



Courses

Practice

Roadmap

Pro



# Algorithms and Data Structures for Beginners

29 / 35

## 29 - Matrix DFS

### Matrix - DFS

0	0	0	0
1	1	0	0
0	0	0	1
0	1	0	0

Q: Count the unique paths from the top left to the bottom right. A single path may only move along 0's and can't visit the same cell more than once.

```
# Count paths (backtracking)
def dfs(grid, r, c, visit):
    rows, cols = len(grid), len(grid[0])
    if (min(r, c) < 0 or
        r == rows or c == cols or
        (r, c) in visit or grid[r][c] == 1):
        return 0
    if r == rows - 1 and c == cols - 1:
        return 1

    visit.add((r, c))

    count = 0
    count += dfs(grid, r + 1, c, visit)
    count += dfs(grid, r - 1, c, visit)
    count += dfs(grid, r, c + 1, visit)
    count += dfs(grid, r, c - 1, visit)

    visit.remove((r, c))
    return count

print(dfs(grid, 0, 0, set()))
```



Mark Lesson Complete

View Code

Prev

Next

## 27 Hash Implementation

## Suggested Problems

Status	Star	Problem 	Difficulty 	Video Solution	Code
<input type="checkbox"/>		Number of Islands	Medium		
<input type="checkbox"/>		Max Area of Island	Medium		

Copyright © 2023 NeetCode.io All rights reserved.

Contact: [neetcodebusiness@gmail.com](mailto:neetcodebusiness@gmail.com)

[Github](#) [Privacy](#) [Terms](#)