

TODO Today : First session : 5 Nov 2023

62. Unique Paths

Medium Topics Companies

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

Given two integers n and m , 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 .

Example 1:

Input: $n = 3$, $m = 7$
Output: 28

$\text{arr}[i][j] = \text{arr}[i-1][j] + \text{arr}[i][j-1]$



63. Unique Paths II

Medium Topics Companies

You are given an $n \times n$ integer array grid . There is a robot initially located at the top-left corner (i.e., $\text{grid}[0][0]$). The robot tries to move to the bottom-right corner (i.e., $\text{grid}[n-1][n-1]$). The robot can only move either down or right at any point in time.

An obstacle and space are marked as 1 or 0 respectively in grid . A path that the robot takes cannot include any square that is an obstacle.

Return the number of possible unique paths that the robot can take to reach the bottom-right corner.

The testcases are generated so that the answer will be less than or equal to 2×10^9 .

Example 1:

$\text{arr}(x+1, y) = \text{arr}(x, y+1) = 0$

70. Climbing Stairs

Easy Topics Companies

You are climbing a staircase. It takes n steps to reach the top.

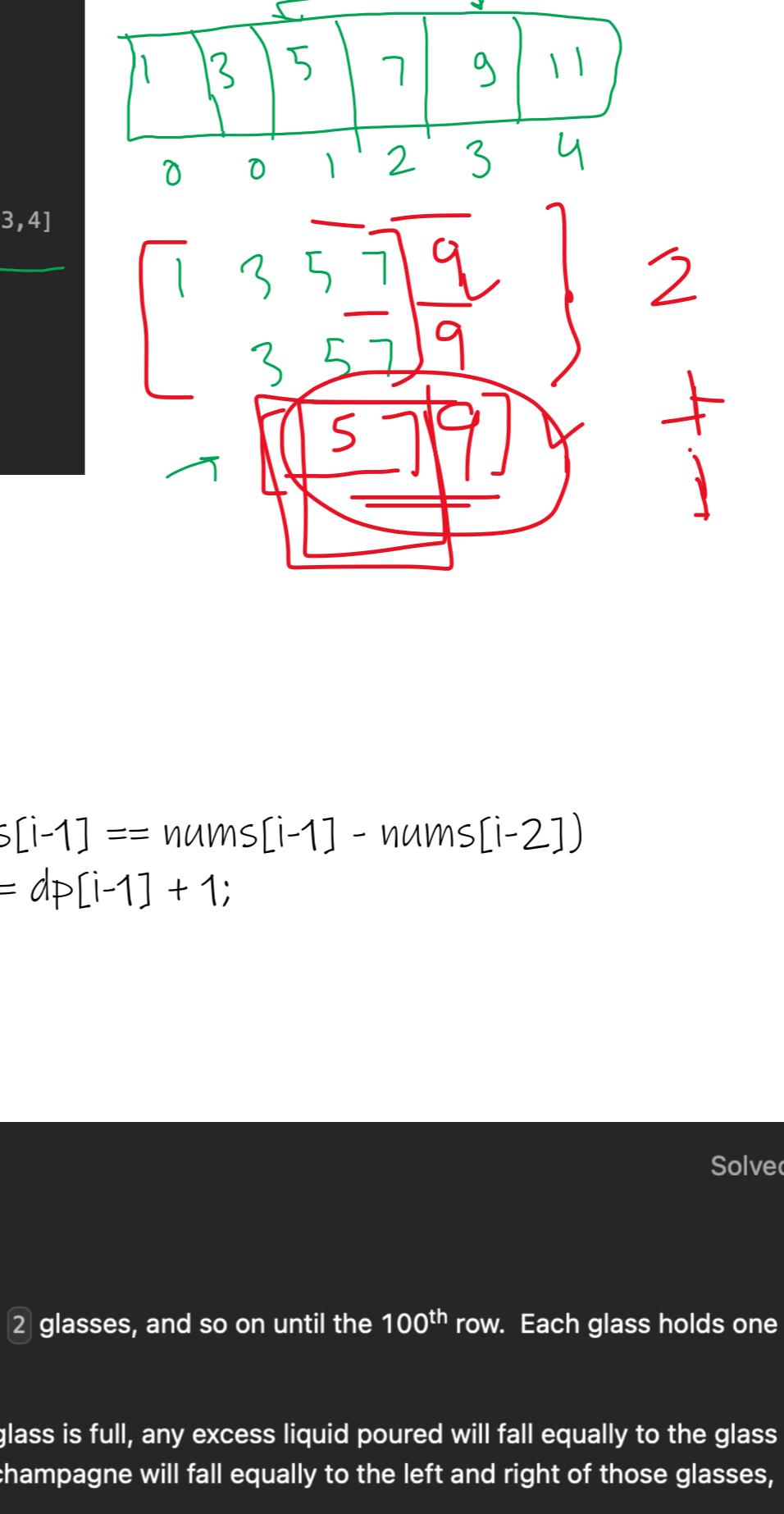
Each time you can either climb 1 or 2 steps. In how many distinct ways can you climb to the top?

