GTU Department of Computer Engineering
CSE 222/505 - Spring 2023
Homework 1
Due date: March 19, 2023 – 23:59

This homework is designed for you

- to review OOP concepts and Java programming,
- to get used to reporting conventions that will be used in the course.

In this homework, you are not allowed to use any data structure other than simple arrays.

Later, we may reuse the same scenario and ask you to implement the same system using some data structures introduced during this course.

You are asked to design and implement basic social media software that provides sharing posts with people. The software should meet the following requirements:

- Each account has a profile which contains username, birth date, location, posts (if exists), follower accounts (if exists), following accounts (if exists).
- Each account can follow or be followed by another accounts. The number of following/follower accounts can vary from zero to many.
- In order to do the following actions, the user must log into their accounts.
    - Following another account
    - Viewing the profile of another account (in order to view the profile, there is no need to follow the account.)
    - Sharing a post.
    - Interacting with a post which belongs to another account.
    - Sending a message to another account (in order to send a message, the account must be followed.)
- The interaction with a post of another account can be as a like or a comment.
- The interaction information (number of likes/comments, the accounts that liked/commented, the content of the comments) of each post can be viewed by all users.

The basic design of the software is illustrated in Figure 1. Extra attributes may be needed to implement the functions in the classes. Declaring member variables in the classes as private, public, protected etc. will be decided by you.

