```csharp
// Copyright © 2020 AlgoExpert, LLC. All rights reserved.

using System.Collections.Generic;

public class Program {
  // O(nm*min(n, m)) time | O(nm*min(n, m)) space
  public static List<char> LongestCommonSubsequence(string str1, string str2) {
    List<List<List<char> > > lcs = new List<List<List<char> > >();
    for (int i = 0; i < str2.Length + 1; i++) {
      lcs.Add(new List<List<char> >());
      for (int j = 0; j < str1.Length + 1; j++) {
        lcs[i].Add(new List<char>());
      }
    }
    for (int i = 1; i < str2.Length + 1; i++) {
      for (int j = 1; j < str1.Length + 1; j++) {
        if (str2[i - 1] == str1[j - 1]) {
          List<char> copy = new List<char>(lcs[i - 1][j - 1]);
          lcs[i][j] =  copy;
          lcs[i][j].Add(str2[i - 1]);
        } else {
          if (lcs[i - 1][j].Count > lcs[i][j - 1].Count) {
            lcs[i][j] =  lcs[i - 1][j];
          } else {
            lcs[i][j] =  lcs[i][j - 1];
          }
        }
      }
    }
    return lcs[str2.Length][str1.Length];
  }
}
```