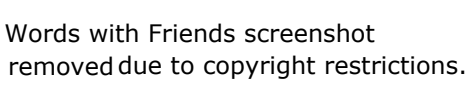
Introduction to Computers and Engineering Problem Solving
Spring 2012
Problem Set 4: Scrabble word generator
Due: 12 noon, March 16, 2012

Problem statement

You will write a program to help find possible two and three letter Scrabble words that one could make with a given 'hand' of 7 letters. The classes you should include are given, but you have freedom to design and implement the remaining methods and logic as you choose. We strongly encourage you to plan this assignment out on paper before you write your first line of code.



The list of all acceptable two and three letter Scrabble words has been placed in a class called `Dictionary`. The `Dictionary` class contains two arrays: a 1D String array called `twoLetterWords` in which each entry is a possible two-letter word, and a 2D String array called `threeLetterWords`, in which there are 26 rows that correspond to the 26 letters in the alphabet from A-Z (in order), and each row contains an array of three letter word Strings that all begin with that corresponding letter. The public methods `getTwoLetterWords()` and `getThreeLetterWords()` can be used to retrieve these arrays in the dictionary.

Program

Your program should contain:

`Solver` class

- This class will contain the `main()` method, which runs the program.
- Your `main()` method should prompt the user to enter the 7 letters, and then print the full list of possible words. You may assume that the user will always enter the letters in all capital letters.

`Dictionary` class

- This class contains the data members `twoLetterWords` and `threeLetterWords`, and the methods `getTwoLetterWords()` and `getThreeLetterWords()`.

`findWords()` method

This method manages the search. Place it in an appropriate class. Given a user's hand of letters, it returns an ArrayList of Strings that correspond to possible valid two or three letter words the user can make.

Example Outputs:

If your hand contains the letters: ZETRLCH, your program should output the following. Words can be listed in any order.

```
Possible two or three letter words are:
ET, ETH, ER, EL, EH, TEL, THE, RE, RET, REC, LEZ, LET, CEL, HE, HET, HER
```

If your hand contains the letters: RRRTTEZ, your program should output the following. Duplicate words in your solution are acceptable for this homework assignment.

```
Possible two or three letter words are:
RE, RET, TET, ER, ERR, ET
```

Hint 1:

The `String` class contains many methods you might need. Below are the method signatures of several methods in the `String` class that you might find useful in this assignment. Check the Javadoc to learn more.

`public int length()`

Returns the number of characters in a String.

`public char charAt(int index)`

Returns the character at the input index position of a String.

`public String concat(String str)`

Returns a new String that is the result of the original string concatenated with the input String `str`. For example, if `String a = 'hello'`, then `a.concat('there')` would return `'hellothere'`.

`public String substring(int beginIndex, int endIndex)`

Returns a new String that is the substring of the original String, from index position `beginIndex` to index position `endIndex-1`. For example, if `String a = 'hello'`, then `a.substring(2,4)` will return `'ll'`.

Hint 2:

If I have a variable `letter` that is of type `char`, the statement `letter - 'A'` will produce an integer value that reflects how many letters away `letter` is from `'A'`. For example, if `letter = 'C'`, then `letter - 'A'` will result in the integer 2.

Turn In

1. Place a comment with your full name, section, TA name and assignment number at the beginning of the .java file for your solution.
2. Place all of the files in your solution in a single zip file.
    a. Do not turn in copies of compiled byte code or backup (.class or .java~ files)
    b. Do not turn in printed copies of your solution.
3. Submit the single zip file on the 1.00 Web site under the appropriate section. For directions see *How To: Submit Homework* on the 1.00 Web site.
4. Your uploaded files should have a timestamp of no later than noon on the due date.
5. After you submit your solution, please recheck that you submitted your .java file. If you submitted your .class file, you will receive **zero credit**.

Penalties

- 30 points off if you turn in your problem set after Friday noon but before noon on the following Monday. You have one no-penalty late submission per term for a turn-in after Friday noon and before Monday noon.
- No credit if you turn in your problem set after noon on the following Monday.

1.00 / 1.001 / 1.002 Introduction to Computers and Engineering Problem Solving
Spring 2012