62 Unique Paths

Created	@March 7, 2022 4:16 AM
Tags	Medium
link	https://leetcode.com/problems/unique-paths/
Problem Number	62
companies	amazon facebook goldman google salesforce

Description

There is a robot on an `m x n` grid. The robot is initially located at the **top-left corner** (i.e., `grid[0][0]`). The robot tries to move to the **bottom-right corner** (i.e., `grid[m - 1][n - 1]`). The robot can only move either down or right at any point in time.

Given the two integers `m` and `n`, return *the number of possible unique paths that the robot can take to reach the bottom-right corner*.

The test cases are generated so that the answer will be less than or equal to `2 * 10 9`.

```
# uuurrrrrrr 10! / 7!3!

# (m + n - 2) C (n-1)
class Solution:
    def factorial(self, n: int) -> int:
        result = 1

        for i in range(2, n+1):
            result = result * i

        return int(result)

    def combinations(self, m: int, n: int) -> int:
        # calculates mCn
        return int(self.factorial(m) / (factorial(n) * factorial(m-n)))

    def uniquePaths(self, m: int, n: int) -> int:
        return self.combinations(m + n - 2, n - 1) if m > n else self.combinations(m + n - 2, m - 1)
```