CCSCI 275 Assignment #5
Due: Saturday 2/26/22
Late: Saturday 3/5/22 (10 point penalty)
Points: 60

Task:
This program looks for words in a dictionary. It practices dynamic arrays, and use of the algorithm library.

Input:
File **jumbledwords.txt** contains a list of words, one per line. File **poem.txt** contains some lines of the poem "Jabborworky" by Charles Lewis.

Processing:

1. Create a dictionary using a dynamic array of capacity 25000. The **jumbledwords.txt** file has about 20000 words so you will not need to worry about resizing. Read the words into the dictionary and sort it using the function in the **<algorithm>** library. Report the number of words.

2. Read each line of the input file and report words that are not found in the dictionary. Words should be lower case with no trailing punctuation. Hint: Check out the built-in **isalpha** function which is true for letter characters and false otherwise. Use the **binary_search** function in the **<algorithm>** library. Note this function is only applicable to a sorted array. Ask the user if the word should be added and do so, if requested. Stored added words also in a dynamic array of capacity 100. You will not need to worry about resizing.

3. Report the words added to the dictionary and its new size. Do not write the dictionary back to disk so you can run the program over and over for testing.

Functions:
Write the following two functions. Do not change the function prototype and honor the POST comment. Add other functions at your discretion.

```
// POST: dictionary is filled from file and size is number of words
void fillDictionary (string * dictionary, int& size);

// POST: word is now lower case with no trailing punctuation
void cleanup (string & word);
```

Required Code:
1. Code must use dynamic arrays.
3. No global variables/objects may be used.
4. No break or continue out of loops; the loop should end when the loop test is false.

Sample Output:

```
Dictionary size is 20068
Enter input file name: poem.txt
Did not find jabberwocky
Would you like to add it to dictionary (y/n)? y
Did not find twas
Would you like to add it to dictionary (y/n)? y
Did not find brillig
Would you like to add it to dictionary (y/n)? n
Did not find slithy
Would you like to add it to dictionary (y/n)? n
Did not find gyre
Would you like to add it to dictionary (y/n)? n
Did not find gimble
Would you like to add it to dictionary (y/n)? n
Did not find mimsy
Would you like to add it to dictionary (y/n)? n
Did not find mome
Would you like to add it to dictionary (y/n)? n
Did not find raths
Would you like to add it to dictionary (y/n)? n
Did not find jaws
Would you like to add it to dictionary (y/n)? y
Did not find claws
Would you like to add it to dictionary (y/n)? y
Did not find jubjub
Would you like to add it to dictionary (y/n)? n
Did not find frumious
Would you like to add it to dictionary (y/n)? n
List of words added to the dictionary:
jabberwocky
twas
jaws
claws
Dictionary size is 20072
```

Submission:

Upload the source code (**Source.cpp**) to Brightspace. I will download and run it for testing. Note: Code is not accepted that does not compile.

Need Help?

1. Email your question with your attached .cpp source code file. Do not attach an image or a pdf. Do not paste your code into the email body. I would like to download your code so I can test it as needed.

2. Use the scheduling software on the left side of Brightspace to schedule a Zoom meeting with me. If you cannot make the listed times, email me a list of your free times.