CS401 Project

1. Project Description

Create to compare running complexity among data search algorithms. You need to compare:

1.1 Unsorted list in array and linked list
1.2 Sorted list (make the same data from the UL (by 1) but sorted data (refer a sorting algorithm in chapter 11, you can choose one)
1.3 Binary Search Tree
1.4 Total of 5 structures (use only one data set)

Requirements

1. Read a data file to store data into a data structure (both array and linked list each).
2. Data in a data file: product name, customer name, or any string (size up to 100)
3. Search target data and compare run time to conclude (found or not found)
4. Use sequential search for 1.1 and 1.2 (four structures) and compare running complexities
5. Use Binary search of sorted array and use BST
6. Additional data: generate 2,000 random numbers and search data
7. Your project also produces a Big-O notation per ADT and per algorithm based on the result

Thus, your project uses two sets of data:
(1) your string data file and
(2) random generated numbers (integer number)

How to compare run time or complexity? Choose one.

1. An easy way is to count comparison or
2. Use system running time measurement, or
3. Something else you create.

** Count only for searching data (i.e. each comparison in a loop) but do not count any other algorithm such as sorting data

GUI program is not required but if your project runs as full GUI (mouse can choose project menu) frame and if your GUI is easy to operate, your project may be qualified to earn extra credit (up to 5%). Command window GUI is not considered as a full GUI.

2. Additional Requirements

- ➢ Your project has a menu to select each algorithm to run and run all once
- ➢ When your project runs, provide instructions how to run
- ➢ When a list is created, print out the list (UL)
- ➢ Use both implementations (array and LL: such as bounded and unbounded model)
- No standard interface is given. (Use your own idea)

Programming language: JAVA ONLY

What to submit?

Submit softcopy through the Course BB

1. Documents based on Software Development Life Cycle
   a. Problem specification – What problems are solving?
   b. Software specification - What functions are there?

c. Design diagram document (including UML diagram and flow charts or pseudo code)
d. Source codes (with detail comments)
e. Operational document (user's manual: how to use your program, what is expected result or screen shots)
f. Testing document (your own created one) with input data file
g. Debugging note (if used or skip)
h. Future improvement document (if available)
i. Project management/schedule – daily, weekly progress plan
Hours per each task to be done

2. Complexity analysis based on your results with the theory you learn

- Every file name should include your Last and First name.
- Submit multiple files as one submission in the BB

- All documents except the source codes formats: either PDF or MS Word

- Do not submit the Google doc link for file (we had some issues in the past)

Final Project Due Date: ***December 1st, 2020 Tuesday*** 23:59 or earlier

- Late submission penalty: 10% in every 2 hours for 12 hours
  - e.g: submit on 5/2 3:59 AM – 20% penalty
  - e.g: submit on 5/2 12:01 PM or later – No accept

- When TA needs more instruction than the operational document to review your project, you may get a minor deduction (up to 5%).

- No demo or re-evaluation after final exam. The final exam means END of the semester. No more extra efforts can be allowed by the department.

- Suggest this project as a team project (up to two team members). TA will provide list of the class so you can find your mate. If you prefer to do alone, you can.

- If you form a team, only a primary submitter of the team submits the full submission. The secondary person only submits a cover page with your team mate info so TA can figure out who is the primary submitter of the team. Both will get the same score unless you form a team but the secondary partner does not cooperate well. There is no free rider. Each member must work equal amount time and efforts. If you think it's not, any team member can notify a report to your TA then the free rider will get 0 point.