Assignment #6: dictionary

11/13/2023

10/10 Points

Attempt 1 ⌄   ◯ **Review Feedback**   Attempt 1 Score: 🗨 View Feedback
                      11/13/2023   **10/10**

Anonymous Grading: **no**

Unlimited Attempts Allowed

⌄ **Details**

Develop a simple spell checker and word suggestion program using Hashing.

A dictionary of words should be given as a first argument (filename) in the program. This input file is the simple text file "Dictionary.txt", where each line is a word (see Files section).

The program must ask the user to give a word.

(case 1) The program must respond with "True" if the word exists in the dictionary, and must suggest words (from Dictionary) starting with the same first two letters. At the end of each suggestion the exec. time (in micro-sec) must be displayed (see "TimeInterval.h" in Files section).

(case 2) If the input word does not exist in Dictionary due to user misspelling, and considering the first letter is correct, the program should return a recommended word from the dictionary (only one word). For example, instead of the word *sentence,* the user enters *snetence, setnence, sentens,* or *setence*, etc; the program should return: "do you mean *sentence*?" If "no", return "false". If "yes", return "true", and continues as normal (returns words starting with the same two letters).

(case 3) If the input word does not exist in Dictionary (no "resemblance" with an existing word), the program should return "false".

Note: Any other case (e.g., the word contains only one letter), should be treated in any way you decide (make a note in the readme file and comment in the code).

Example:

user enters:

$ hi

the program responds:

$ True

hic

him

hip

his

hit

500 micro-sec

Required files (Files section)

Dictionary.txt

TimeInterval.h

Submit

a single file, **X_Y_PA6.zip** or **X_Y_PA6.tar.gz**, where X is your first name (capital letters) and Y is your last name (capital letters). You get this file from compressing a folder named **X_Y_PA6** containing the following:

- Your code files (.cpp, .h, etc.), **including** "Dictionary.txt" and "TimeInterval.h"
- A Makefile file that contains all the commands needed to compile your code on **Tesla**. All the code will be tested on Tesla with its g++. I should be able to compile your code by executing **make**
- A **README.txt** file showing how the users should execute/run your program
- A screen recording of your compilation/execution

Grading:

-5pts for not following cases (1, 2, and 3). Note: If word found, then case 1; if word not found, and you detect potential misspelling, then case 2; if user confirms misspelling, then case 1; if user rejects misspelling, then case 3; if word does not exist and no misspelling detected, then case 3. These cases yield a clear output (for example: only case 1 yields exec time, and only case 2 asks for confirmation of misspelling).

-8pts for not using hash table as dictionary

-1pt...-5pts for any minor implementation issues that yield wrong results yet the code looks mostly correct.
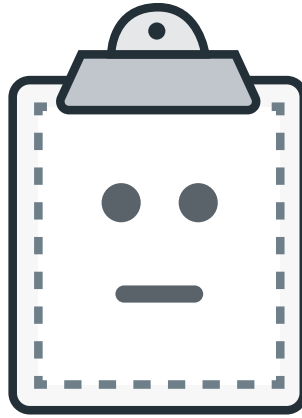
Preview Unavailable

PARMINDAR_SINGH_PA6.zip.zip

↓ **Download**

**(https://iu.instructure.com/files/165116205/download?
download_frd=1&verifier=9chot4gY5oC9xTCseYTglCUc4xs4nQLM3ciwS8qD)**

*You are unable to submit to this assignment as your enrollment in this course
has been concluded.*