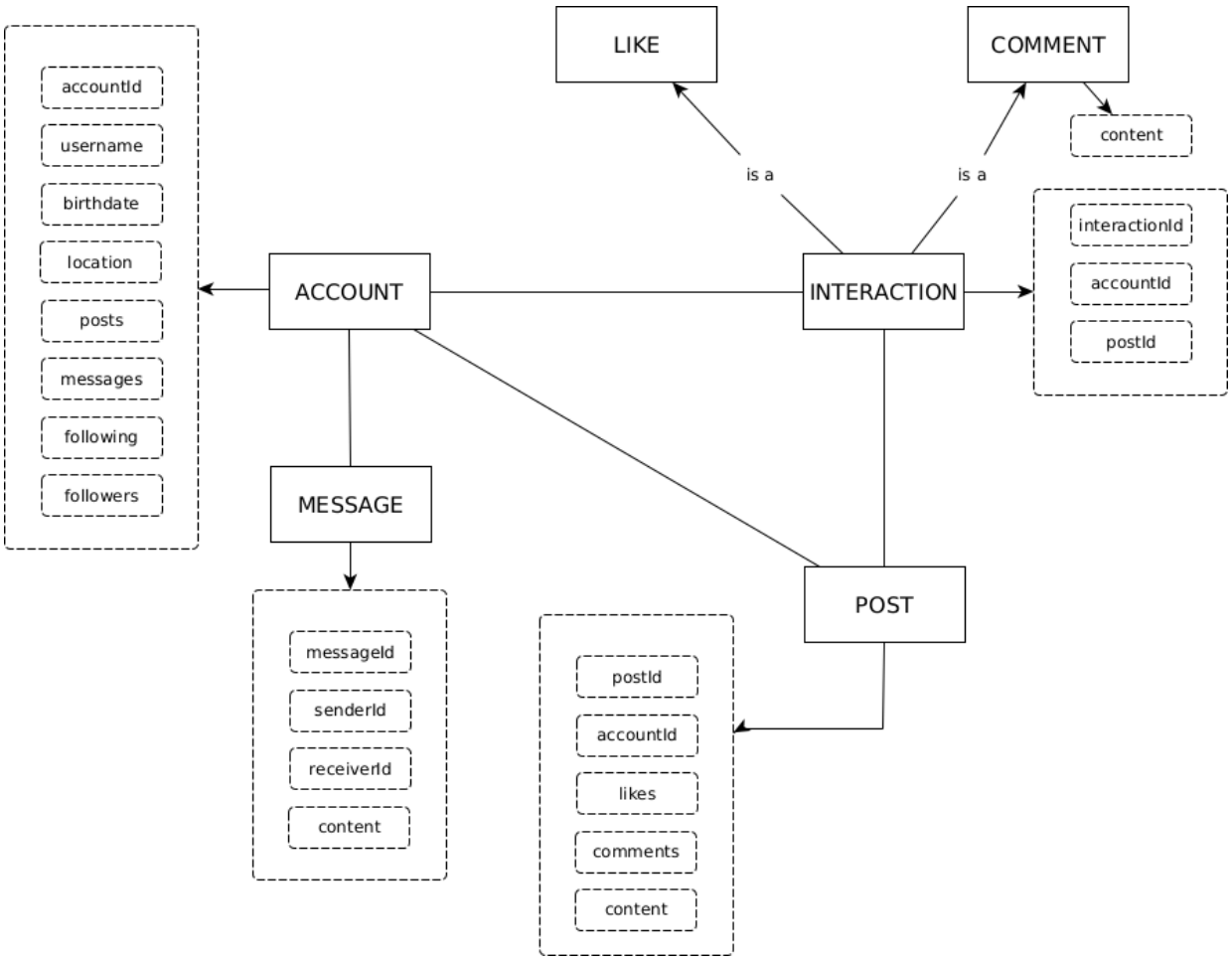

Figure 1. ER Diagram of The Software

The software will be tested in 3 scenarios. The first scenario is given with its test class code and expected output as follows. The second and the third scenarios will be written by you. You will use the test class code of the first scenario in order to implement the other scenarios.

SCENARIO 1

This is a sample scenario for you to test your coding.

- Step 1: Create 3 accounts with usernames “gizemsungu”, “sibelgulmez”, and “gokhankaya”.
- Step 2: Login to the account with username “sibelgulmez”.
    - Step 3: Share two posts.
    - Step 4: Follow “gizemsungu” and “gokhankaya”.
    - Step 5:Log out from the account.
- Step 6: Login to the account with username “gokhankaya”.
    - Step 7: View “sibelgulmez”s profile.
    - Step 8: View “sibelgulmez”s posts.
    - Step 9: Like a post of “sibelgulmez”.
    - Step 10: Comment on a post of “sibelgulmez”.

    - Step 11: Follow “sibelgulmez” and “gizemsungu”.
    - Step 12: Send a message to “gizemsungu”.
    - Step 13: Log out from the account.
- Step 14: Login to the account with username “gizemsungu”.
    - Step 15: Check the number of messages in the outbox.
    - Step 16: Check the number of messages in the inbox.
    - Step 17: View the messages in the inbox.
    - Step 18: View “sibelgulmez”s profile.
    - Step 19: View “sibelgulmez”s posts.
    - Step 20: View “sibelgulmez”s posts’ interactions.
    - Step 21: Like “sibelgulmez”s posts.
    - Step 22: View “sibelgulmez”s posts’ interactions.

The expected output format:

```
CSE222 – HW1

Step 1... Creating accounts...
An account with username gizemsungu has been created.
An account with username sibelgulmez has been created.
An account with username gokhankaya has been created.

Step 2... Logging into an account (username: sibelgulmez)…

Step 3... Sharing two posts…

Step 4... Following gizemsungu and gokhankaya...

Step 5... Logging out from account 'sibelgulmez'…

Step 6... Logging into another account (username: gokhankaya)…

Step 7... Viewing sibelgulmez's profile...
------------------------------
User ID: 2
Username: sibelgulmez
Location: Istanbul
Birth Date: 10.03.1995
sibelgulmez is following 2 account(s) and has 0 follower(s).
sibelgulmez is following: gizemsungu, gokhankaya,
sibelgulmez has 2 posts.

Step 8... Viewing sibelgulmez'posts...
(PostID: 1) sibelgulmez: I like Java.
(PostID: 2) sibelgulmez: Java the coffee...

Step 9... Liking a post of sibelgulmez...

Step 10... Adding a comment on a post of sibelgulmez...

Step 11... Following sibelgulmez and gizemsungu...

Step 12... Sending a message to gizemsungu...
```

```
Step 13... Logging out from account 'gokhankaya'..
.
Step 14... Logging into another account (username: gizemsungu)…

Step 15... Checking outbox...
There is/are 0 message(s) in the outbox.

Step 16... Checking inbox...
There is/are 1 message(s) in the inbox.

Step 17... Viewing inbox...
------------------------------
Message ID: 0
From: gokhankaya
To: gizemsungu
Message: This homework is too easy!

Step 18... Viewing sibelgulmez's profile...
------------------------------
User ID: 2
Username: sibelgulmez
Location: Istanbul
Birth Date: 10.03.1995
sibelgulmez is following 2 account(s) and has 1 follower(s).
The followers of sibelgulmez are: gokhankaya,
sibelgulmez is following: gizemsungu, gokhankaya,
sibelgulmez has 2 posts.

Step 19... Viewing sibelgulmez's posts...
(PostID: 1) sibelgulmez: I like Java.
(PostID: 2) sibelgulmez: Java the coffee...

Step 20... Viewing sibelgulmez's posts' interactions...
------------------------------
(PostID: 1): I like Java.
The post was liked by the following account(s): gokhankaya,
The post has 1 comment(s)...
Comment 1: 'gokhankaya' said 'me too!'
------------------------------
(PostID: 2): Java the coffee...
The post has no likes.
The post has no comments.

Step 21... Liking sibelgulmez's posts...

Step 22... Viewing sibelgulmez's posts' interactions...
------------------------------
(PostID: 1): I like Java.
The post was liked by the following account(s): gokhankaya, gizemsungu,
The post has 1 comment(s)...
Comment 1: 'gokhankaya' said 'me too!'
------------------------------
(PostID: 2): Java the coffee...
The post was liked by the following account(s): gizemsungu,
The post has no comments.
```

