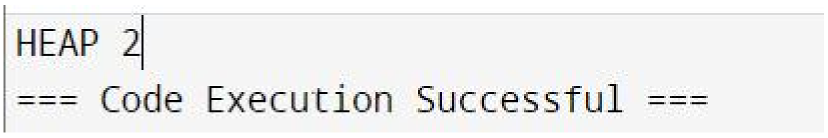
190.There is a robot on an m x n grid. The robot is initially located at the top-left corner (i.e., grid[0][0]). The robot tries to move to the bottom-right corner (i.e., grid[m - 1][n - 1]). The robot can only move either down or right at any point in time. Given the two integers m and n, return the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

PROGRAM:

```
class Solution:

    def uniquePaths(self, m: int, n: int) -> int:

        dp = [[1] * n for _ in range(m)]



        for i in range(1, m):

            for j in range(1, n):

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



        return dp[m - 1][n - 1]
```

OUTPUT:



```
HEAP 2
=== Code Execution Successful ===
```

TIME COMPLEXITY:O(M*N)