b

```
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.

а

Example:

- s1: ggtabe
- s2: tgatasb

s1	a	g	g	t	а	b

t

The length is 4

s2

Solveing it using Dynamic Programming

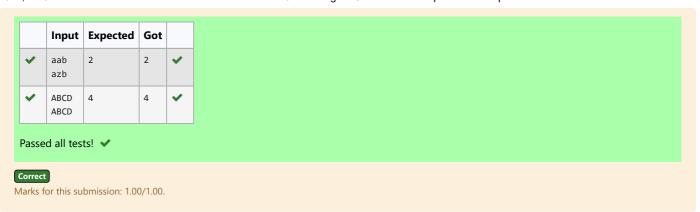
g

## For example:

Input	Result
aab	2
azb	

Answer: (penalty regime: 0 %)

```
#include <stdio.h>
1
 2
    #include <string.h>
3
4
    int longestCommonSubsequence(char* X, char* Y) {
         int m = strlen(X);
int n = strlen(Y);
5
6
7
         int dp[m + 1][n + 1];
8
9
         for (int i = 0; i <= m; i++) {
             for (int j = 0; j <= n; j++) {
  if (i == 0 || j == 0) {
10
11
12
                       dp[i][j] = 0;
13
                  }
                  else if (X[i - 1] == Y[j - 1]) {
14
                       dp[i][j] = dp[i - 1][j - 1] + 1;
15
16
17
                  else {
                       dp[i][j] = (dp[i - 1][j] > dp[i][j - 1]) ? dp[i - 1][j] : dp[i][j - 1];
18
19
20
21
22
23
         return dp[m][n];
24
    }
25
    int main() {
26
27
         char X[50];
         char Y[50];
28
         scanf("%s %s",X,Y);
int length = longestCommonSubsequence(X,Y);
29
30
         printf("%d", length);
31
32
         return 0;
33
34
35 }
```



## ■ 2-DP-Playing with chessboard

Jump to... \$

4-DP-Longest non-decreasing Subsequence ►