Given an unsorted array of integers nums, return *the length of the longest* ***continuous increasing subsequence*** *(i.e. subarray)*. The subsequence must be **strictly** increasing.

1-step: Объявление переменных:

maxLen = 1: Переменная для хранения максимальной длины непрерывной возрастающей подпоследовательности.

count = 1: Переменная для хранения текущей длины текущей возрастающей подпоследовательности.

2-step: Цикл for:

Мы проходим по массиву с индексом i, начиная с 0 и заканчивая на len(nums) - 2. Это связано с тем, что мы сравниваем текущий элемент с **следующим** (nums[i] < nums[i + 1]).

3-step: **Условие if nums[i] < nums[i + 1]:**

Если текущий элемент меньше следующего, это значит, что подпоследовательность продолжает увеличиваться.

В этом случае мы увеличиваем счетчик count на 1.

4-step:**Сброс длины подпоследовательности:**

Если последовательность перестала быть возрастающей (т.е. nums[i] >= nums[i + 1]), мы сбрасываем count обратно на 1.

5-step **Обновление максимальной длины:**

После каждого шага цикла мы проверяем, больше ли текущая длина count, чем уже найденная максимальная длина maxLen. Если больше, обновляем maxLen.

Result:
По завершении цикла возвращаем maxLen

Result:
код эффективно решает задачу за O(n) времени, проходя массив только один раз.