CST8130: Data Structures --- Assign #5- Dictionary

Using ArrayList OR Dyamically Allocated Array with Hashing Data Structure

DUE: Wednesday April 19th by 10PM SHARP!

Problem Description:

In this Assignment, we will write a program to handle inserts and searches to a dynamically allocated array using a hash algorithm and a collision resolution algorithm. We will rewrite Assignment 4 (which had O(log n) efficiency for insert and search) to have O(1) insert and search.

NOTE: you can choose to use an **ArrayList** or a Dynamically Allocated Array ([ ]) for this assignment. (Hint – you should know how to do this assignment for BOTH of these for Lab Test)

- Write a class called **DictionaryEntry** which has as data members, the **Strin**g word and the **int** (or **Integer**) count of number of that word.
- Write a class called **Dictionary** which holds the data structure you chose (**ArrayList** or **Array**) of **DictionaryEntry** objects.
    - use a hash algorithm to calculate the index of where to store the DictionaryEntry object in the data structure
    - the hash algorithm should be sum of each of the letters of the word (each char typecast to an int) – modulus size of the structure (so that you make sure the result is a valid index).
    - collision - if there is already a **DictionaryEntry** at this index position, display a message that the word cannot be inserted. **BONUS** – when you have a collision – move to the next sequential element position in the array until you find an empty location (but not past size of data structure! – if you get to the end of the data structure, display error message –"word cannot be added")
- In main, write a menu loop which allows these options
    1. Clear the data structure (and set new size)
    2. Enter a word (or text) from keyboard
    3. Enter text from a file
    4. Display count of a specific word
    5. Display number of entries in the structure (which is not the size – it is the number of entries with data)

Sample Output:

```
Enter 1 to clear dictionary,
2 to add text from keyboard,
3 to add text from a file,
4 to search for a word count,
5 to display number of entries,
6 to quit
4
Enter word to search for:
hello
532
hello does not occur in dictionary
```

```
Enter 1 to clear dictionary,
2 to add text from keyboard,
3 to add text from a file,
4 to search for a word count,
5 to display number of entries,
6 to quit
5
There are 0 entries

Enter 1 to clear dictionary,
2 to add text from keyboard,
3 to add text from a file,
4 to search for a word count,
5 to display number of entries,
6 to quit
2
Enter text (end input with space#):
hello there#

Enter 1 to clear dictionary,
2 to add text from keyboard,
3 to add text from a file,
4 to search for a word count,
5 to display number of entries,
6 to quit
2
Enter text (end input with space#):
hi he'llo#

Enter 1 to clear dictionary,
2 to add text from keyboard,
3 to add text from a file,
4 to search for a word count,
5 to display number of entries,
6 to quit
4
Enter word to search for:
hello
hello occurs 2 times
```

Submission:

You must submit to the assignment link in Blackboard by the due date and time a zip file (named LastnameFirstNameAssign5) containing:

- all source code – ie .java files
- Your test plan

Failure to provide any of the above will have an effect on your grade for this assignment. Marking guide will be published shortly.