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

Started on	Friday, 25 October 2024, 2:00 PM
State	Finished
Completed on	Friday, 25 October 2024, 2:05 PM
Time taken	4 mins 55 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
```

g x t x a y b

The length is 4

Solveing it using Dynamic Programming

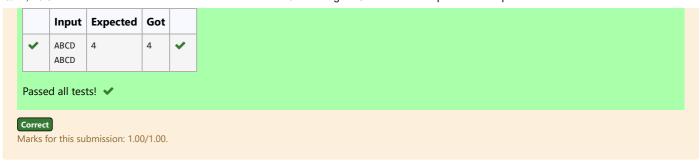
For example:

Input	Result
aab	2
azb	

Answer: (penalty regime: 0 %)

```
#include <stdio.h>
#include <string.h>
    #define MAX 1000
 3
 4 v int Common(char *s1, char *s2) {
          int m = strlen(s1);
int n = strlen(s2);
 5
 6
          int dp[MAX][MAX];
 7
 8
          for (int i = 0; i <= m; i++) {</pre>
               for (int j = 0; j <= n; j++) {
   if (i == 0 || j == 0) {
 9
10
                         dp[i][j] = 0;
11
12
                    else if (s1[i - 1] == s2[j - 1]) {
    dp[i][j] = dp[i - 1][j - 1] + 1;
13
14
15
                    }
16
                    else {
17
                         dp[i][j] = (dp[i - 1][j] > dp[i][j - 1]) ? dp[i - 1][j] : dp[i][j - 1];
18
19
20
21
          return dp[m][n];
22
23
   int main() {
          char s1[MAX], s2[MAX];
24
          scanf("%s", s1);
scanf("%s", s2);
25
26
27
          int lcsLength = Common(s1, s2);
          printf("%d\n", lcsLength);
28
29
          return 0;
30
```

		Input	Expected	Got	
•	~	aab azb	2	2	~



■ 2-DP-Playing with chessboard

4-DP-Longest non-decreasing Subsequence ►