

CS 145
Assignment 1 – Word Search Generator

Key topics:

Printing, data types, methods, operators, expressions, variables, for loop, parameters, returns, String objects, Scanner object, if/else statements, while loop, arrays

Learning Outcomes:

- Begin to become familiar with setup, design, execution and testing of basic Java programs
- Design and develop a multi-method program in good style
- Demonstrate the ability to decompose a problem and reduce redundancy using methods
- Apply tools and techniques introduced in class to form a working program
- Produce a working complex program that includes user input

By the end of this assignment, you should be familiar with the material covered in class so far and be able to demonstrate the ability to use it properly in practice.

Task:

Your task is to create a basic word search generator. A word search is a puzzle with many letters and inside are several words that a person is supposed to find. For information about word search puzzles, you can read the Wikipedia article at https://en.wikipedia.org/wiki/Word_search.

You will be creating a program that takes in a certain number of words from the user and generates a word search using those words. Initially, your program should display an intro to the user describing what the program does. There should then be options for the user to select by typing in a letter. The user should be able to perform several actions during one run of your program. A sample run of your program might look like this:

```
Welcome to my word search generator!  
This programs will allow you to generate your own word search puzzle  
Please select and option:  
Generate a new word search (g)  
Print out your word search (p)  
Show the solution to your word search (s)  
Quit the program (q)  
g  
|
```

In the instance above, the user typed a g and would then proceed to generating a new word search. After generating a new words search, the menu should come up again and prompt the user for another action. This should occur after every action until the user inputs "q" to quit the program.

When the user decides to generate a new word search, the program should prompt them for how many words they would like to enter and proceed to ask the user for each of these words. The program would then use these words to generate the word search. Note that you may need to adjust the size of your word search to accommodate more words or larger words. It is ok to limit the number of tries placing a word to 100. If you try to place a word within the word search more than 100 times, you may omit that word from the word search.

Printing out the word search simply prints out the word search that has been generated by the user. If the user has not generated a word search and attempts to print, you should prompt them to generate a word search and display the menu again and wait for the user to choose an action.

Selecting "s" to show the solution should display the word search, but with all the random letters as "X". This means that only the words input by the user would be visible and it would be easy to see the solution to the word search.

Implementation Details:

Your program must include at least the following method (you are welcome to and encouraged to have more):

printIntro

This method prints out the intro to the program. It gives the user an idea of what the program does and how to use it.

generate

This method does the bulk of the work. It will prompt the user for how many words and what the words are and generate a word search based on those words.

print

This method prints the current word search that has been generated.

showSolution

This method prints the solution to the word search that has been generated.

Helpful Information:

Since you will be able to show the solution and print the word search, it may be helpful to store two copies of your word search, one that is the generated word search with the random letters included and one that has "X" for all of the random letters instead so the solution is easy to see. You should be storing your word search in a two dimensional array (an array of arrays).

You may use any material covered in class up through chapter 8. You may make use of libraries that are introduced in those chapters as well. However, you may not use other external libraries beyond what are covered in the book through chapter 8.

I strongly encourage you to tackle this assignment in parts and practice iterative design. Work on a few things and then test them to make sure they work. Then, add some more and test those. Make sure that you come up with a plan before you start coding. Writing pseudocode can also be very helpful. I encourage you to speak with your classmates about your development plan for this assignment. I am also available to discuss how you might approach this assignment. Feel free to reach out to me with any questions or concerns.

Style:

It is important that you get used to writing code in good style. What is demonstrated in examples in class is considered good style. Additionally, you should look at the style guide located on Canvas. Code written not in good style will lose points.

Extra Credit Opportunities (Optional);

You can choose to do as many or as few of these as you would like. Each one completed successfully and in good style will earn some extra credit points towards this assignment.

1. Instead of having the user input all of their words, you may make it so that your program offers the option to read the list of words from a text (.txt) files instead. You must allow the user to choose if they want to read from the file or input the words manually. Completing this task successfully and in good style will result in extra credit.
2. You may also implement the ability to save a word search and its solution to a file (using PrintStream objects). You can add the option to the menu to allow the user to save their word search and its solution. Completing this task successfully and in good style will result in extra credit.

3. If you think of additional, cool functionality that you would like to add to your word search generator, discuss it with me and I may be open to allowing it to be an extra credit opportunity as well. You must discuss the functionality with me before your assignment is turned in and I may or may not deem it worthy of extra credit. I will let you know when we discuss the functionality if it will be worth some or not.

Grading:

Style	10
printIntro	5
generate	35
print	15
showSolution	15
Program functionality	20
Total:	100