

Task 2:

Longest Common Subsequence

Implement `int LCS(string text1, string text2)` to find the length of the longest common subsequence between two strings.

ANS:

```
package com.Day20;

public class LongestCommonSubsequence {
    public static int LCS(String text1, String text2) {
        int m = text1.length();
        int n = text2.length();
        // Create a 2D array to store lengths of LCS
        int[][] dp = new int[m + 1][n + 1];
        // Build the dp array bottom-up
        for (int i = 1; i <= m; i++) {
            for (int j = 1; j <= n; j++) {
                if (text1.charAt(i - 1) == text2.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]);
                }
            }
        }
        // Return the length of LCS
        return dp[m][n];
    }

    public static void main(String[] args) {
        String text1 = "abcde";
        String text2 = "ace";
        int length = LCS(text1, text2);
        System.out.println("Length of Longest Common Subsequence:
" + length);
    }
}
```

OUTPUT:

Length of Longest Common Subsequence: 3