SCENARIO 2

After the actions in Scenario 1 are completed, your code is responsible for doing the following tasks including Scenario 1.

1. “gizemsungu” logs in.
    a. shares a post (let’s call it Post1).
    b. shares another post (let’s call it Post2).
    c. logs out.
2. “sibelgulmez” logs in.
    a. views the profile of “gizemsungu”.
    b. likes Post1.
    c. logs out.
3. “gokhankaya” logs in.
    a. views the profile of “gizemsungu”.
    b. comments on Post2 (ex: Nice!”).
    c. sends a message to “gizemsungu” (ex: “Hello!”)
    d. logs out.
4. “gizemsungu” logs in.
    a. views her own profile.
    b. reads the message from “gokhankaya”.

SCENARIO 3 (Bonus 20 points)

In this scenario, the account can block any accounts in the software. Remember that you will continue from Scenario 1 to implement the following tasks.

1. “gizemsungu” logs in.
    a. blocks “sibelgulmez”.
    b. logs out.
2. “sibelgulmez” logs in.
    a. tries to view the profile of “gizemsungu”.
    b. tries to send a message to “gizemsungu”.

RESTRICTIONS

- Use only Arrays, any other data type is prohibited.
- Use only one main class to drive the others from that superclass.
- Don’t use any other third-party library.

GENERAL RULES

- No late submissions are accepted.

TECHNICAL RULES

You must write a driver function that demonstrates all possible actions in your homework.

Implement clean code standards in your code:

- Classes, methods, and variables names must be meaningful and related to the functionality.
- Your functions and classes must be simple, general, reusable, and focus on one topic.
- Use standard java code name conventions.

REPORT RULES

- Add all javadoc documentations for classes, methods, variables …etc. All explanations must be meaningful and understandable.
- You should submit your homework code, Javadoc, and report to MS Teams in a "studentid_hw1.tar.gz" file.
- Use the given homework format including selected parts from the table below:
    - Detailed system requirements ✓
    - The Project use case diagrams (extra points) ✓
    - Problem solutions approach ✓
    - Test cases ✓
    - Running command and results ✓

GRADING:

- No OOP design: -100
- No method overriding: -95
- No error handling: -50
- No inheritance: -95
- No polymorphism: -95
- No javadoc documentation: -50
- No report: -90
- Disobey restrictions: -100
- Cheating: -200
- Your solution is evaluated over 120 as your performance.

Q & A:

- How to log in to an account?
    - Just type down the userID / username, no password needed.
- How to implement login account mechanism?
    - You can just keep track of the account which is currently logged in. You are not asked to design something complicated.
- Which errors should be considered?
    - After logging into an account, only that account can execute operations e.g. you cannot send a message *from* another account, but you can send messages *to* anyone.
    - You cannot view or send a message to an account that doesn't exist, you cannot like a post that doesn't exist etc.
    - If a username is already used, it cannot be used again to create a new account.
    - Different types of objects (like a post, message, interaction, account) might have the same ID but the same type of objects should have different ID numbers.
- Is following necessary to view a profile?
    - No.

- Is following necessary to send a message?
    - Yes.
- How to manage ID numbers?
    - It's up to you.
- Can I use additional variables in classes?
    - Other than loop counters, no, you won't need to.
- While viewing the posts of an account / viewing inbox or outbox, which entries should be printed if there are more than 1?
    - Print them all at once.
- How to manage two-sided operations like following and sending a message?
    - When an account follows another, two operations must be done. A following info should be added to one account, and a follower info should be added to the other account. Same goes for the message sending; the message should be added to the sender's outbox and then it should be added to the receiver's inbox. Do not use any trigger mechanism, make it simple.