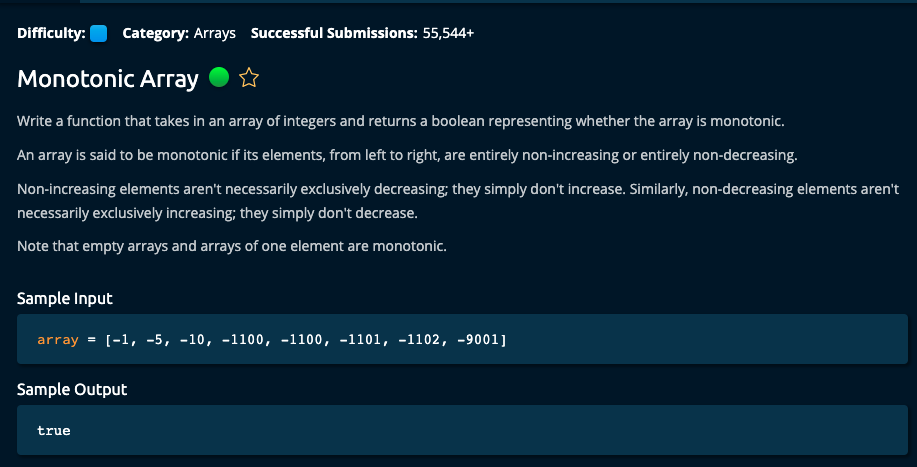
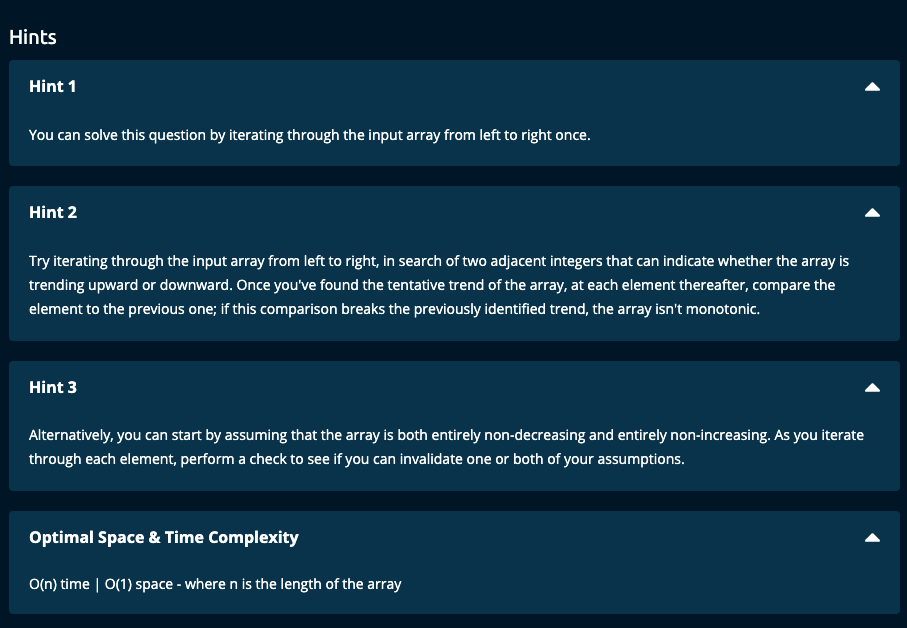
Monotonic Array (Medium). https://www.algoexpert.io/questions/monotonic-array



My Solution:

def isMonotonic(array):

n = len(array)

if n <= 1:

return True

if array[0] < array[-1]: # monotonic non decreasing array

for i in range(n-1):

if array[i] > array[i+1]:

return False

else: # monotonic non increasing array

for i in range(n-1):

if array[i] < array[i+1]:

return False

return True

JJ Notes:

1. Use the first and last elements in the array to determine the direction of monotonicity.
2. Iterate through the array comparing each element to the next element to see if the direction does not change. If there is a change, return False.
3. Finally return True if iterating through the array does not show any change in the direction.

Time Complexity : O(n), Space Complexity: O(1)

Algoexpert Solution:

def isMonotonic(array):

isNonDecreasing = True

isNonIncreasing = True

for i in range(1, len(array)):

if array[i] < array[i - 1]:

isNonDecreasing = False

if array[i] > array[i - 1]:

isNonIncreasing = False

return isNonDecreasing or isNonIncreasing

1. Here we are setting isNonDecreasing and isNonIncreasing to True initially.
2. Iterate through the array comparing each element with its previous element. If there is a violation set the corresponding Boolean to False.
3. We return isNonDecreasing or isNonIncreasing – if one of them is False and the other True, then we will return True. However, if we have a change in direction, both of them will be False and so we will return False (since False or False = False).