Behjat Bahmani

Professor Suporn Chenhansa

CS-124-01

26 February 2021

Lab 2

Purpose:

This assignment mostly looks like the last assignment, I will try to clean (that is a challenge to remove all space and other punctuations) and read three text files and also figure out how to use five algorithms such as selection sort and a bubble sort, insertion sort, bubble sort, and quicksort. I can write how I should write the code for each algorithm and compare them together. I should figure out how to use an array or vector for coding. I think vectors are more appropriate because our text files are long. Moreover, learn the differences between them such as how long they need time to parse the data(sorted or unsorted), and also I can learn which algorithm is more proper for each data. After that, I figured out how to run a binary search algorithm on sorted data and also run a linear search algorithm on unsorted data five times and calculate the time of them. In the end, the user can ask to print the last 50 words from the sorted list.

Plan:

Since all of the functions except bubble sort are in the book and the bubble sort function is already done in class. I'll be reworking to fit them again and change them. I should clean the text file and read it and also push-back them in vector. (I chose vectors because I am not sure about the size of our file text. It is possible to do it by array but vector is better) Firstly, I am testing each function once it's done and once the program is done I'll test it with shortened text files to

make sure it works in addition to the main text file. I should write a code to calculate the time of them so I can compare them together. After that, I ask the user if he/she wants to search for a word. I try to write a code to search linear and binary. At the end print 50 last words of the text file.

Input:

1. Users choose he/she want to continue or not (Yes or No).
2. Users choose two functions from five functions.
3. Users choose two texts from three files.
4. Ask the user he/she wants to choose a word.
5. Ask the user he/she wants to print 50 last words.

Rework Functions:

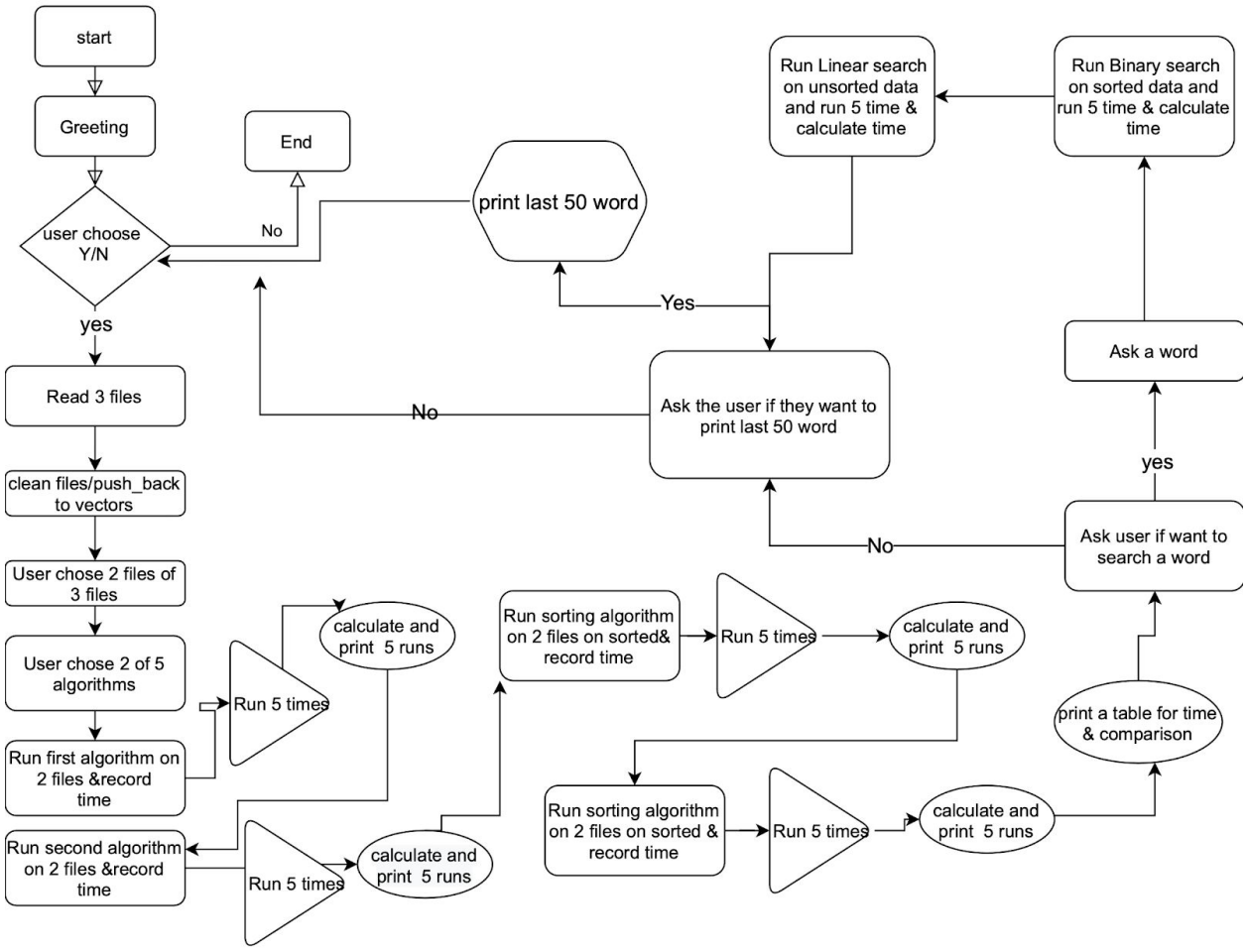
1. I should change all the algorithms to sort words in alphabetical order instead of numbers. Because of this, I choose a vector of string instead of an array.
2. After that, the function swaps the string in a vector instead of a number in an array.

Functions:

1. main function: I will read my three files in the main function and the end push-back them in the vector.
2. Void parseText:and try to clean them
3. Void selectionSort: Find the smallest element and move it in index[0] and swap the value of index[0] to the last minimum index and continue.

4. Void bubbleSort: Compare every two indexes and swap the minimum of them on the left side.
5. Void insertionSort: function starts to read the array and move the small value to the right place on the left side.
6. Void mergeSort: The merge sort algorithm sorts an array by cutting the array in half, recursively sorting each half, then merging the sorted halves.
7. Void quickSort: The sort a range a [from]... an [to] of the array, first rearrange the elements in the range so the no element in the range a[from]...a[p] is larger than any element in the range a[p+1]... a[to]. This step is called partitioning the range. Sort each partition, by recursively applying the same algorithm on the two partitions. That sorts the entire range because the largest element in the first partition is at most as large as the smallest element in the second partition.
8. void binarySearch
9. void linearSearch
10. Void print: this function prints the last 50 words.

- I don't know exactly of each website I will use now but most of the time I use: https://stackoverflow.com/ definitely if I will use it I will reference it in Lab2.


start
Greeting
user choose Y/N
No
End
yes
Read 3 files
clean files/push_back to vectors
User chose 2 files of 3 files
User chose 2 of 5 algorithms
Run first algorithm on 2 files &record time
Run 5 times
calculate and print 5 runs
Run second algorithm on 2 files &record time
Run 5 times
calculate and print 5 runs
print last 50 word
Ask the user if they want to print last 50 word
No
Yes
Run Linear search on unsorted data and run 5 time & calculate time
Run Binary search on sorted data and run 5 time & calculate time
Ask user if want to search a word
No
yes
Ask a word
Run sorting algorithm on 2 files on sorted& record time
Run 5 times
calculate and print 5 runs
Run sorting algorithm on 2 files on sorted & record time
Run 5 times
calculate and print 5 runs
print a table for time & comparison