Longest Consecutive Subsequence

128. Longest Consecutive Sequence

Medium 18.2K 817

Companies

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 1:

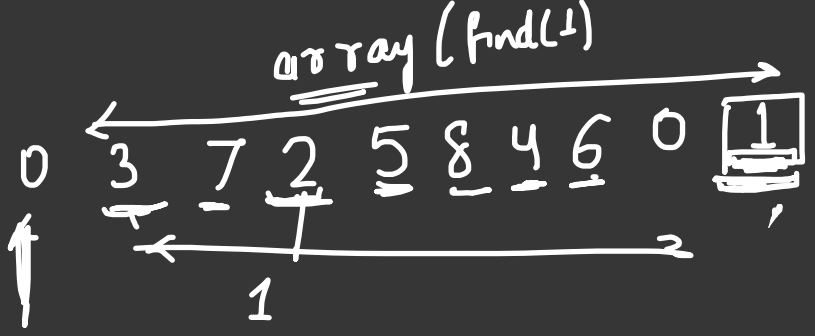
> **Input:** nums = [100,4,200,1,3,2]
> **Output:** 4
> **Explanation:** The longest consecutive elements sequence is `[1, 2, 3, 4]`. Therefore its length is 4.

Example 2:

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

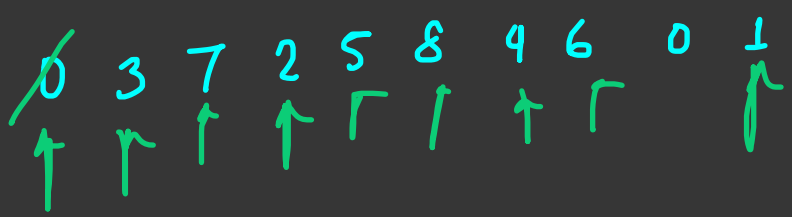
eg: arr = {100, 4, 200, 1, 3, 2}

1 2 3 4 } Consecutive
longest sequence



Brute force:

- Find the next Consecutive element in array and count it.



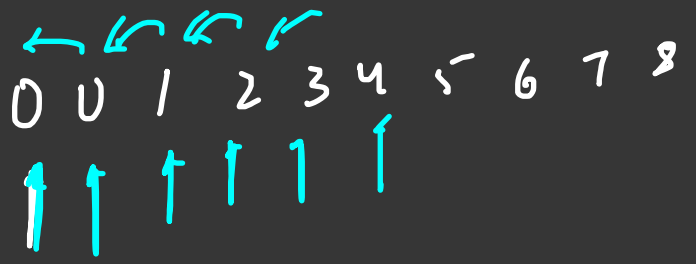
0+1 = 1
1+1 = 2
2+1 = 3
3+1 = 4

4+1 = 5
5+1 = 6
6+1 = 7
7+1 = 8

Better Approach

We have asked to the Length of longest Consecutive sequence.

Sort the array



check prev if

arr[i] = arr[i-1]

count++;

else

count=1;

0 0 1 2 3 4 5 6 7 8

if(arr[i]

1 2 3 4 100 200

arr[i] - 1 == arr[i-1]

2 - 1 : 1 == 1

count = 1 2 3 4

len = 4

Problem

100 - 1

arr : 4 count : 1

duplicates → 0 1 1 2 2

count = 1

3 → 2

0 1 1 2

count

last prev

1 → 0

arr[i] - 1 == last prev

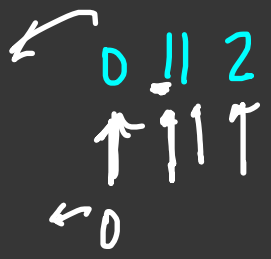
count ++,

last prev = i

last number = 0

longest = 1

count = ~~0~~ 2



- We need to skip duplicates

for (i = 0 ; i < n ; i++)

if (arr[i] - 1 == last) count++ ,

last = arr[i] ,

else if (arr[i] != last)

count = 1 ,

last = arr[i] ,

longest = max (longest , count)

Optimal Approach:

100 1 200 2 4 3

-1 2 -2 -4 -1 -3

0 1 2 0 1 2
3 4 4
5

search kar next consecutive element present hai kiske nahi

hashmap<int,int>

while(

100	1
1	1
200	1
2	1
4	1

3 1

using set

1 2 3 4 100 200

set → Store the distinct element in sorted fashion.

1 2 3 4 100 200

iterate over set

2-1 = 1

Set

→ Idea of using set the find the element next of x is present or not in O(1) time Complexity (this is assumption)

→ So we push everything set and iterate over array and try to find starting point rather than finding next of every element

Starting point = if (X-1 doesn't exists)

if (st.find(x-1) == st.end())

→ So, We start from starting point and find the length of consecutive sequence.

Pseudocode:

st.push(every element)

~~for (i=0; i<n; i++) push(arr~~

x = arr[i];

if(st.(x-1) == ~~for~~ st.~~find~~ end())

x = arr[i-1];

~~for~~