Design for Final Database Project

Bryant Wong 107571009

Due Date: 4/5/2017

Program Description:

This program will be a database system that stores 15 different fields of contact information per entry found in an address book. Users will be able to search the database using any of the 15 fields as well as the search mode and field, and also run a secondary search, read data from a file, write data to a file, delete a record, modify a record, insert new records, and overwrite the read file or save a new file. Users should be able to create an output in text format of specifically chosen contacts, the specific fields they wish to show about those contacts, and to sort the final output based on a key and to sort the output on a secondary key.

Input Requirements:

For menu: a char to select menu choice of:

1. Read File

2. Search Entries: a char to select which field.

2a. A char to select "exact" or "contains"

2b. A string to enter what to search for

2c. A char to select if they wish to run a secondary search or new search

2d. A char to select "exact" or "contains"

2e. A string to enter what to search for

3. Edit entry: (runs search function first to find entry to be edited)

3a. a char to select which field to edit

3b. a string to change the data in the field. All names must be composed of letters, all phone numbers must be 10 digits and numbers, all emails must include @ and .

4. Delete entry: (runs search function first to find entry to be edited)

5. Add entry: 14 strings for the first 14 fields, plus 4 more strings for affiliates information.

5a. All names must be composed of letters, using isalpha()

5b. All phone numbers must be 10 digits and numbers using isdigit()

5c. All email must include @ and .

6. Output to file: an int to represent how many contacts the output file will have.

6a. a char for each field they wish to include.

6b. a char to select which field to sort entries by.

6c. a char to select whether they want to sort entries by another field.

6d. a char to select which field to sort entries by.

6e. a char to select overwrite read in file or save as file

6f. a string to enter file name

7. Exit

Output Requirements:

Maximum 14 strings + 4 strings(for affiliates). All names must be composed of letters, all phone numbers must be 10 digits and numbers, all emails must include @ and .

Problem solution discussion and Data Structures.

Menu while loop for choosing add, delete, edit, search entry (log n), read from file, print to file, exit. Additional menus in search entry for secondary searches, in edit entry for which field to edit, and in print to file to sort (n^2) or secondary sort (n^2) and which field to sort for. Searching for "contains" will probably use a recursive function to step through the data of the field one character at a time to find the user input with complexity of (n^n).

The searching and secondary searching should be done in the primary binary search tree and a secondary binary search tree because searching a search tree has much lower complexity (log n) than sorting something like a vector and then searching it, despite the fact that creating the secondary tree would have complexity of (n^2). The sorting and secondary sorting will be done in a vector I believe because, while sorting an array does have a complexity of (n^2), the trade off of using a binary search tree to sort, which is essentially already sorted, actually requires more time because populating the tree in the first place has a complexity of (n^2) and if the user wants to apply a secondary sort, the result of the first sort would have to be added to a secondary binary tree with the same complexity. It really depends on how much the user wants to search, sort, add, edit, or delete, so in the cases where the user would be searching more, a binary search tree will be used while the cases where the user would be sorting more, a vector will be used.

User Interface:

Menu

1. Read File

2. Add entry

3. Search Entry

3a. Enter 1-15 for which field you would like to search for

3b. Choose "exact" or "contains"

3c. Enter data to search for

3d. Would you like to refine your search?

3e. Enter 1-15 for which field you would like to search for

3f. Choose "exact" or "contains".

3g. Enter data to search for

4. Edit entry

4a. Enter field to search for (runs through menus in choice 3)

4b. Choose the item you wish to edit

4c. Enter field to edit

4d. Enter data you would like to change that field to.

4e. Would you like to edit another field in this entry?

5. Delete entry

5a. Enter field to search for (runs through menus in choice 3)

5b. Choose the item you wish to delete

6. Write to file

6a. Enter the number of contacts you would like to write to file

6b. Choose the fields you would like have displayed (enter some char when finished)

6c. Would you like to sort them?

6d. (If yes), Enter field you would like to sort by

6e. Would you like to sort as another field?

6f. (If yes), Enter field you would like to sort by

7. Exit

Data Flow Chart

Data Flow Chart.emf

UML Class Diagram

