UNIVERSITY OF THE WEST INDIES

Department of Computing
COMP2130 – Systems Programming (Semester II, 2015)
Lecturer: Mr. Kevin Miller

Assignment 1 - Problem Definition

Write a program that allows the user to spell check an ASCII text file. The program should read in a dictionary of words from the plain ASCII file linux.words (You will be given this file) and store it in a suitable array. The program should then read words, one at a time, from the user-given input file. Each word should be compared to the dictionary. If an exact match is found, the program should continue to the next word. If no exact match is found, then the program should begin searching for suitable suggestions for replacement.

The search for replacement words should work as follows. Let *N* be the length of the original input word, minus 1. Starting with that value of *N*, the program should search for any exact matches of the first *N* characters of the input word with any words in the dictionary, and add them to a suggestion list. Following that, the program should search for any exact matches of an *N* substring anywhere inside the input word, as compared to anywhere inside a dictionary word. Decrease the value of *N* by 1, and repeat. This process should stop when 10 words have been suggested, or when *N* reaches 0.

The program should provide the list of suggestions to the user through a simple text menu. One option in that menu should be to keep the original word. Either upon verifying that the original word is found in the dictionary, or upon the user selecting a replacement, the program should write the appropriate word to an output file. The program should discover the names of both the input file and the output file from command line arguments. The program should check to make sure that the appropriate number of command line arguments is given by the user and, if not, report an error along with the proper usage of the program.

The program does not have to be concerned with spacing or arrangement between words. Words can be output one per line, for example, even though that does not match how they were arranged in the input file. The program does not need to handle punctuation characters and can disregard them while checking spelling.

This assignment is due on **Friday, March 6th 2015 by 4pm.**
No late assignment will be accepted.

A submission container will be created for you to submit your assignment. Your code should be well commented and you should only submit one C source file. Use functions where appropriate.

DO NOT COPY CODE.