1:

```
Input:
N = 4
m[][] = {{1, 0, 0, 0},
         {1, 1, 0, 1},
         {1, 1, 0, 0},
         {0, 1, 1, 1}}
Output:
DDRDRR DRDDRR
Explanation:
The rat can reach the destination at
(3, 3) from (0, 0) by two paths - DRDDRR
and DDRDRR, when printed in sorted order
we get DDRDRR DRDDRR.
```

Example 2:

```
Input:
N = 2
m[][] = {{1, 0},
         {1, 0}}
```

```
Output:
-1
Explanation:
No path exists and destination cell is
blocked.
```

Your Task:

You don't need to read input or print anything. Complete the
function **printPath()** which takes **N** and 2D array **m[ ][ ]** as input parameters and
returns the list of paths in lexicographically increasing order.

Note: In case of no path, return an empty list. The driver will output **"-
1"** automatically.

Expected Time Complexity: $O((3^{N^2}))$.
Expected Auxiliary Space: $O(L * X)$, L = length of the path, X = number of paths.

Constraints:

$2 \leq N \leq 5$
$0 \leq m[i][j] \leq 1$