Dashboard / My courses / CS23331-DAA-2023-CSE / Dynamic Programming / 3-DP-Longest Common Subsequence

Started on	Tuesday, 19 November 2024, 11:56 PM
State	Finished
Completed on	Tuesday, 19 November 2024, 11:56 PM
Time taken	35 secs
Marks	1.00/1.00
Grade	**10.00** out of 10.00 (**100**%)

Question **1**

Correct

Mark 1.00 out of 1.00

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 %)

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

#define MAX_LEN 100

int main() {
    char s1[MAX_LEN], s2[MAX_LEN];
    int dp[MAX_LEN + 1][MAX_LEN + 1];
    scanf("%s", s1);
    scanf("%s", s2);
    int m = 0, n = 0;
    while (s1[m] != '\0') m++;
    while (s2[n] != '\0') n++;
    while (s2[n] != '\0') n++;

    // Initialize the DP table
    for (int i = 0; i <= m; i++) {
        for (int j = 0; j <= n; 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] = (dp[i - 1][j] > dp[i][j - 1]) ? dp[i - 1][j] : dp[
            }
        }
    }
    printf("%d\n", dp[m][n]);

    return 0;
}

```

	Input	Expected	Got	
✓	aab azb	2	2	✓
✓	ABCD ABCD	4	4	✓

Passed all tests! ✓

Correct

Marks for this submission: 1.00/1.00.

◀ 2-DP-Playing with chessboard

Jump to...

4-DP-Longest non-decreasing Subsequence ▶