

59 螺旋矩阵 II

题目描述

给定一个正整数 n ，生成一个包含 1 到 n^2 所有元素，且元素按顺时针顺序螺旋排列的正方形矩阵。

示例:

```
输入: 3
输出:
[
  [ 1, 2, 3 ],
  [ 8, 9, 4 ],
  [ 7, 6, 5 ]
]
```

代码

```
class Solution:
    def generateMatrix(self, n: int) -> List[List[int]]:
        spiral = [[0 for _ in range(n)] for _ in range(n)]
        row, col = 0, 0
        d_r, d_c = 0, 1
        count = 1
        while count <= n * n:
            spiral[row][col] = count
            count += 1
            if row + d_r < 0 or row + d_r >= n or col + d_c < 0 or col + d_c >= n or spiral[row+d_r][col+d_c] != 0:
                d_r, d_c = d_c, -d_r
            row += d_r
            col += d_c

        return spiral
```

成功 [显示详情](#) >

执行用时 : **52 ms**, 在Spiral Matrix II的Python3提交中击败了**87.47%** 的用户

内存消耗 : **13.1 MB**, 在Spiral Matrix II的Python3提交中击败了**57.84%** 的用户

进行下一个挑战 :

[螺旋矩阵](#)

炫耀一下:    

提交时间	状态	执行用时	内存消耗	语言
几秒前	通过	52 ms	13.1 MB	python3
19 分钟前	通过	68 ms	13.2 MB	python3
19 分钟前	通过	72 ms	13 MB	python3