Our Solution(s) Run Code

Solution 1 Solution 2

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

#include <vector>
using namespace std;

// O(nm) time | O(nm) space
int levenshteinDistance(string str1, string str2) {
  vector<vector<int>> edits(str2.length() + 1,
                            vector<int>(str1.length() + 1, 0));
  for (int i = 0; i < str2.length() + 1; i++) {
    for (int j = 0; j < str1.length() + 1; j++) {
      edits[i][j] = j;
    }
    edits[i][0] = i;
  }
  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]) {
        edits[i][j] = edits[i - 1][j - 1];
      } else {
        edits[i][j] =
            1 + min(edits[i - 1][j - 1], min(edits[i - 1][j], edits[i][j - 1]));
      }
    }
  }
  return edits[str2.length()][str1.length()];
}

```

Your Solutions Run Code

Solution 1 Solution 2 Solution 3

```cpp
using namespace std;

int levenshteinDistance(string str1, string str2) {
  // Write your code here.
  return -1;
}

```

Custom Output Raw Output Submit Code

Run or submit code when you're ready.