3-DP-Longest Common Subsequence

Started on	Thursday, 20 November 2025, 9:37 PM
State	Finished
Completed on	Thursday, 20 November 2025, 9:39 PM
Time taken	2 mins 38 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

Solveing it using Dynamic Programming

For example:

Input	Result
aab azb	2

Answer: (penalty regime: 0 %)

```c
#include <stdio.h>
#include <string.h>

#define MAX 1000
int max(int a, int b) {
    return (a > b) ? a : b;
}

int main() {
    char s1[MAX], s2[MAX];
    scanf("%s", s1);
    scanf("%s", s2);

    int n = strlen(s1);
    int m = strlen(s2);

    int dp[n + 1][m + 1];

    for (int i = 0; i <= n; i++) {
        for (int j = 0; j <= m; j++) {
            if (i == 0 || j == 0)
                dp[i][j] = 0;
            else if (s1[i - 1] == s2[j - 1])
                dp[i][j] = dp[i - 1][j - 1] + 1;
            else
                dp[i][j] = max(dp[i - 1][j], dp[i][j - 1]);
        }
    }

    printf("%d\n", dp[n][m]);
    return 0;
}
```

	Input	Expected	Got	
✔	aab	2	2	✔

	azb			
✔	ABCD ABCD	4	4	✔

Passed all tests! ✔

Correct

Marks for this submission: 1.00/1.00.

Finish review

Back to Course