

Description

Solution

Discuss (561)

Submissions

C++

Auto

490. The Maze

Medium

1556

161

Add to List

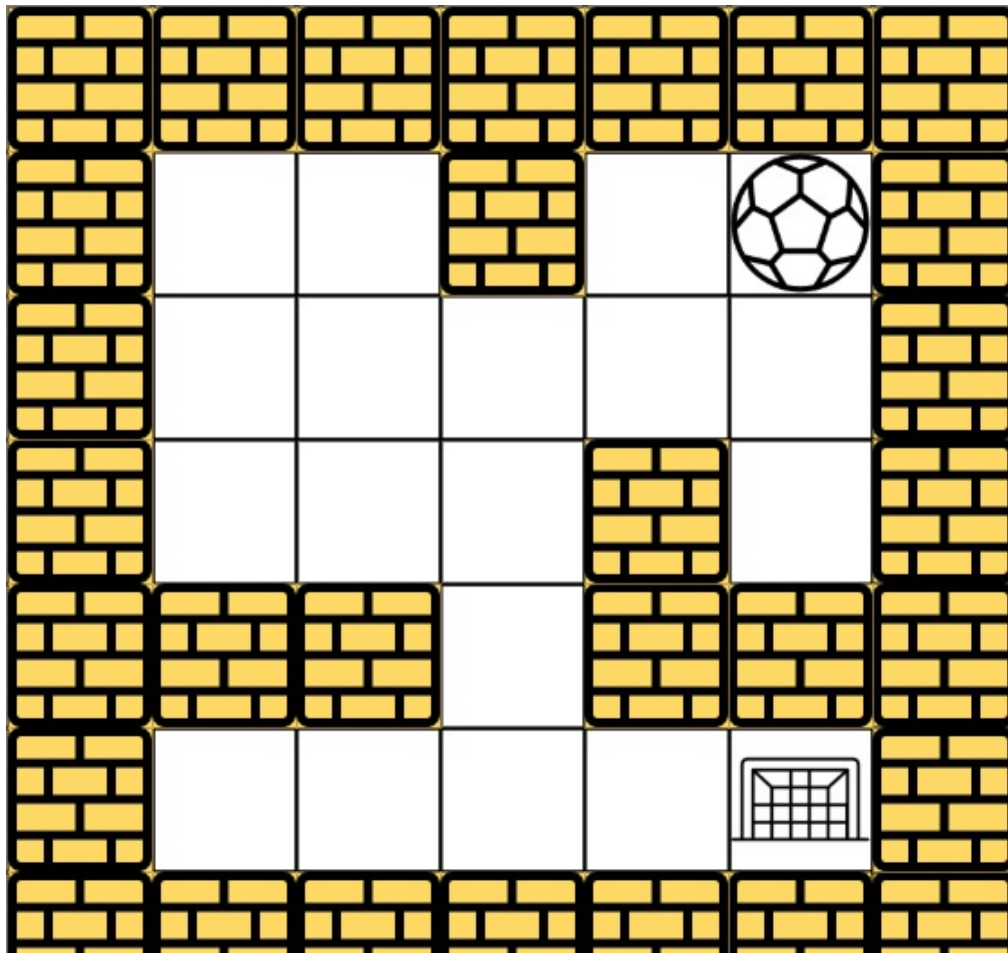
Share

There is a ball in a `maze` with empty spaces (represented as `0`) and walls (represented as `1`). The ball can go through the empty spaces by rolling **up**, **down**, **left** or **right**, but it won't stop rolling until hitting a wall. When the ball stops, it could choose the next direction.

Given the `m x n` `maze`, the ball's `start` position and the `destination`, where `start = [startrow, startcol]` and `destination = [destinationrow, destinationcol]`, return `true` if the ball can stop at the destination, otherwise return `false`.

You may assume that **the borders of the maze are all walls** (see examples).

Example 1:



```

1  class Solution
2      vector<vector<int>> maze;
3  public:
4      bool isValid(int x, int y, int n, int m) {
5          if(x < 0 || x >= n || y < 0 || y >= m) return false;
6          return true;
7      }
8      bool canWeReach(vector<vector<int>> &maze, int x, int y, vector<int> destination, int i) {
9          if(maze[x][y] == 1) return false;
10         if(x == destination[0] && y == destination[1]) return true;
11         if(i >= 4) return false;
12         vector<int> dirs = {0, 1, 0, -1};
13         for(int j = 0; j < 4; j++) {
14             int newx = x + dirs[j];
15             int newy = y + dirs[j+1];
16             while(isValid(newx, newy, maze)) {
17                 newx++;
18                 newy++;
19                 if(newx == destination[0] && newy == destination[1]) return true;
20             }
21             dirs[j] = 0;
22             dirs[j+1] = 0;
23         }
24         return false;
25     }
26     bool canWeReach(vector<vector<int>> &maze, int x, int y, vector<int> destination) {
27         return canWeReach(maze, x, y, destination, 0);
28     }
29     bool canWeReach(vector<vector<int>> &maze, int x, int y, vector<int> destination) {
30         return canWeReach(maze, x, y, destination, 0);
31     }

```

Your previous code was re

Problems

Pick One

< Prev

1/149

Next >

Contribute i

Run Code