

CS23331-DAA-2024-CSE / 3-DP-Longest Common Subsequence



3-DP-Longest Common Subsequence

Started on	Sunday, 12 October 2025, 7:02 PM
State	Finished
Completed on	Sunday, 12 October 2025, 7:03 PM
Time taken	45 secs
Marks	1.00/1.00
Grade	10.00 out of 10.00 (100%)

Question 1 | Correct | Mark 1.00 out of 1.00 | [Flag question](#)

Given two strings find the length of the common longest subsequence(need not be contiguous) between the two.

Example:

s1: ggtabe

s2: tgatasb

s1	a	g	g	t	a	b	
s2	g	x	t	x	a	y	b

The length is 4

Solving it using Dynamic Programming

For example:

Input	Result
aab	2
azb	

Answer: (penalty regime: 0 %)

```
1 #include <stdio.h>
2 #include <string.h>
3
4 int max(int a, int b) {
5     return (a > b) ? a : b;
6 }
7
8 int main() {
9     char s1[100], s2[100];
10    scanf("%s", s1);
11    scanf("%s", s2);
12
13    int n = strlen(s1);
14    int m = strlen(s2);
15    int dp[n + 1][m + 1];
16
17    for (int i = 0; i <= n; i++) dp[i][0] = 0;
18    for (int j = 0; j <= m; j++) dp[0][j] = 0;
19
20    for (int i = 1; i <= n; i++) {
21        for (int j = 1; j <= m; j++) {
22            if (s1[i - 1] == s2[j - 1])
23                dp[i][j] = 1 + dp[i - 1][j - 1];
24            else
25                dp[i][j] = max(dp[i - 1][j], dp[i][j - 1]);
26        }
27    }
28
29    printf("%d\n", dp[n][m]);
30    return 0;
31 }
32
```

	Input	Expected	Got	
✓	aab	2	2	✓

	azb			
✓	ABCD	4	4	✓
	ABCD			

Passed all tests! ✓

Correct

Marks for this submission: 1.00/1.00.

[Finish review](#)

[Back to Course](#)

[Data retention summary](#)