GTU Department of Computer Engineering

CSE 222/505 - Spring 2023

Homework 3

Due date: April 5, 2023 – 23:59

In this homework, you are supposed to modify the program you have written for the first homework.

NEW FEATURES OF YOUR SOCIAL MEDIA

In addition to the existing features of your social media, a user can also do the following actions:

- Unlike a post they liked
- Unblock a blocked account
- Uncomment a post they commented
- Unfollow an account they followed
- Show their own action history. For instance, assume that gizemsungu logged and did some actions. When she wants to see her history, your program provides to show the following list related to her actions:
    - You liked sibelgulmez’s post id: 2
    - You followed sibelgulmez
    - You followed gokhankaya
    - You unfollowed sibelgulmez
    - You commented gokhankaya’s post id:3
    - You followed sibelgulmez

You have to create a test scenario (like you did in your first homework) to show that your program is allowed to do all the new features above. In your new test scenario(Scenario 4), you have to create 10 accounts to use in the actions of the test scenario.

NEW IMPLEMENTATIONS TO YOUR SOCIAL MEDIA

In the first homework, you are allowed to use only the basic array data structure. This time you have to use the List interface in Java Collections Framework. You should generate three separate versions of your program by using following three different implementations of the List interface;

a) Array List structure, ArrayList in Java Collections Framework,

b) Linked List structure, LinkedList in Java Collections Framework,

c) Your own Linked list structure named LDLinkedList. LDLinkedList should implement the List interface and extend the AbstractList in Java Collections Framework. You should use lazy deletion strategy in your Linked List implementation. In lazy deletion strategy, a node is deleted from its linked list logically but not physically. This is done by marking the node as “lazily deleted” (using a boolean value). The removed node is kept in the list with unremoved nodes until another node is removed from the same list. When two nodes are deleted from the list, then the list is traversed and two “lazily deleted” nodes are removed from the list **pyhsically**. For instance, gizemsungu follows sibelgulmez, gokhankaya, account3 and account4. Then, each account becomes a new node in gizemsungu's following list. When gizemsungu unfollows account3, account3 is marked as "lazily deleted". After that, if gizemsungu also unfollows sibelgulmez, then these two accounts are deleted pyhsically from the following list.

EXPERIMENTAL STUDY

You should analyze time complexity of four versions (including the one in the first homework). Analyze the theoretical running time. Measure experimental running time of your code for your test scenarios (including three test scenarios in the first homework) and compare it with the theoretical running time. Fill the following table with the experimantal running times of your implementations.

Implementation Type	Scenario 1	Scenario 2	Scenario 3	Scenario 4
Basic Array Structure (HW1)				Not available
Array List Structure (a)				
Linked List Structure (b)				
LD Linked List Structure (c)				

- Scenario 4 cannot be tested with HW1 implementations due to the lack of the new features.
- Running times will be given in seconds.

REPORT RULES

Your report should include for each implementatio:

- Running command and results
- Time complexity analysis
- Experimental running time analysis (with the table)
- Problem solution approach for LDLinkedList

GRADING

- No OOP design: -100
- No method overriding: -95
- No error handling: -50 –
- No inheritance: -95
- No polymorphism: -95
- No report: -90
- Disobey restrictions: -100
- Cheating in any part of your homework: -200
- Your solution is evaluated over 100 as your performance.