Example 1:

Input: $n = 2$
Output: 2
Explanation: There are two ways to climb to the top.
1. 1 step + 1 step
2. 2 steps

Example 2:

Input: $n = 3$
Output: 3
Explanation: There are three ways to climb to the top.
1. 1 step + 1 step + 1 step
2. 1 step + 2 steps
3. 2 steps + 1 step



413. Arithmetic Slices

Medium Topics Companies

An integer array is called arithmetic if it consists of at least three elements and if the difference between any two consecutive elements is the same.

For example, $[1, 3, 5, 7, 9]$, $[7, 7, 7, 7]$, and $[3, -1, -5, -9]$ are arithmetic sequences.

Given an integer array nums , return the number of arithmetic subarrays of nums .

A subarray is a contiguous subsequence of the array.

Example 1:

Input: $\text{nums} = [1, 2, 3, 4]$
Output: 3
Explanation: We have 3 arithmetic slices in nums : $[1, 2, 3]$, $[2, 3, 4]$ and $[1, 2, 3, 4]$ itself.

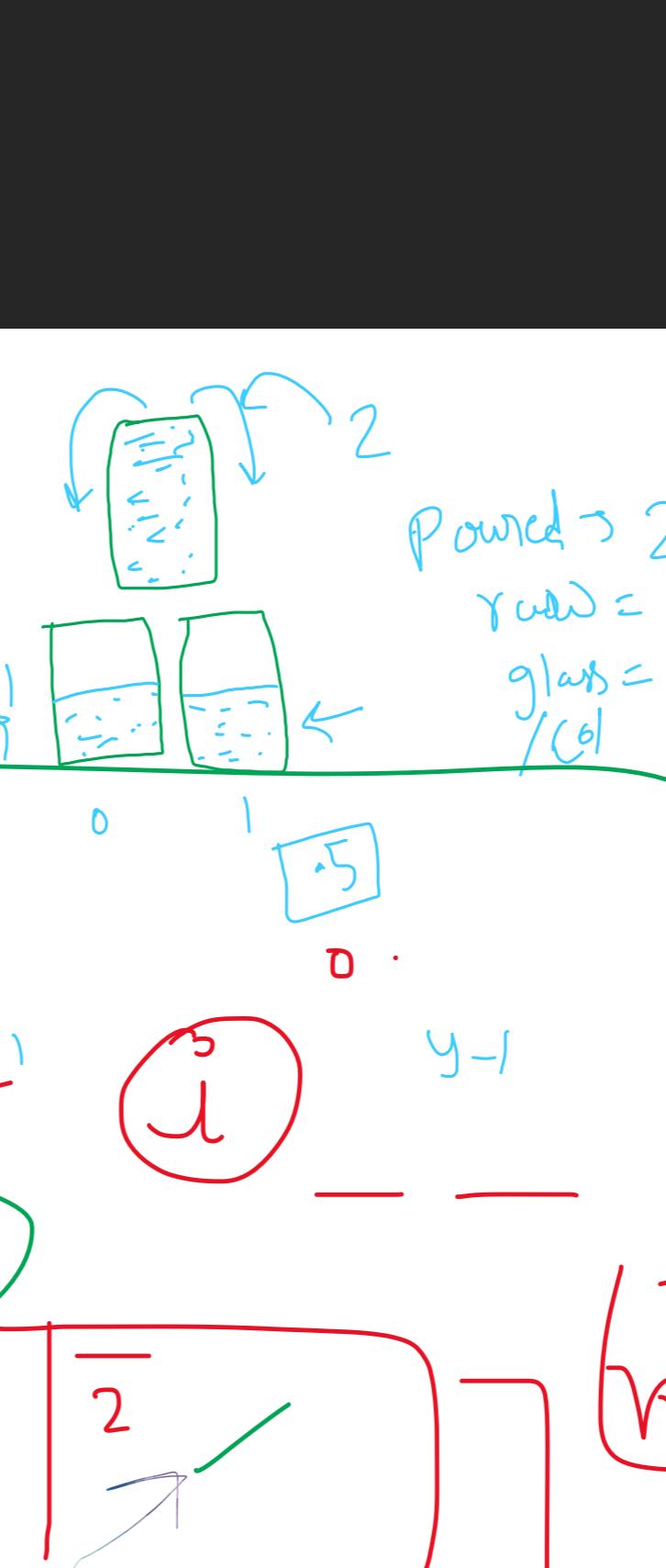
Example 2:

Input: $\text{nums} = [1]$
Output: 0

$\text{if}(\text{nums}[i] - \text{nums}[i-1] == \text{nums}[i-1] - \text{nums}[i-2])$

$\text{dp}[i] = \text{dp}[i-1] + 1;$

$\text{ans} += \text{dp}[i];$



799. Champagne Tower

Medium Topics Companies

We stack glasses in a pyramid, where the first row has 1 glass, the second row has 2 glasses, and so on until the 100th row. Each glass holds one cup of champagne.

