

# Maze

Problem

Submissions

Leaderboard

Discussions

## Problem Statement

Rezia is trapped in a 2D maze of  $N \times M$  size, starting at position R, and her goal is to reach the exit marked by D. The maze contains blocks represented by #, and she can only traverse through cells marked with dots (.). As she need to escape as early as possible, we need to determine the path she will follow.

Place an X in each cell representing Rezia's route to exit the maze. If there is no viable path for her to exit, leave the maze unchanged.

Note: Rezia can move in four directions – right, left, up, and down. It is crucial to adhere to the specified order: attempting right first, then left, followed by up, and finally down.

## Input Format

- First line will contain  $N$  and  $M$ .
- Next you will be given the 2D matrix.

## Constraints

1.  $1 \leq N, M \leq 10^3$

## Output Format

- Output the final maze with marked X indicating the path she will follow.

## Sample Input 0

```

5 6
...D.#
.##...#
....#.
.R#...
.#.##.
    
```

## Sample Output 0

```

...D.#
.##X.#
.XXX#.
.R#...
.#.##.
    
```

## Sample Input 1

```
5 6
...D.#
.R...#
....#
.#...
.#.##.
```

#### Sample Output 1

```
...D.#
.RXX.#
....#
.#...
.#.##.
```

#### Sample Input 2

```
5 6
...D.#
....#
.##.#.
.R#...
.#.##.
```

#### Sample Output 2

```
...D.#
XXXX.#
X##.#.
XR#...
.#.##.
```

#### Sample Input 3

```
5 6
...D.#
....#
###.#.
.R#...
.#.##.
```

#### Sample Output 3

```
...D.#
....#
###.#.
.R#...
.#.##.
```

[f](#) [t](#) [in](#)

Submissions: [63](#)

Max Score: 25

Difficulty: Easy

Rate This Challenge:

☆☆☆☆☆

[More](#)

```
1 #include <bits/stdc++.h>
2 using namespace std;
3
4
5 const int max_n = 1000+1;
6
7 char grid[max_n][max_n];
8 bool vis[max_n][max_n];
9 //                                right    left    up    down
10 vector<pair<int, int>> movement = {{0, -1}, {0, 1}, {-1, 0}, {1, 0}};
11
12 int n,m;
13 bool dfs(int x, int y){
14     if (grid[x][y]=='D') return true;
15     grid[x][y] = 'X';
16
17     for (auto i: movement){
18         int cx = x+i.first;
19         int cy = y+i.second;
20         if (cx<0 || cx>=n || cy<0 || cy>=m || grid[cx][cy]=='X' || grid[cx][cy]=='#') continue;
21
22         if (dfs(cx, cy)) return true;
23     }
24
25     grid[x][y] = '.';
26     return false;
27 }
28
29 int main(){
30     memset(vis, false, sizeof(vis));
31     // input
32     cin >> n >> m;
33     int srcx, srcy;
34
35     for (int x=0; x<n; x++){
36         for (int y=0; y<m; y++){
37             cin >> grid[x][y];
38             if (grid[x][y]=='R'){
39                 srcx = x;
40                 srcy = y;
41             }
42         }
43     }
44
45     dfs(srcx, srcy);
46     grid[srcx][srcy] = 'R';
47
48     // cout << endl;
49     for (int x=0; x<n; x++){
50         for (int y=0; y<m; y++){
51             cout << grid[x][y];
52         }
53         cout << endl;
54     }
55     return 0;
56 }
57 }
```

Line: 49 Col: 21

Testcase 0 ❌

Testcase 1 ❌

Testcase 2 ❌

Testcase 3 ✅

Your code did not pass this test case.

#### Compile Message

```
Solution.cpp: In function 'int main()':  
Solution.cpp:47:22: warning: 'srcy' may be used uninitialized in this function [-Wmaybe-uninitialized]  
    grid[srcx][srcy] = 'R';  
    ~~~~~^~~~~~  
Solution.cpp:47:22: warning: 'srcx' may be used uninitialized in this function [-Wmaybe-uninitialized]
```

Compile Time

#### Input (stdin)

```
5 6  
...D.#  
.##...#  
....#.  
.R#...  
#.##.
```

Run Time

#### Your Output (stdout)

```
...DX#  
.##XX#  
XXXX#.  
XR#...  
#.##.
```

#### Expected Output

```
...D.#  
.##X.#  
.XXX#.  
.R#...  
#.##.
```

#### Compiler Message

Wrong Answer