

• 动态规划：64. 最小路径和

题目图解：

64. 最小路径和

已解答

中等 相关标签 相关企业 Ax

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。

示例 1：

1	3	1
1	5	1
4	2	1

输入: `grid = [[1,3,1],[1,5,1],[4,2,1]]`  
输出: 7  
解释: 因为路径 1→3→1→1→1 的总和最小。

1	3	1
1	5	1
4	2	1

INT_MAX	INT_MAX	INT_MAX	INT_MAX
0	$\min(dp[i-1][j], dp[i][j-1]) + g[i-1][j-1]$	$\min(dp[i-1][j], dp[i][j-1]) + g[i-1][j-1]$	$\min(dp[i-1][j], dp[i][j-1]) + g[i-1][j-1]$
INT_MAX	$\min(dp[i-1][j], dp[i][j-1]) + g[i-1][j-1]$	$\min(dp[i-1][j], dp[i][j-1]) + g[i-1][j-1]$	$\min(dp[i-1][j], dp[i][j-1]) + g[i-1][j-1]$
INT_MAX	$\min(dp[i-1][j], dp[i][j-1]) + g[i-1][j-1]$	$\min(dp[i-1][j], dp[i][j-1]) + g[i-1][j-1]$	$\min(dp[i-1][j], dp[i][j-1]) + g[i-1][j-1]$

最终结果

代码实现：

```
1 class Solution {
2 public:
3     int minPathSum(vector<vector<int>>& grid) {
4         int row = grid.size();
5         int col = grid[0].size();
6         vector<vector<int>> dp(row+1,vector<int>(col+1,INT_MAX));
7         dp[1][0] = 0;
8         for(int i = 1;i <= row ;i++)
9         {
10             for(int j = 1; j <= col;j++)
11             {
12                 dp[i][j] = min(dp[i-1][j],dp[i][j-1]) + grid[i-1][j-1];
13             }
14         }
15         return dp[row][col];
16
17     }
18 };
```