Longest Consecutive Sequence

Difficulty	Medium
Category	Arrays
Question	https://leetcode.com/problems/longest-consecutive-sequence/
Solution	https://youtu.be/P6RZZMu_maU
Status	Done

Question

> Given an unsorted array of integers `nums`, return *the length of the longest consecutive elements sequence.*
>
> You must write an algorithm that runs in `O(n)` time.

Example

Example 1:

```
Input: nums = [100,4,200,1,3,2]
Output: 4
Explanation: The longest consecutive elements sequence is[1, 2, 3, 4]. Therefore its lengt
h is 4.
```

Example 2:

```
Input: nums = [0,3,7,2,5,8,4,6,0,1]
Output: 9
```

Idea

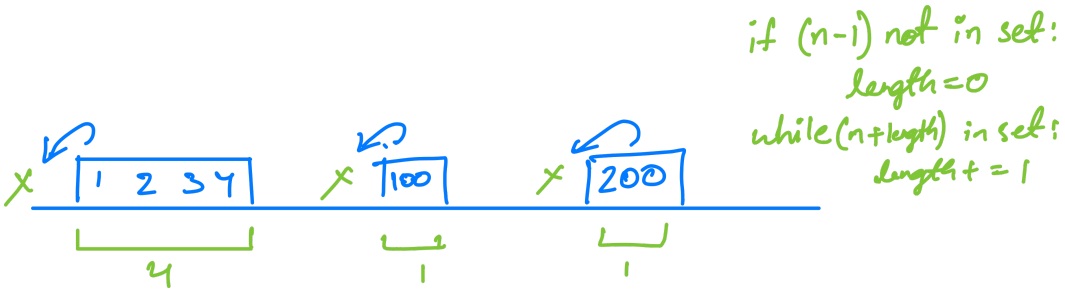
💡 Check from a random number to see if there's value -1 smaller than it. If not, it means that this random number is the start of a sequence. Then use for loop to continuously check for +1 value starting from that number, record the length and update the maxLength

Solution

```
class Solution:
    def longestConsecutive(self, nums: List[int]) -> int:
        # Create a set to efficiently check for the presence of numbers
        numSet = set(nums)
        longest_seq = 0

        # Iterate through each number in the input list
        for num in nums:
            # Check if the previous number (num - 1) is not in the set
            if (num - 1) not in numSet:
                length = 0
                # Count the consecutive numbers starting from the current number (num)
                while (num + length) in numSet:
                    length += 1
                # Update the longest sequence length
                longest_seq = max(length, longest_seq)

        # Return the length of the longest consecutive sequence
        return longest_seq
```



1. The `longestConsecutive` method takes a list of integers `nums` as input and returns an integer representing the length of the longest consecutive sequence.
2. A set named `numSet` is created from the input list `nums`. This set allows for efficient membership checks, making it easy to determine whether a number is present in

the input list.

3. The variable `longest_seq` is initialized to 0. This variable will keep track of the length of the longest consecutive sequence found.
4. The code then iterates through each number in the input list `nums` using a `for` loop.
5. Within the loop, it checks if the previous number (i.e., `num - 1`) is not present in the `numSet`. This check is used to identify the starting point of a potential consecutive sequence.
6. If `(num - 1)` is not in `numSet`, it means that the current number `num` can be the beginning of a consecutive sequence.
7. Inside the loop, a variable `length` is initialized to 0. This variable is used to count the length of the consecutive sequence starting from the current number `num`.
8. A `while` loop is used to increment `length` as long as the next consecutive number `(num + length)` is present in the `numSet`.
9. The code then updates `longest_seq` by taking the maximum of the current `length` and the previous `longest_seq`. This ensures that `longest_seq` always stores the length of the longest consecutive sequence encountered so far.
10. After processing all numbers in the input list, the method returns `longest_seq`, which represents the length of the longest consecutive sequence found in the input list.