

CS 142

Project 2: “Word Jumbler”

Name: _____

A **Word Jumble** is a type of puzzle where the “puzzled” must re-order letters into an actual word.

Your task is to design and write a GUI-based **2-tier** program (*using the JavaFX Scene Builder*) in Java that, through a separate **Dictionary** class opens a standard English dictionary file (*where all words containing "" or "-", less than four letters or more than seven letters have been removed, and all words are all-caps*)--available on **Canvas**. It should open automatically, if it can, and should be able to be read in via a menu option and a **FileChooser** if it can't, or if the user wishes to change dictionaries.

Clicking a “**get JUMBLE**” button will cause a jumbled word to display (*created from a separate **PermutationGenerator** class*), along with an appropriately-sized array of **ComboBoxes**, set to blank, with options for all the available characters. It will also display the number of possible solutions. To accomplish this the program will need to select a random word from the dictionary and call a **recursive**¹ **scramble** method that repeatedly uses a **swap** method to swap pairs of randomly chosen letters a randomly chosen number of times (*greater than a defined minimum and less than a defined maximum*). This method should be called repeatedly until the scramble that comes up is not a word in the dictionary. The program should then call a **recursive**¹ method that returns a list of possible solutions.

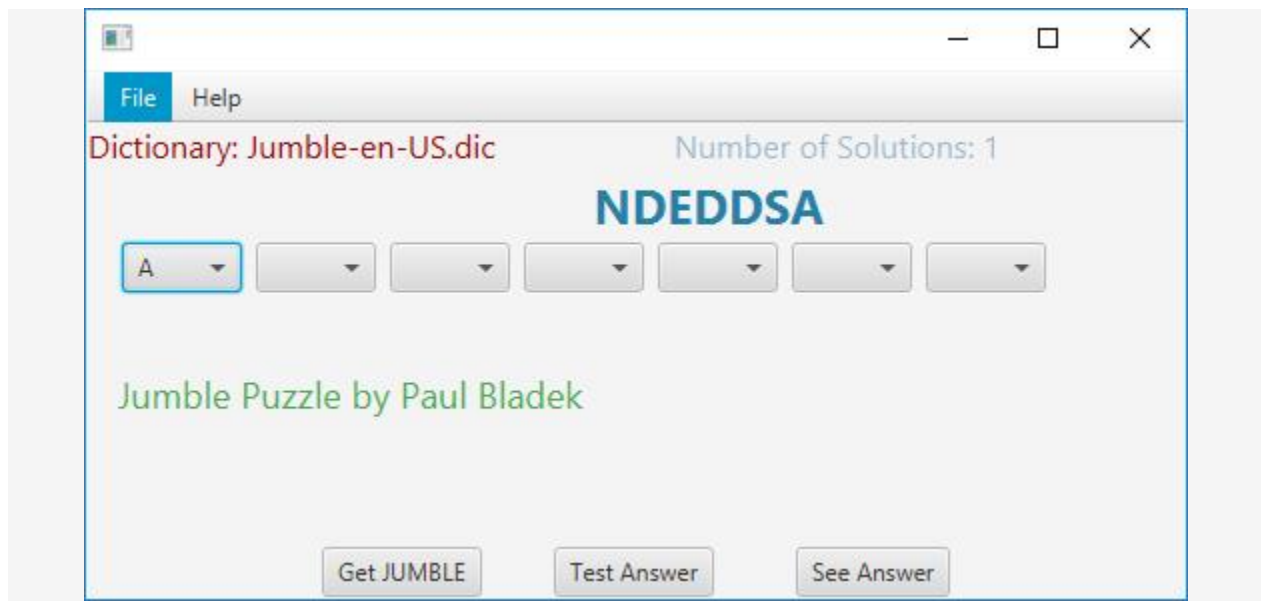
Using **recursion**², **this method** will reorder the letters in each possible way, and using a **recursive**¹ **binarySearch(Comparable[], Comparable, int, int)** -- in a separate **BinSearch** class -- will add the attempt to the list *iff* it is a word in the dictionary,

Separate buttons will allow the user to test to see if the chosen answer is correct or to see a list of all the possible answers.

Points Possible: 80

¹ **recursive** (rè-cur'-siv)adj. 1. *see recursion*

² **recursion** (rè-cur'-zhun)n. 1. *see recursion*



Deliverables:

Physical:

The Project should be turned in inside a clear **plastic** file folder. This folder should have a simple flap to hold paper in place--NO buttons, strings, Velcro, etc. Pages should be in order, **not** stapled.

- Assignment Sheet (*printed pdf from the web*), with your **name** on it, as a cover sheet.
- Printed Source Code with Comments (*including heading blocks -- a file header for each file plus a javadoc header for **each** class and method. Describe parameters, any input or output, etc., no line wrapping*). Print in portrait mode, 10 - 12 point font.

Electronic:

- All **.class**, **.jar**, **.html** (*javadocs*) and **.java** files.
- Sample Output (*as .rtf -- run the program, copy the window using <Alt/PrtScn>, paste into Paint, invert colors (<Ctrl/Shift/I>), copy, open Wordpad, paste, save.*)
- A simple test plan including explanations of any discrepancies and reasons for each test. Show actual input and **ALL** values output as well as **ALL** expected output. Test each possible **action**. Save as **.xls**, **.xlsx**, **.doc** or **.docx** file
- **Zip** all of the above files together. **Do not** use **rar** or any archive format other than **zip**. Rename the file: "<YourName>_p2.zip".
- Submit this single **zip** file by going to **Canvas**, select this class, select the **Assignment** tab on the left, select the **Assignment 2**, select the **submission** tab at the top, find the file, and **Submit**.

Due: Monday, May 8, 2017, 11:30 a.m. (*beginning of class*)