**376. Wiggle Subsequence**

Medium

135867Add to ListShare

Given an integer array nums, return *the length of the longest****wiggle sequence***.

A **wiggle sequence** is a sequence where the differences between successive numbers strictly alternate between positive and negative. The first difference (if one exists) may be either positive or negative. A sequence with fewer than two elements is trivially a wiggle sequence.

* For example, [1, 7, 4, 9, 2, 5] is a **wiggle sequence** because the differences (6, -3, 5, -7, 3) are alternately positive and negative.
* In contrast, [1, 4, 7, 2, 5] and [1, 7, 4, 5, 5] are not wiggle sequences, the first because its first two differences are positive and the second because its last difference is zero.

A **subsequence** is obtained by deleting some elements (eventually, also zero) from the original sequence, leaving the remaining elements in their original order.

**Example 1:**

**Input:** nums = [1,7,4,9,2,5]

**Output:** 6

**Explanation:** The entire sequence is a wiggle sequence.

**Example 2:**

**Input:** nums = [1,17,5,10,13,15,10,5,16,8]

**Output:** 7

**Explanation:** There are several subsequences that achieve this length. One is [1,17,10,13,10,16,8].

**Example 3:**

**Input:** nums = [1,2,3,4,5,6,7,8,9]

**Output:** 2

**Constraints:**

* 1 <= nums.length <= 1000
* 0 <= nums[i] <= 1000

**Follow up:** Could you solve this in O(n) time?

Accepted

83,351

Submissions

206,395