Problem | Editorial | Submissions

Longest Consecutive Subsequence

Difficulty: **Medium** Accuracy: **33.0%** Submissions: **342K+** Points: **4**

Given an array **arr[]** of non-negative integers. Find the **length** of the longest sub-sequence such that elements in the subsequence are consecutive integers, the **consecutive numbers** can be in **any order.**

Examples:

Input: arr[] = [2, 6, 1, 9, 4, 5, 3]
Output: 6
Explanation: The consecutive numbers here are 1, 2, 3, 4, 5, 6. These 6 numbers form the longest consecutive subsquence.

Input: arr[] = [1, 9, 3, 10, 4, 20, 2]
Output: 4
Explanation: 1, 2, 3, 4 is the longest consecutive subsequence.

Input: arr[] = [15, 13, 12, 14, 11, 10, 9]
Output: 7
Explanation: The longest consecutive subsequence is 9, 10, 11, 12, 13,

Java (1.8) | Average Time: 25m | Start Timer

```java
// } Driver Code Ends
class Solution {
    public int longestConsecutive(int[] arr) {
        if (arr == null || arr.length == 0) {
            return 0;
        }
        Set<Integer> set = new HashSet<>();
        for (int num : arr) {
            set.add(num);
        }
        int longestStreak = 0;
        for (int num : arr) {
            if (!set.contains(num - 1)) {
                int currentNum = num;
                int currentStreak = 1;
                while (set.contains(currentNum + 1)) {
                    currentNum += 1;
                    currentStreak += 1;
                }
                longestStreak = Math.max(longestStreak, currentStreak);
            }
        }
        return longestStreak;
    }
}
```

Custom Input | Compile & Run | Submit