

[Dashboard](#) / [My courses](#) / [CS23331-DAA-2023-CSE](#) / [Dynamic Programming](#) / [4-DP-Longest non-decreasing Subsequence](#)**Started on** Tuesday, 19 November 2024, 11:56 PM**State** Finished**Completed on** Tuesday, 19 November 2024, 11:57 PM**Time taken** 28 secs**Marks** 1.00/1.00**Grade** 10.00 out of 10.00 (100%)**Question 1**

Correct

Mark 1.00 out of 1.00

Problem statement:

Find the length of the Longest Non-decreasing Subsequence in a given Sequence.

Eg:

Input:9

Sequence:[-1,3,4,5,2,2,2,2,3]

the subsequence is [-1,2,2,2,2,3]

Output:6

Answer: (penalty regime: 0 %)

```
1 #include <stdio.h>
2 int main()
3 {
4     int n;
5     scanf("%d", &n);
6
7     int sequence[n];
8     for (int i = 0; i < n; i++) {
9         scanf("%d", &sequence[i]);
10    }
11
12    int dp[n];
13    for (int i = 0; i < n; i++) {
14        dp[i] = 1;
15    }
16    for (int i = 1; i < n; i++) {
17        for (int j = 0; j < i; j++) {
18            if (sequence[i] >= sequence[j]) {
19                dp[i] = (dp[i] > dp[j] + 1) ? dp[i] : (dp[j] + 1);
20            }
21        }
22    }
23    int maxLength = 0;
24    for (int i = 0; i < n; i++) {
25        if (dp[i] > maxLength) {
26            maxLength = dp[i];
27        }
28    }
29
30    printf("%d\n", maxLength);
```

```
31 |
32 |         return 0;
33 |     }
34 |
```

	Input	Expected	Got	
✓	9 -1 3 4 5 2 2 2 2 3	6	6	✓
✓	7 1 2 2 4 5 7 6	6	6	✓

Passed all tests! ✓

Correct

Marks for this submission: 1.00/1.00.

◀ 3-DP-Longest Common Subsequence

Jump to...

1-Finding Duplicates- $O(n^2)$ Time Complexity, $O(1)$ Space Complexity
▶