

PromptScratchpadOur Solution(s)Video Explanation

Run Code

Solution 1Solution 2Solution 3Solution 4

```
1 // Copyright © 2020 AlgoExpert, LLC. All rights reserved.
2
3 import java.util.*;
4
5 class Program {
6     // O(nm) time | O(nm) space
7     public static List<Character> longestCommonSubsequence(String str1, String str2) {
8         int[][][] lcs = new int[str2.length() + 1][str1.length() + 1][];
9         for (int i = 0; i < str2.length() + 1; i++) {
10             for (int j = 0; j < str1.length() + 1; j++) {
11                 lcs[i][j] = new int[] {0, 0, 0, 0};
12             }
13         }
14         for (int i = 1; i < str2.length() + 1; i++) {
15             for (int j = 1; j < str1.length() + 1; j++) {
16                 if (str2.charAt(i - 1) == str1.charAt(j - 1)) {
17                     int[] newEntry = {(int) str2.charAt(i - 1), lcs[i - 1][j - 1][1] + 1, i - 1, j - 1};
18                     lcs[i][j] = newEntry;
19                 } else {
20                     if (lcs[i - 1][j][1] > lcs[i][j - 1][1]) {
21                         int[] newEntry = {-1, lcs[i - 1][j][1], i - 1, j};
22                         lcs[i][j] = newEntry;
23                     } else {
24                         int[] newEntry = {-1, lcs[i][j - 1][1], i, j - 1};
25                         lcs[i][j] = newEntry;
26                     }
27                 }
28             }
29         }
30         return buildSequence(lcs);
31     }
32
33     public static List<Character> buildSequence(int[][][] lcs) {
34         List<Character> sequence = new ArrayList<Character>();
35         int i = lcs.length - 1;
36         int j = lcs[0].length - 1;
37         while (i != 0 && j != 0) {
38             int[] currentEntry = lcs[i][j];
39             if (currentEntry[0] != -1) {
40                 sequence.add(0, (char) currentEntry[0]);
41             }
42             i = currentEntry[2];
43             j = currentEntry[3];
44         }
45         return sequence;
46     }
47 }
48
```