Future University in Egypt
Faculty of Computers and Information Technology
Computer Science Department
Semester: Spring 2023
Course: CS222/CS 222 Data Structures
Lecturer: Dr. Sabah
Teaching Assistants: AL. Yasmin Amr / TA. Aya Morsy / TA. Nadeen Ramadan

I. Phone Book Problem (Doubly Linked List)

You have a phone book but you are not quite sure how many entries this telephone book will have. So you decide to use ***a doubly linked list*** to represent your phone book and to allow you to dynamically add some ***Contacts*** to phone book. Each contact will consists of (A person's first name, person's last name, person's contact number). Implement the contacts class and linked list class to keep track of contacts and identify any needed class functions.

Once your program is started, the user will interact with it using a menu like the one below:

Choose one of the following:

1. **Display all names and phone numbers sorted by First name.**
2. **Display all names and phone numbers starting with a given letter.**
3. **Search for a name (by last name).**
4. **Add a new contact as last element in the phone book.**
5. **Delete a person from the list (using his/her last name).**
6. **Add a new contact after specific contact (last name).**
7. **Exit the phone book.**

Option 1 - if the list contains some elements, every time you choose option 1, your program should display this list of users sorted by their first name after asking the user about the type of sorting *[Bubble Sort, Insertion Sort or Selection Sort]*. If the list is empty, an appropriate message should be displayed.

Option 2 – here you will have to ask the user to enter a letter. If the list contains some ***first*** names which start with this letter, then **All** such names should be displayed. If not, then an appropriate message should be displayed.

Option 3 – Display the element that matches the first name that is equal to the last name provided by the user.

Option 4 – Ask the user to enter the new contact needed information, and then insert that in your phone book. Make sure to make the user aware if the program has any limitations such as size of name, etc.

Option 5- Ask the user to enter the last name of the person to be removed. Make sure to show some feedback to the user stating that user "x" has been removed, or person "x" was not found.

Option 6- Ask the user about the new contact data, and the last name for the customer you want to add the new contact after.

Option 7- exit the program.

Keep the menu running constantly until user chooses to quit the program.

II. Printer Service (*Queue*)

A company has one printer that can be used by any of its employees. A print job can be processed and completed every 5 minutes, and requests to use the printer are accepted from any employee every minute. This can be represented by a list of integers: every minute you get a new integer which is one of the following:

- 0 means no one has requested permission to use the printer
- -99 means that the company must close for the night (and any remaining jobs will have to be cancelled).
- Any other number represents the ID of the job for which a print request is being issued.
- For example, you will ask user to enter a set of jobs consider the following list of integers:
  54, 27, 0, 0, 82, 0, 23, 0, 24, 1, 0, 0, 13, 0, 0, 0, 89, 12, 234, 32, 66, 18, -99

Given the above list, the your program output should look like this

Time t=1. A request for print job 54 is issued
Time t=2. A request for print job 27 is issued
Time t=5. A request for print job 82 is issued Time
t=5. Job 54 is processed
Time t=7. A request for print job 23 is issued
Time t=9. A request for print job 24 is issued
Time t=10. A request for print job 1 is issued
Time t=10. Job 27 is processed
Time t=13. A request for print job 13 is issued
Time t=15. Job 82 is processed
Time t=17 A request for print job 89 is issued
Time t=18. A request for print job 12 is issued
Time t=19. A request for print job 234 is issued
Time t=20. A request for print job 32 is issued Time
t=20. Job 23 is processed
Time t=21. A request for print job 66 is issued
Time t=22. A request for print job 18 is issued
Company is now closing
Remaining number of jobs in the queue: 9

Hints:

- Start out by writing a loop that reads an integer and keeps reading integers until it reads a -99.
- Next, modify the above loop so that it prints out the "time" every time it reads an integer.
- You must check the queue's isEmpty status before dequeueing a print job.
- Also, you may wish to provide a new member function in the queue class, called count, that returns the number of items currently in the queue

Evaluating Arithmetic Expression (*Trees*)

Given an expression as input an (infix expression – postfix expression – Prefix expression), write a program to evaluate the value of the expression.

Example : Ans is 21.