

CS162 - Assignment 4: Recursive Word Search

This assignment is intended to give you experience with a complex recursion problem. Turn this assignment in via TEACH. The due date is 02 September, 2012 by 23:59.

Demo Applet

Assignment Overview:

A word search is a word puzzle that consists of seemingly random letters arranged in a square or rectangular grid. The object of the game is to find all of the words from a given list hidden in the grid. Words may be hidden vertically, horizontally or diagonally within the grid. Specifically, a word may start at a particular row, column coordinate, and the rest of the word can be found by following one of eight possible directions: northwest, north, northeast, west, east, southwest, south, or southeast.

In this assignment, you will complete the WordSearchFrame class by writing functions to find the words in a list in a word search. Your tasks are as follows:

1. Complete the actionPerformed(ActionEvent event) function in the SearchListener.
The first time the search button is clicked, the first word in the word list changes color to red. The program then finds that word in the word search. If the word is found, the letters corresponding to that word are changed to red on the word search. Note that if the word is hidden multiple times in the word search, you only need to find the word once (there is no need to find all instances of that word). When the search button is clicked again, the first word in the word list changes color back to black. The letters of the first word in the word search are also converted back to black. The second word now becomes red and the program finds and highlights the second word in the word search puzzle. This continues until the end of the word list, at which point, the current word wraps back to the first word in the word list. You can play with the applet below to get a feel for how this works.
2. In order to do the first task, you will need a find function to look for a specific word in the word search. **Your find function will consist of two parts. The starter function loops through each row, column, and direction. For a given row, column, and direction, you need to write a helper function which finds the word recursively. Note that you will get zero on this part if your find helper function is not recursive.** Remember that a recursive function has a base case that tells it when to stop and a recursive step in which the function calls itself with a simpler input.

In order to run the code, you will need to pass in command line arguments (take a look at assignment #2 if you forget how to do this). There are two command line arguments: a word search file, and a word list file. You can download samples from below. Remember that you may have to specify the full path name of your files eg. C:\\CS162\\assn4\\wordsearch.txt instead of just wordsearch.txt

Finally, you may add functions and member variables to the code as you wish.

Code

The following are files for the assignment:

- [WordSearchException.java](#) which contains code for a custom exception class for the word search.
- [WordSearchFrame.java](#) which contains code for the word search frame.

Files

There are also sample word search files and their corresponding word lists.

- [java.wordsearch.txt](#) and [java.wordlist.txt](#) which is a word search of java terms

FAQ / Hints

- **When I download the wordsearch and wordlist files from my web browser, why does everything end up on one line?**
In your web browser, click on the wordsearch and/or wordlist file. Highlight everything and select "Copy" from the "Edit" menu. Then, create an empty text file. You can create an empty text file in Windows by right-clicking on the desktop or your assignment folder and selecting New -> Text Document. Open the new text file by double-clicking on it, select "Paste" from the "Edit" menu. Then, make sure the last line of the file is in fact the last line. Go to the last character, hit Delete a bunch of times.
If you're using Unix, you can use vi or emacs to open a new text file.
- **I keep getting the "Error: only one word per line allowed in the word list"**
To fix this problem, cut and paste the word list or word search to a new file. Then make sure the last line of the file is in fact the last line. Do this by going to the last character in the file, hit the "Delete" key a bunch of times.
- Make sure your command line arguments are in the following order: [word search file] [word list file]

What to Turn In: (Remember to turn in the .java files NOT the .class files)

Please hand in all the .java files for your assignment. Turn them in via TEACH

- All .java files for your assignment. Do NOT hand in the .class files. If you have more than 10 Java files, please zip up your handin files and submit as a .zip file.

Grading Scheme:

You will be graded on the correctness of the recursion.