Number of Ways to Stay in the Same Place After Some Steps

You have a pointer at index 0 in an array of size arrLen. At each step, you can move 1 position to the left, 1 position to the right in the array, or stay in the same place (The pointer should not be placed outside the array at any time).

Given two integers steps and arrLen, return the number of ways such that your pointer is still at index 0 after **exactly** steps steps. Since the answer may be too large, return it **modulo** $10^9 + 7$.

Example 1:

Input: steps = 3, arrLen = 2

Output: 4

Explanation: There are 4 differents ways to stay at index 0 after 3 steps.

Right, Left, Stay

Stay, Right, Left

Right, Stay, Left

Stay, Stay, Stay

Example 2:

Input: steps = 2, arrLen = 4

Output: 2

Explanation: There are 2 differents ways to stay at index 0 after 2 steps

Right, Left

Stay, Stay

Example 3:

Input: steps = 4, arrLen = 2

Output: 8

Constraints:

- 1 <= steps <= 500
- 1 <= arrLen <= 10^6