

[3,6,9,11,4,5]

[1,6,9,1,4,5,9]

Нам нужно найти длину самой длинной **непрерывной** возрастающей подпоследовательности в массиве. Подпоследовательность считается возрастающей, если каждый следующий элемент больше предыдущего

Решение для [1, 6, 9, 1, 4, 5, 9]

max\_length = 1

current\_length = 1

1. **Проход по массиву:**

**Шаг 1** (i = 1, nums[i] = 6):

Сравниваем 6 и 1 — возрастающая последовательность продолжается.

Увеличиваем current\_length до 2.

**Шаг 2** (i = 2, nums[i] = 9):

Сравниваем 9 и 6 — последовательность продолжается.

Увеличиваем current\_length до 3.

**Шаг 3** (i = 3, nums[i] = 1):

Сравниваем 1 и 9 — последовательность прерывается.

Обновляем max\_length до 3 (так как current\_length = 3).

Сбрасываем current\_length в 1.

**Шаг 4** (i = 4, nums[i] = 4):

Сравниваем 4 и 1 — последовательность начинается заново.

Увеличиваем current\_length до 2.

**Шаг 5** (i = 5, nums[i] = 5):

Сравниваем 5 и 4 — последовательность продолжается.

Увеличиваем current\_length до 3.

**Шаг 6** (i = 6, nums[i] = 9):

Сравниваем 9 и 5 — последовательность продолжается.

Увеличиваем current\_length до 4.

class Solution:

def findLengthOfLCIS(self, nums):

max\_length = 1

current\_length = 1

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

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

current\_length += 1

else:

current\_length = 1

max\_length = max(max\_length, current\_length)

return max\_length