AlgoExpert Quad Layout Python 12px Sublime Monokai 00:00:00

Prompt Scratchpad Our Solution(s) Video Explanation Run Code

Solution 1 Solution 2 Solution 3 Solution 4

```python
# Copyright © 2020 AlgoExpert, LLC. All rights reserved.

# O(nm) time | O(nm) space
def longestCommonSubsequence(str1, str2):
    lengths = [[0 for x in range(len(str1) + 1)] for y in range(len(str2) + 1)]
    for i in range(1, len(str2) + 1):
        for j in range(1, len(str1) + 1):
            if str2[i - 1] == str1[j - 1]:
                lengths[i][j] = lengths[i - 1][j - 1] + 1
            else:
                lengths[i][j] = max(lengths[i - 1][j], lengths[i][j - 1])
    return buildSequence(lengths, str1)


def buildSequence(lengths, string):
    sequence = []
    i = len(lengths) - 1
    j = len(lengths[0]) - 1
    while i != 0 and j != 0:
        if lengths[i][j] == lengths[i - 1][j]:
            i -= 1
        elif lengths[i][j] == lengths[i][j - 1]:
            j -= 1
        else:
            sequence.append(string[j - 1])
            i -= 1
            j -= 1
    return list(reversed(sequence))
```