Wed-LCS

ZPRAC-16-17-Lab9

Given two strings, compute the length of longest common sub-sequence of those two strings. A sub-sequence differs from sub-string: unlike sub-strings, sub-sequences are **not** required to occupy consecutive positions within the original sequences.
Example: Longest common sub-sequence of
A**BC**DE**F**UV**Z** and
U**B**X**C**WX**F**WW**Z**L
is BCFZ (marked in bold)

Recursive Formulation: Consider two strings str1 and str2 of size m and n respectively.

lcslen(m,n)	= lcslen(m-1,n-1) + 1	**if** str1[m-1] == str2[n-1]
	= max(lcslen(m-1,n), lcslen(m,n-1))	**otherwise**

Now add boundary cases!
(Can you guess, Why this formulation works? Can you come up with another formulation?)

Input:
2 strings separated by white-space. The strings will be at most 15 characters long.

Output:
The length of the a longest common sub-sequence.