

Our Solution(s)

Run Code

Your Solutions

Run Code

Solution 1Solution 2Solution 3Solution 4

```
1 // Copyright © 2020 AlgoExpert, LLC. All rights reserved.
2
3 package main
4
5 // O(nm*min(n, m)) time | O(nm*min(n, m)) space
6 func LongestCommonSubsequence(s1 string, s2 string) string {
7     lcs := make([][]string, len(s2)+1)
8     for i := range lcs {
9         lcs[i] = make([]string, len(s1)+1)
10    }
11    for i := 1; i < len(lcs); i++ {
12        for j := 1; j < len(lcs[i]); j++ {
13            if s2[i-1] == s1[j-1] {
14                lcs[i][j] = lcs[i-1][j-1] + string(s2[i-1])
15            } else {
16                if len(lcs[i-1][j]) < len(lcs[i][j-1]) {
17                    lcs[i][j] = lcs[i][j-1]
18                } else {
19                    lcs[i][j] = lcs[i-1][j]
20                }
21            }
22        }
23    }
24    return lcs[len(s2)][len(s1)]
25 }
26
```

Solution 1Solution 2Solution 3

```
1 package main
2
3 // O(nm*min(n, m)) time | O(nm*min(n, m)) space
4 func LongestCommonSubsequence(s1 string, s2 string) string {
5     // Write your code here.
6     return ""
7 }
8
```

Our Tests

Custom Output

Submit Code

