

CSCI 1300 - Starting Computing

Instructor: Fleming

Homework 10: Extra Credit Homework

Due Thursday, December 13th, by 6 pm

This assignment is due **Sunday, March 11th 6pm**.

- ***All components* (Cloud9 workspace, moodle quiz attempts) must be completed and submitted by Thursday, December 13th 6:00 pm for your homework to receive points.**

Objectives:

1. Trying to understand the problem and come up with a solution from the concepts learnt in class.
2. Using files and understanding parsing of files that might not have patterns in them.
3. Utilizing the concepts of vectors, classes and looping statement to solve the problem

Background

Spellchecker are a very useful program that has become a part of our daily lives, in phones, laptops or even in emails. In this homework, we are trying to create our own spell checker that has the ability to load from a database of misspelled words and its respective correct spelling, ability to add a new misspelled word and also correct any given sentence.

For this assignment you will be creating a **Spellchecker** program that will return words and phrases with the correct spelling if it is incorrect.

Two file are provided for you with the list of correctly spelt words, and possible misspelling for those words as two seperate files.

- Words of the vocabulary : The name of the file is vocabulary.txt
 - You can expect exactly one word on each line.
 - There are at most be 4000 such words
- Correct spelling of words and it possible misspellings : The name of the file is misspellings.csv
 - You can expect at most 2500 correctly spelt words (along with its possible misspelling) in each line
 - For each correctly spelt words, you can expect at most 20 possible misspellings
 - Every correctly spelt words, can have a variable number of misspelt words associated with it.

Structure of these files:

<ul style="list-style-type: none">• Words of the vocabulary: vocabulary.txt
<word1> <word2>

<ul style="list-style-type: none">• Correct spelling of words and it possible misspellings : misspelling.csv
<correctword1>,<misspelling1for_word1> <misspelling2for_word1> <correctword2>,<misspelling1for_word2> <correctword3>,<misspelling1for_word3> <misspelling2for_word3> <misspelling3for_word3>

***Note:** Each correct word can have more than one misspelling possible. You can use `split()` or `stringstreams` for the getting the words. Be careful of segmentation faults while reading these words.*

Problem:

Create a **spellchecker** that must be capable of reading in a sentence, and autocorrect every misspelt word in the given phrase utilizing the vocabulary and misspellings files provided to you. You can utilize any of the concepts we learnt in CSCI 1300, to solve the questions.

Mandatory Requirement: You must implement at least 3 functions apart from your main()

Suggestion for concepts that you can use for the project are:

1. Classes
2. 2d Arrays
3. Vectors

Test cases:

- 1) If the file does not exist, print the following and end the program:
`<filename> does not exist`
- 2) After reading in the vocabulary file, print
`Read <number_of_words> words from vocabulary.txt`
- 3) After reading in the misspelling file, print
`Read <number_of_words> lines from misspelling.csv`
- 4) Get the phrase input from the user by prompting the following in a single line: *Note: The phrase can have spaces, so advised to use getline*
`Enter the phrase you would like to correct:`
- 5) If the word in the phrase is not in the vocabulary and you have not come across it in both the files then replace it with the word **unknown**
- 6) If the word in the phrase is wrongly spelt then replace it with its correct spelling. *Note: The misspelling and its respective correct spelling are provided to you in misspelling.csv*
- 7) After correcting all the words in the phrase, print the corrected phrase.

Corner cases :

- 1) When reading the misspellings.csv you might come across some of the correctly spelled words that might not be in the vocabulary.txt. Add these correctly words to the dictionary that you are creating.
- 2) *Punctuations:*
 - a) Remove any leading punctuations or spaces before the first word. On the other hand, if the word has trailing punctuations, do not change.
Example :
`..... graEt Trip! Mississippi river !!!`
`great trip! mississippi river !!!`
 - b) If there are single or group of punctuations which are not the first word, ignore them.
 - c) If a word starts and ends with a punctuation and is also misspelt, then remove the starting punctuation, correct the misspelling and leave the ending punctuation untouched.
Example :
`!!!!Helo,,,`
`Hello,,,`
 - d) These are possible characters considered as punctuations in this Homework
`[.,/?:'!"@#$%^&*()_-=]`
- 3) Consider the words to be case insensitive i.e., Bridge and bridge refers to the same word.
- 4) Modify multiple spaces to a single space.
- 5) While printing the corrected sentence, make sure all the words are in lower case.

Note:

- 1) *You must have your own main() function.*

2) *You must not use global variables*

Sample run 1: vocabulary.txt not available
vocabulary.txt does not exist

Sample run 2: Simple output
Read 3006 words from vocabulary.txt Read 2908 lines from misspelling.csv Enter the phrase you would like to correct: I loev C++ i love c++

Sample run 3: Punctuation and corner case
Read 3006 words from vocabulary.txt Read 2908 lines from misspelling.csv Enter the phrase you would like to correct: Do you knwo youself? do you know yourself?

Sample run 4: Leading punctuation and spaces
Read 3006 words from vocabulary.txt Read 2908 lines from misspelling.csv Enter the phrase you would like to correct: ...,, \$%!HELLO WORLD! My firends are qwer. hello world! my friends are unknown.

Things to submit by due date	Points
Coderunner	100