

EC 4060 – Data Structures and Algorithms

LAB 09

K.J.M.U.G.S. Eranda Jayasinghe

2021/E/075

GROUP CG8

EC4070

29.11.2023

Q1.

```
import java.util.ArrayList;

import java.util.List;

import java.util.Scanner;

public class LCS{

    public static void main(String[] args) {

Scanner scanner = new Scanner(System.in);

String a;


        System.out.print("Enter your first word = ");

String A = scanner.nextLine().trim();

        System.out.print("Enter your second word = ");

String B = scanner.nextLine().trim();


        List<String> lcsResults = findLCS(A, B);

        for (String result : lcsResults) {

            System.out.println(result);

                break;

        }

    }

private static List<String> findLCS(String str1, String str2) {

    int m = str1.length();

    int n = str2.length();

    int[][] dp = new int[m + 1][n + 1];

    for (int i = 1; i <= m; i++) {

        for (int j = 1; j <= n; j++) {

            if (str1.charAt(i - 1) == str2.charAt(j - 1)) {

                dp[i][j] = dp[i - 1][j - 1] + 1;

            } else {
```

```

        dp[i][j] = Math.max(dp[i - 1][j], dp[i][j - 1]);
    }
}

List<String> lcsResults = new ArrayList<>();
backtrack(dp, str1, str2, m, n, "", lcsResults);

return lcsResults;
}

private static void backtrack(int[][] dp, String str1, String str2, int i, int j, String current, List<String> results) {
    if (i == 0 || j == 0) {
        results.add(new StringBuilder(current).reverse().toString());
        return;
    }

    if (str1.charAt(i - 1) == str2.charAt(j - 1)) {
        backtrack(dp, str1, str2, i - 1, j - 1, current + str1.charAt(i - 1), results);
    } else {
        if (dp[i - 1][j] >= dp[i][j - 1]) {
            backtrack(dp, str1, str2, i - 1, j, current, results);
        }
        if (dp[i][j - 1] >= dp[i - 1][j]) {
            backtrack(dp, str1, str2, i, j - 1, current, results);
        }
    }
}
}

```

```

1  import java.util.ArrayList;
2  import java.util.List;
3  import java.util.Scanner;
4
5  public class LCS{
6      public static void main(String[] args) {
7          Scanner scanner = new Scanner(System.in);
8          String a;
9
10         System.out.print("Enter your first word = ");
11         String A = scanner.nextLine().trim();
12         System.out.print("Enter your second word = ");
13         String B = scanner.nextLine().trim();
14
15         List<String> lcsResults = findLCS(A, B);
16         for (String result : lcsResults) {
17             System.out.println(result);
18             break;
19         }
20     }
21
22     private static List<String> findLCS(String str1, String str2) {
23         int m = str1.length();
24         int n = str2.length();
25
26         int[][] dp = new int[m + 1][n + 1];
27
28         for (int i = 1; i <= m; i++) {
29             for (int j = 1; j <= n; j++) {
30                 if (str1.charAt(i - 1) == str2.charAt(j - 1)) {
31                     dp[i][j] = dp[i - 1][j - 1] + 1;
32                 } else {
33                     dp[i][j] = Math.max(dp[i - 1][j], dp[i][j - 1]);
34                 }
35             }
36         }
37
38         List<String> lcsResults = new ArrayList<>();
39         backtrack(dp, str1, str2, m, n, "", lcsResults);
40
41         return lcsResults;
42     }
43
44     private static void backtrack(int[][] dp, String str1, String str2, int i, int j, String current, List<String> results) {
45         if (i == 0 || j == 0) {
46             results.add(new StringBuilder(current).reverse().toString());
47             return;
48         }
49
50         backtrack(dp, str1, str2, m, n, "", lcsResults);
51
52         return lcsResults;
53     }
54
55     private static void backtrack(int[][] dp, String str1, String str2, int i, int j, String current, List<String> results) {
56         if (i == 0 || j == 0) {
57             results.add(new StringBuilder(current).reverse().toString());
58             return;
59         }
60
61         if (str1.charAt(i - 1) == str2.charAt(j - 1)) {
62             backtrack(dp, str1, str2, i - 1, j - 1, current + str1.charAt(i - 1), results);
63         } else {
64             if (dp[i - 1][j] >= dp[i][j - 1]) {
65                 backtrack(dp, str1, str2, i - 1, j, current, results);
66             }
67             if (dp[i][j - 1] >= dp[i - 1][j]) {
68                 backtrack(dp, str1, str2, i, j - 1, current, results);
69             }
70         }
71     }
72 }

```

C:\WINDOWS\system32\cmd.exe

Microsoft Windows [Version 10.0.19045.3693]
(c) Microsoft Corporation. All rights reserved.

C:\Users\erand\OneDrive - University of Jaffna\lab9\New folder (2)>javac LCS.java

C:\Users\erand\OneDrive - University of Jaffna\lab9\New folder (2)>java LCS.java

Enter your first word = president
Enter your second word = providence
priden

C:\Users\erand\OneDrive - University of Jaffna\lab9\New folder (2)>java LCS.java

Enter your first word = creative
Enter your second word = colourful
cr

C:\Users\erand\OneDrive - University of Jaffna\lab9\New folder (2)>