1 Leetcode64 Minimun Path Sum 二维坐标型动规

笔记本： Dynamic Programing
创建时间： 10/14/2019 3:27 PM **更新时间：** 10/20/2019 12:22 PM
作者： tanziqi1756@outlook.com

64. Minimum Path Sum

难度 中等 300 收藏 分享 切换为中文

题目描述 评论 (322) 题解(76) New 提交记录

Given a *m* x *n* grid filled with non-negative numbers, find a path from top left to bottom right which *minimizes* the sum of all numbers along its path.

Note: You can only move either down or right at any point in time.

Example:

```
Input:
[
  [1,3,1],
  [1,5,1],
  [4,2,1]
]
Output: 7
Explanation: Because the path 1→3→1→1→1 minimizes the sum.
```

```java
    public int minPathSum(int[][] grid) {
        int height = grid.length;
        int width = grid[0].length;
        int[][] dp = grid;
        // initialize
        for( int j = 1; j < width; j++ ) {
            dp[0][j] += dp[0][j-1];
        }
        for( int i = 1; i < height; i++ ) {
            dp[i][0] += dp[i-1][0];
        }
        // dp
        for( int i = 1; i < height; i++ ) {
            for( int j = 1; j < width; j++ ) {
                dp[i][j] += Math.min(dp[i-1][j], dp[i][j-1]);
            }
        }
        for( int i = 0; i < height; i++ ) {
            System.out.println(Arrays.toString(dp[i]));
        }
        return dp[height-1][width-1];
    }
}
```

您上次编辑到这里，代码已从您浏览器本地的临时存储中恢复了 还原默认代码模版

测试用例 | 代码执行结果

已完成 执行用时: 0 ms

输入	[[1,3,1],[1,5,1],[4,2,1]]
输出	7
预期结果	7
stdout	[1, 4, 5] [2, 7, 6] [6, 8, 7]