

Problem 2912: Number of Ways to Reach Destination in the Grid

Problem Information

Difficulty: Hard

Acceptance Rate: 58.23%

Paid Only: Yes

Tags: Math, Dynamic Programming, Combinatorics

Problem Description

You are given two integers `n` and `m` which represent the size of a **1-indexed** grid. You are also given an integer `k`, a **1-indexed** integer array `source` and a **1-indexed** integer array `dest`, where `source` and `dest` are in the form `[x, y]` representing a cell on the given grid.

You can move through the grid in the following way:

* You can go from cell `[x1, y1]` to cell `[x2, y2]` if either `x1 == x2` or `y1 == y2`. * Note that you **can't** move to the cell you are already in e.g. `x1 == x2` and `y1 == y2`.

Return _the number of ways you can reach_ `dest` _from_ `source` _by moving through the grid_ **exactly** `k` _times._

Since the answer may be very large, return it **modulo** `109 + 7`.

Example 1:

Input: n = 3, m = 2, k = 2, source = [1,1], dest = [2,2] **Output:** 2 **Explanation:** There are 2 possible sequences of reaching [2,2] from [1,1]: - [1,1] -> [1,2] -> [2,2] - [1,1] -> [2,1] -> [2,2]

Example 2:

Input: n = 3, m = 4, k = 3, source = [1,2], dest = [2,3] **Output:** 9 **Explanation:** There are 9 possible sequences of reaching [2,3] from [1,2]: - [1,2] -> [1,1] -> [1,3] -> [2,3] - [1,2] ->

```
[1,1] -> [2,1] -> [2,3] - [1,2] -> [1,3] -> [3,3] -> [2,3] - [1,2] -> [1,4] -> [1,3] -> [2,3] - [1,2] -> [1,4]
-> [2,4] -> [2,3] - [1,2] -> [2,2] -> [2,1] -> [2,3] - [1,2] -> [2,2] -> [2,4] -> [2,3] - [1,2] -> [3,2] ->
[2,2] -> [2,3] - [1,2] -> [3,2] -> [3,3] -> [2,3]
```

****Constraints:****

```
* `2 <= n, m <= 109` * `1 <= k <= 105` * `source.length == dest.length == 2` * `1 <= source[1], dest[1] <= n` * `1 <= source[2], dest[2] <= m`
```

Code Snippets

C++:

```
class Solution {
public:
    int numberOfWays(int n, int m, int k, vector<int>& source, vector<int>& dest)
    {

    }
};
```

Java:

```
class Solution {
    public int numberOfWays(int n, int m, int k, int[] source, int[] dest) {
        }

    }
```

Python3:

```
class Solution:
    def numberOfWays(self, n: int, m: int, k: int, source: List[int], dest: List[int]) -> int:
```