DATA STRUCTURES
HOMEWORK #7

Due Date: 11/29/2018 @ 23.00

Submit your homework using the corresponding link on notebowl

Problem Definition

Many applications, including word processors, text editors, and email clients, include a spell-checking feature. A spell checker's job is relatively simple. Given a list of correctly-spelled words (which we'll call a *wordlist*) and the input text, the spell checker reports all of the *misspellings* (i.e. words that are not found in the wordlist) in the input. When a misspelling is found in the input, a good spell checker will also suggest words appearing in the wordlist that are somewhat like each of the misspelled words.

As an example, suppose that the wordlist contains the following words:

```
bake cake main rain vake
```

If the input file contains the word **vake**, a good spell checker will state that **vake** is misspelled, since it does not appear in the wordlist, and will suggest that perhaps **bake**, **cake**, or **vase** was the word that was actually intended.

Of course, a spell checker's task is centered around searching for words in a potentially large wordlist. An efficient search is critical to the performance of the spell checker, since it will be searching the wordlist not only for the words in the input, but possibly hundreds of suggestions that it might like to make for each misspelling. As you will see, a poor hashing technique can render a spell checker effectively useless.

Program

The program will read a wordlist from a file, then spell-check an input sentence. For each misspelling found in the input sentence, your program will report it, along with a sorted list of suggestions (if any).

To do this:

- Write a class named **HashTable** to implement your hash table that stores strings, using separate chaining (i.e. linked lists) to resolve collisions. The class has three operations:
  - **insert(I):** Adds an item I to the table bucket
  - **contains(I):** Returns true if the item I is in the set and false otherwise.
  - **Rehash():** If ratio of number of input by hash size is greater than the load factor, hash-size is doubled and re-insert all the elements in the old hash array to new table.

- Write a class **Word** that generate alternative words as describe below and generates 2 hash codes: one is using with std::hash<type >, the other one will be designed by you.
- Write a class called **WordChecker** that checks the spelling of one word and makes an appropriate set of suggestions for a misspelled word. You will use 4 techniques for generating suggestions for misspelled words

  - Swapping each adjacent pair of characters in the word.
  - In between each adjacent pair of characters in the word (also before the first character and after the last character), each letter from 'A' through 'Z' is inserted.
  - *Deleting each character from the word.
  - *Replacing each character in the word with each letter from 'A' through 'Z'.

- Files to submit
  - Wordlist.txt
  - Word.h
  - HashTable.h // Not generic class
  - WordChecker.cpp // Main class, it will read the wordlist.txt file and store words into a hash table. then spell-check an input sentence.

- Sample Output :

Menu

1. Load wordlist to Separate Chain Hash Table with std::hash function
2. Load wordlist to Separate Chain Hash Table with your hash function
3. *Load wordlist to Double Hashing Hash Table with std::hash function
4. *Load wordlist to Double Hashing Hash Table with your hash function
5. Run spell-checker

Enter (1- 4) >> 1
Results:
Initial hash table size: 101
Max Load factor: 4.00
X times rehashing were done.
Final hash table size is : XXX
Current load factor=YYY
Max chain length: ZZZ
Avg chain Length: QQQ
Print hash table
……..
Enter (1- 4) >> 3
Initial hash table size: 101
Max Load factor: 0.7
X times rehashing were done.
Final hash table size is : XXX
Current load factor=YYY
Max ZZZ times probe value was calculated
Total QQQ times probe value was calculated
Print hash table
……
Enter (1- 4) >> 5

```
Enter a sentence >>
This is a lne of text that has a missspelling in it.
Result
*********
word not found: LNE
perhaps you meant:
          LANE
          LEE
          LIE
          LINE
          LONE
          ONE
word not found: MISSSPELLING
perhaps you meant:
         MISSPELLING
Enter a sentence >>
I will learn hashing technique very well with this homework.
Result
*********
No misspelling in the sentence.
```