Then, some champagne is poured into the first glass at the top. When the topmost glass is full, any excess liquid poured will fall equally to the glass immediately to the left and right of it. When those glasses become full, any excess champagne will fall equally to the left and right of those glasses, and so on. (A glass at the bottom row has its excess champagne fall on the floor.)

For example, after one cup of champagne is poured, the top most glass is full. After two cups of champagne are poured, the two glasses on the second row are half full. After three cups of champagne are poured, those two cups become full - there are 3 full glasses total now. After four cups of champagne are poured, the third row has the middle glass half full, and the two outside glasses are a quarter full, as pictured below.

Example 1:

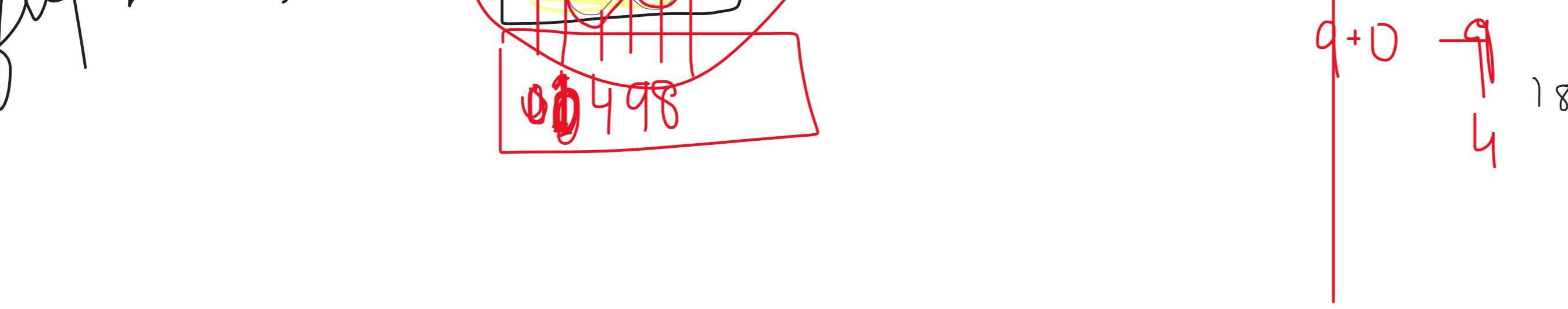
Input: $\text{poured} = 1$, $\text{query_row} = 1$, $\text{query_glass} = 1$
Output: 0.00000
Explanation: We poured 1 cup of champagne to the top glass of the tower (which is indexed as $(0, 0)$). There will be no excess liquid so all the glasses under the top glass will remain empty.

Example 2:

Input: $\text{poured} = 2$, $\text{query_row} = 1$, $\text{query_glass} = 1$
Output: 0.00000
Explanation: We poured 2 cups of champagne to the top glass of the tower (which is indexed as $(0, 0)$). There is one cup of excess liquid. The glass indexed as $(1, 0)$ and the glass indexed as $(1, 1)$ will share the excess liquid equally, and each will get half cup of champagne.

Constraints:

- $0 \leq \text{poured} \leq 10^9$
- $0 \leq \text{query_glass} \leq \text{query_row} < 100$



740. Delete and Earn

Medium Topics Companies

You are given an integer array nums . You want to maximize the number of points you get by performing the following operation any number of times:

Pick any $\text{nums}[i]$ and delete it to earn $\text{nums}[i]$ points. Afterwards, you must delete every element equal to $\text{nums}[i] - 1$ and every element equal to $\text{nums}[i] + 1$.

Return the maximum number of points you can earn by applying the above operation some number of times.

Example 1:

Input: $\text{nums} = [3, 4, 2]$
Output: 6
Explanation: You can perform the following operations:
- Delete 4 to earn 4 points. Consequently, 3 is also deleted. $\text{nums} = [2]$.
- Delete 2 to earn 2 points. $\text{nums} = []$.
You earn a total of 6 points.

Example 2:

Input: $\text{nums} = [2, 2, 3, 3, 3, 4]$
Output: 9
Explanation: You can perform the following operations:
- Delete 3 to earn 3 points. All 2's and 4's are also deleted. $\text{nums} = [3, 3]$.
- Delete a 3 again to earn 3 points. $\text{nums} = []$.
- Delete a 3 once more to earn 3 points. $\text{nums} = []$.
You earn a total of 9 points.

