Assignment #3

Problem Solving and Programming in C++
Department of Computer Science
Old Dominion University

Objective: The main objective of this assignment is to assess the student's ability to provide a complete Black-Box testing plan for a computer program that is provided.

Description:

You need to design a complete Black-Box testing plan for the similarity program, which is provided (*posted on Blackboard*). Make sure you run the program multiple times to understand how the program works before writing the Black-Box test suite.

The provided program requests for **2 words** (*strings*). After given input, the program computes the similarity between the two entered strings based on the Levenshtein distance. The Levenshtein distance is a numerical metric for measuring the difference between two strings. The program provides a report after the user decides to end the game. This can be seen in the sample output below:

You must design a complete Black-Box testing plan for the program described. Your task is to apply what you have learned about black-box testing techniques to develop a full suite of test data for this program. Use the template file provided in the Instructions and supporting files on BlackBoard to organize your tests and test data. Fill in the information for each test that you develop for the 4 tables (*representative input, functional coverage, boundary values,* and *special values*). Please note that you

will need to add more rows as you might need to list **all possible** test cases for each type. Do not repeat tests which have already been used to test a particular description. Use as many rows as needed to capture all of your tests. Each of the four tables may span more than one page if necessary.

<u>Submission notes</u>:

- Submit a single file (*text*, *MS word*, or *pdf*). Name your file "Assg3_cslogin", where the cslogin is your login ID for the computers at the Department of Computer Science at ODU.

- Your file must include the following four sections with test data for each:
    1. Test data that covers representative inputs
    2. Test data that provides functional coverage
    3. Test data that provides for boundary-values testing
    4. Test data that implements special-values testing

- Submit your file in the respective Blackboard link.