

## 64 最小路径和

Label: 动态规划

给定一个包含非负整数的  $m \times n$  网格 `grid`，请找出一条从左上角到右下角的路径，使得路径上的数字总和为最小。

说明：每次只能向下或者向右移动一步。

输入: `grid = [[1,3,1],[1,5,1],[4,2,1]]`

输出: 7

1	3	1
1	5	1
4	2	1

- 动态规划

```
class Solution {
    public int minPathSum(int[][] grid) {
        int m = grid.length;
        int n = grid[0].length;
        int[][] path = new int[m][n];
        for (int i = 0; i < m; i++) { // 行
            for (int j = 0; j < n; j++) {
                if (i == 0 && j == 0) {
                    path[i][j] = grid[i][j];
                } else if (i == 0 && j != 0) {
                    path[i][j] = path[i][j-1] + grid[i][j];
                } else if (i != 0 && j == 0) {
                    path[i][j] = path[i-1][j] + grid[i][j];
                } else { // i != 0 && j != 0
                    path[i][j] = Math.min(path[i][j-1], path[i-1][j]) + grid[i][j];
                }
            }
        }
        return path[m-1][n-1];
    }
}
```