Algorithm Design

LAB 05 : ADVANCED SEARCH DESIGN

Due Saturday at 5:00 PM MST

This week, we will create a design for an advanced search algorithm.

Program Description

The traditional search algorithm involves starting at the beginning of a list and progressing to the end, visiting each element in succession. While this will certainly work, it is not very efficient. Imagine looking up a word in a dictionary using this algorithm! When working with a list that is sorted (alphabetized, for example), we can use a much more efficient design.

Consider a person looking up the word "glitch" in a dictionary. At first, the word could be in any of the 688 pages in the dictionary. The user cuts the dictionary in half (to page 344) and notices the word "magnet" at the top of the page. This word sorts after "glitch." Because we can assume that the dictionary is sorted, we can deduce that the word "glitch" must be between pages 1 and 343. The user then cuts the remaining pages in half (to page 171) and finds the word "drift" on that page. Since "drift" sorts before "glitch", we can rule out pages 1 through 171. Thus, the possible pages must be 172 through 343. The user then cuts to page 257 and finds the word "guidance." Since "guidance" sorts after "glitch," we can rule out pages 257 through 343. This leaves pages 172 through 256. This process continues until the page containing "glitch" is located.

Your program must do the following:

- Prompt the user for a file containing an alphabetically ordered collection of words.
- Read the file in JSON format.
- Prompt the user for a word to search for.
- Perform a the advanced search described above to locate the word.
- Display a message indicating whether the word exists in the list.

To see how the program will function, look at

For this assignment, our job is to create a pseudocode design of the algorithm. Make sure your pseudocode uses a fixed-width font such as Consolas or Courier. Also, make sure you single-space your code. You might also need to set the space-before and space-after to 0.

Assignment

To submit this assignment, one thing is needed: a pseudocode program. This will be submitted through I-Learn as a **single-file PDF**.

As with two weeks ago, please use the "Comments..." field to answer the following questions:

- How long did it take for you to create this pseudocode?
- What was the hardest part of the assignment?
- Was there anything unclear about the instructions or how you were to complete this lab?

Assessment

Your grade for this activity will be according to the following rubric:

	Exceptional 100%	Good 90%	Acceptable 70%	Developing 50%	Missing 0%
Pseudocode Format 30%	All of the pseudocode conventions are used correctly	There is one minor discrepancy	At least two elements are used correctly but at least two elements are not used correctly	One or two elements resemble pseudocode	Pseudocode is missing or the attached document does not contain anything looking like pseudocode
Design Quality 50%	The most elegant and correct solution is found	The design completely covers the problem definition	One aspect of the problem definition is missing or one aspect of the design will not work as expected	Elements of the solution are present	Pseudocode is missing or the provided solution does not resemble the problem definition
Professionalism 20%	Professional, beautiful, elegant, single-spaced, using a fixed-width font	Everything is clear and legible	Misspelling, smudge, or examples of unprofessionalism	At least one aspect of the design is too messy to read	Difficult or impossible to read