MSU CSC 232, Fall 2014
Lab 8. Use iterators to access a thesaurus
Given: Oct. 20, 2014

As an introduction, a "Madlib" game replaces certain words of a story with other words that are the same parts of speech but may not make as much sense. See this for yourself by playing a Madlib game at http://www.eduplace.com/tales/content/wwt_050.html.

In this lab, you will create a thesaurus of provided words, where each word in the thesaurus will have several synonyms. To use the thesaurus, iterate in sequence through each word of three provided sentences. If a word of the sentence is in the thesaurus, choose a random synonym to replace the word in the sentence. Print out all three sentences. The program should give different sentences every time it is run.

- Each sentence is a set of strings or words.
- Each paragraph is a set of sentences.
- A synonym is an alternate, replacement word for a given word.

Use iterators to access the elements in a sequence.

Important: Lab8Provided.cpp will use the auto feature. You must compile the program using
            g++ -std=c++11 Lab8Provided.cpp

The basics of words, synonyms, and a paragraph on which to run the thesaurus are provided in **Lab8Provided.cpp.**

Your lab tasks:

1. Complete a class **WordInThesaurus** which will contain synonyms for a given word. The class **WordInThesaurus** has the following characteristics:
- Private data
    - the word for which there exist synonyms
    - a set of synonyms for the word, probably contained in a vector of strings.
- constructor
    - constructor has one parameter, which initializes the word for which there exist synonyms. There must be one or more other public functions that add synonyms to the object.
- Five public methods:
    - **add_synonym(string s)** adds **s** to the set of synonyms for this word
    - **getWord()** has no parameters and returns the word itself
    - **getNumberOfSynonyms()** returns the number of synonyms of this word
    - **getSynonym(int i)** returns this word's **i**-th synonym
    - **getRandomSynonym()** returns a random synonym for this word

2. In the following steps you will use iterators to access data. The simplest form of iterator to use is an **auto** iterator. Initially, write the code however you wish, but at the end at least one of the iterators should NOT be an **auto** iterator.

3. At the bottom of the provided code (about line 236), use one or more iterators to display the provided paragraph. The paragraph should look like the following:

```
I ran a marathon and led several miles but the provided shoes
gave me blisters.

My team participated in every review and developed an excellently
planned product.

Our led zeppelin tribute polka band named demonstrated commitment
performed to sold out crowds.
```

4. Use one or more iterators to display the thesaurus. Your output for each word in the thesaurus should look similar to the following (note the word "and" prior to the last synonym):

```
The word Assisted has 5 synonyms: Accommodated, Collaborated,
Facilitated, Served, and Supported.
```

5. Use one or more iterators to examine each word of the provided paragraph. If a word of the sentence is in the thesaurus, choose a random synonym to replace that word in the sentence. Print out each modified sentence. The program should give different sentences every time it is run.

Sample expected output (in addition to the output shown above). *Each time your program runs, it most likely produces a different output.*

```
I administered a marathon and directed several miles but the dispensed
shoes gave me blisters.
My team cooperated in every review and coordinated an excellently
designed product.
Our controlled zeppelin tribute polka band named established commitment
completed to persuaded out crowds.
```

A different time you run your program, perhaps it will output:

```
I handled a marathon and managed several miles but the granted shoes
gave me blisters.
My team cooperated in every review and designed an excellently arranged
product.
Our engineered zeppelin tribute polka band named determined commitment
accomplished to persuaded out crowds.
```

Copy your C++ program **Lab8Provided.cpp** into your eccentric folder.
- Only that filename will be graded. I care not that you prefer a different filename.
- If you have changed the filename, I suggest you re-open the provided filename, replace its contents with your work, then save using the original provided filename.