LeetCode

Introduction

Algorithms

Depth-first Search

Breadth-first Search

Union Find

Tree

Dynamic Programming

Topological Sort

Substring Problem Template

Kadane's Algorithm

KMP

Fenwick Tree or Binary Indexed Tree

Segment Tree Range Minimum Query



62. Unique Paths (Medium)

A robot is located at the top-left corner of a $m \times n$ grid (marked 'Start' in the diagram below).

The robot can only move either down or right at any point in time. The robot is trying to reach the bottom-right corner of the grid (marked 'Finish' in the diagram below).

How many possible unique paths are there?

Above is a 3 x 7 grid. How many possible unique paths are there?

Note: *m* and *n* will be at most 100.

Solution 1: DP

这道题让求所有不同的路径的个数,一开始还真把我难住了,因为之前好像没有遇到过这类的问题,所以感觉好像有种无从下手的感觉。在网上找攻略之后才恍然大悟,原来这跟之前那道 Climbing Stairs 爬梯子问题 很类似,那道题是说可以每次能爬一格或两格,问到达顶部的所有不同爬法的个数。而这道题是每次可以向下走或者向右走,求到达最右下角的所有不同走法的个数。那么跟爬梯子问题一样,我们需要用动态规划 Dynamic Programming来解,我们可以维护一个二维数组dp,其中dp[i][j]表示到当前位置不同的走法的个数,然后可以得到递推式为: dp[i][j] = dp[i - 1][j] + dp[i][j - 1],这里为了节省空间,我们使用一维数组dp,一行一行的刷新也可以,代码如下: