

Lab 3 Tasks

Q1. This is a continuation of the phonebook project from the previous labs:

Currently, the contacts that are saved by the user in the phonebook are lost when the program is closed. There should be a way to access all the contacts that have been saved in all the previous runs of the phonebook program. This can be ensured by writing down the contacts in a file when the program is being closed and reading the file when the program is run.

When the user defined `exit()` function is called by your phonebook program, the function will write down all the elements of the linked list into a file.

[Hint: The line `fwrite(current, sizeof(struct contact), 1, fp)` will write down the information stored in a single element of structure `contact` type. `current` is a pointer of structure `contact` type. It is pointing to the element that is being written]

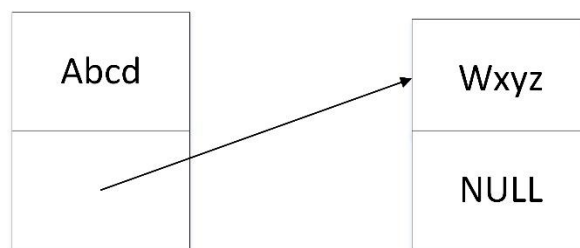
When the phonebook program is run, at first the main function will call a user defined function called `load()`. This function will read the file and store the information in the linked list.

[Hint: If the information in the file is written as the elements of a structure then it can also be read as the elements of the structure. The line `fread(new_contact, sizeof(struct contact), 1, fp)` will read from the file and store the information in an element of structure `contact` type which pointed to by the pointer `new_contact`]

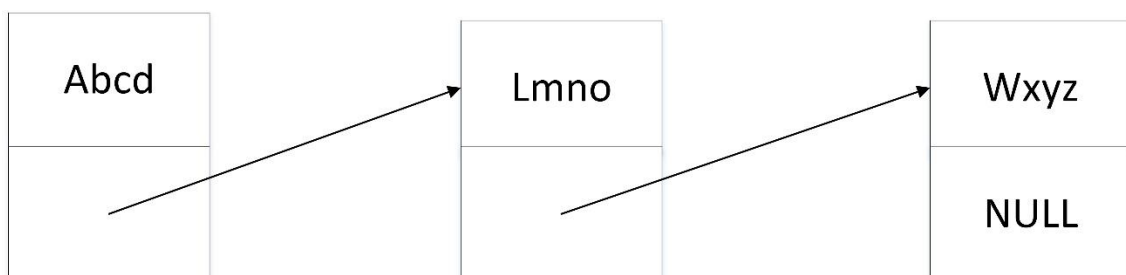
Q2. Usually in a phonebook when all the contacts are displayed in the screen, they are shown alphabetically. One possible solution can be to loop through the entire linked list multiple times and each time searching and printing the contact that comes next alphabetically.

However, a more sufficient solution would be to make changes to the code that adds new contacts to the link list. Instead of always adding new elements at the very end of the linked list, add the elements in the alphabetical order of their names.

For example, suppose there are two existing contacts in the linked list. The name of the first contact is 'Abcd' and the name of the second contact is 'Wxyz'.



Now, if a new contact named 'Lmno' is added in the phonebook previously it would have been added after 'Wxyz'. But now it will be added after 'Abcd'. You can do this by using the `strcmp()` library function. The comparison should not be case sensitive.



Now, when the contacts are printed using the view contact option, they will be shown alphabetically.