**FourPicsCheater Assignment**

**50 Points**

Next Tuesday, I will give each person a 4Pics1Word Quiz, where you will be able to use your program to find solutions to some difficult 4Pics1Word questions.

This is a program designed to provide possible answers to the popular 4Pics1Word app. The user will provide the number of characters in the word and the 12 available letters and the program will respond with all words of that length that contain those letters.

A significant portion of the program is already written, included the part that uses a text file called dictionary.txt which includes 45402 English words. The program will load all of these words into an ArrayList of Strings called *dictionary*. The program works by removing words from the ArrayList that don't meet the criteria, then outputs those that remain as possible solutions.

Your job is to write 3 (or more, if you create helpers) methods to complete this program.

**Method # 1 - checkLength()**

This method will check every word in the dictionary ArrayList and remove all of those that are not of the size provided by the user.

You can verify this part of the program with the following data:

* There are 6176 six letter words.
* There are 7371 seven letter words.
* There are 1880 twelve letter words.

**Method # 2 - checkLetters()**

This method will compare the 12 letters entered by the user and compare them letters used in the *dictionary* ArrayList of Strings. If the String in the ArrayList contains a letter that is not in the letters provided by the user, it should be removed from the ArrayList, since it could not be the word we are searching for.

The biggest challenge to this assignment will be managing multiple occurrences of the same letter. For example, if the word we are searching for is *butter*, and the user would need to give us two Ts in their block of letters in order for that word to be considered as a possible solution. In other words, 6 (*numLetters*) and buteraaaaaaa (*letters*) should not return butter as a possible solution, but *rettubaaaaaa* should.

You will have to find a way to make sure that a given letter is only capable of being used once from the block of letters provided by the user. I created a *containsLetter* (boolean) method for this purpose, which will replace a matched letter in the letters char array. It will be necessary to create a duplicate copy of this array each time we need to compare it to a new String from the dictionary ArrayList, otherwise we will

**Method # 3 - OutputWords**

This one is the easiest of the three. You need only output (one word per line) the words that remain in the dictionary ArrayList. These are the possible answers to the 4Pics1Word question asked.

**Reminders**

1. If you are going to remove items from an ArrayList, traverse the ArrayList from back to front to avoid shifting indices and potentially skipping over elements in the ArrayList.
2. Remember that ArrayLists are manipulated with .get, and .remove methods.
3. You can use the String charAt(int) method to check the character at a given index in a String.
4. Use the break; command to exit a for loop to speed up the time it takes your program to process the data. For example, if I find a letter in the word that isn't in the 12 letters provided by the user, I should break out of that loop after removing it from the ArrayList, without checking any more letters.

Sample outputs

45402 words loaded.

Enter number of letters in word:

7

Enter available letters as a block of 12 letters: (ex: psbnoiyzvlvw)

jwoitwlnsisa

45402

After eliminated words of different length, 7371 remain.

After checking the letters provided to those left, 3 remain.

liaison

stalins

winslow

45402 words loaded.

Enter number of letters in word:

11

Enter available letters as a block of 12 letters: (ex: psbnoiyzvlvw)

ainoxminfrot

45402

After eliminated words of different length, 3069 remain.

After checking the letters provided to those left, 1 remain.

information