**Problem #2210: Count Hills and Valleys in an Array (Easy)**

<https://leetcode.com/problems/count-hills-and-valleys-in-an-array/description/>

**My Solution:**

1. Intiialize count to 0.
2. Iterate with I in range from 1 through (length of nums – 1).

If nums at index I is equal to nums at index (I + 1), then nums at index I is equal to nums at index I – 1.

If nums at index I is greater than its neighbors at index (I – 1) and (I + 1) , or if nums at index I is less than its neighbors at index (I – 1) and (I + 1), then increment count by 1 since we have either a hill or a valley.

1. Return count.

class Solution:

def countHillValley(self, nums: List[int]) -> int:

count = 0

for i in range(1, len(nums) - 1):

if nums[i] == nums[i + 1]:

nums[i] = nums[i - 1]

if (nums[i] > nums[i - 1] and nums[i] > nums[i + 1]) or \

(nums[i] < nums[i - 1] and nums[i] < nums[i + 1]):

count += 1

return count