C – Dynamic Walking

Chin Ka Wang {rina__owo}

2025-05-16

Background

Problem Idea by `rina__owo`

Preparation by `rina__owo`, `pepper1208`

Problem Restatement

Given a *N* * *M* grid. There may exist some obstacles in the grid.

By implementing the dynamic programming approach with the incomplete transitional formula given, find out the number of ways to travel from the top left corner (1,1) to the bottom right corner (N, M).

Subtasks Constraints

Points are given per checkpoint in this problem. You can get the point of the checkpoint when you pass them.

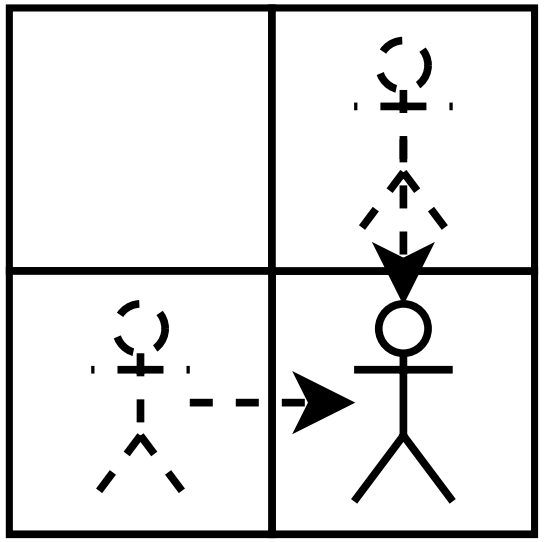
There are 10 checkpoints in total, each carry 10 points.

Why the transitional formula?

$$dp[i][j] = dp[i-1][j] + dp[i][j-1]$$

Why the transitional formula?

- Assume there are no obstacles in the grid.
- You can only move rightwards or downwards 1 step each time. In other words, you can reach the grid from the block leftwards or upwards **only**.
- Therefore, the number of ways travelling to a grid actually equals to the sum of the ways travelling to the block on top of you and the ways travelling to the block on the left of you.



How to deal with obstacles?

- Note that it is impossible to get to a grid having an obstacle.
- We can just simply record the required ways of it to be 0 instead of using the formula.
- In the normal grids, we can just apply the given formula.
- Expected score: **100 AC!**
- Time complexity: O(NM)

How to deal with obstacles?

- Note that it is impossible to get to a grid having an obstacle.
- We can just simply record the required ways of it to be 0 instead of using the formula.
- In the normal grids, we can just apply the given formula.
- Expected score: ~~100 AC!~~ **70** (Why?)
- Time complexity: O(NM)

How to deal with obstacles?

- Remember to use **long long!**
- Expected score: **100 AC!**

Takeaways

- Try to substitute numbers into a formula you haven’t seen before to help you understand more.
- Remember to use **long long**!
- Proficiency in basic syntax is very important.