ОПИС АБСТРАКТНОГО ТИПУ ДАНИХ

Основний абстрактний тип даних, який використовуватиметься в системі називається Profile\_Eval. Він буде займатись отриманням даних, їх збереженням та обробкою для повернення результату. Для виконання всіх цих задач всередині ADT розроблено необхідні функції. Взаємодія з користувачем також прописана в цьому  ADT. Для використання цього типу даних, необхідно буде використовувати лише одну з реалізованих в ньому функцій. Далі програма буде сама взаємодіяти з користувачем в циклі і відповідно до кожного вибору наступної дії, який робитиме користувач, запускати інші свої функції.

Змінні класу:

\_menu – рядок, в якому записане меню пронумерованих можливих майбутніх дій, з яких користувач має обрати те, що він хоче зробити. Вибір виконується шляхом введення номеру обраної дії.

\_look*\_*in – порожній об’єкт LinkedList, який створено для того, щоб зберігати масив імен, серед яких пізніше буде випадково обиратись переможець.

\_check\_likes – порожній об’єкт LinkedList, який створено для того, щоб зберігати масиви назв профілів, які вподобали конкретні пости, вказані користувачем. Така зв’язана структура необхідна для того, щоб потім перевіряти чи є в ній обраний переможець, якщо його там не буде, то треба буде повторно проводити пошук. Важливим є те, що дані, які знаходяться в кожному вузлу цієї структури, є об’єктом типу Array. Кожен масив містить в собі імена людей. Для того, щоб користувачу зарахувались вимоги по лайкам, він має бути в кожному з масивів зв’язаної структури, бо кожен масив показує список людей, які вподобали іншу картинку. А переможець має вподобати всі, що були вказані користувачем.

\_check\_comments – порожній об’єкт LinkedList, який створено для того, щоб зберігати масиви назв профілів, які прокоментували конкретні пости, вказані користувачем. Причина створення цієї зв’язаної структури є такою ж, як і в \_check\_likes. Єдина різниця полягає в тому, що імена в масивах цієї зв’язаної структури означають коментаторів.

\_check\_subscriptions - порожній об’єкт LinkedList, який створено для того, щоб зберігати масиви назв профілів, які підписані на конкретні пости, вказані користувачем. Причина створення цієї змінної та способи її використання такі ж самі, як і в \_check\_likes та \_check\_comments.

\_result – спочатку ця змінна отримує значення None, а в процесі роботи програми вона отримує значення рядка з ім’ям користувача, який переміг.

Функції класу:

\_\_init\_\_ - ця функція ініціалізує об’єкт Profile\_Eval. В цій функції створюються всі змінні класу. \_menu присвоюється значення рядка з пронумерованими варіантами дій.

[1] Set a place for searching a winner.

[2] Add a post to check comments.

[3] Add a post to check likes.

[4] Add a profile to check subscriptions

[5] Start searching.

Enter a number of action:

\_look\_in, \_check\_likes, \_check\_comments, \_check\_subscriptions – отримують значення порожнього об’єкту  LinkedList.

\_data\_collecting – в цій функції проходить основний цикл програми. Друкується список пропозицій і в відповідності до вибору користувача викликаються інші функції, розроблені в класі. Коли користувач хоче почати пошук переможця, то створюється цикл, в якому йде пошук випадкового користувача та його перевірка. Цей цикл зупиниться тільки тоді, коли буде знайдений переможець. Після визначення переможця викликається функція для повернення користувачу результату. Якщо команда, яку вказав користувач, не співпадає з жодною з запропонованих, то з’являється повідомлення про невірну команду і надається можливість ще раз визначити дію.

\_set\_look\_in – в цій функції змінна \_look\_in наповнюється даними. Користувачу надається вибір, які дані він хоче додати в змінну, з якої буде робитись вибір переможця. (лайки, коментарі, підписники). В залежності від вибору додаються дані. Якщо користувач вже встановив значення змінної \_loook\_in, то вдруге його встановити не вдасться. Функція напише повідомлення про те, що змінна вже встановлена і поверне -1.

\_add\_check\_subscriptions, \_add\_check\_likes, \_add\_check\_comments працюють за однаковим принципом.

Ці функції приймають аргумент, який вказує на зв’язану структуру, в яку будуть додаватись дані про підписників, лайки та коментаторів. Ця функція запитує в користувача логін та пароль до профіля, в якому треба уде шукати дані. В функціях, де треба буде знаходити дані відносно конкретного посту, також треба буде ввести URL-адресу цього посту. Функція стягує дані та зберігає їх. Далі створюється масив, в який додаються стягнуті дані. Пізніше цей масив додається в зв’язану структуру, яка була надана в якості аргументу. Якщо в процесі стягування даних виникає помилка, то на екран виводиться повідомлення про помилкові логін або пароль. В цій ситуації користувача повертає в функцію \_data\_collecting.

search\_process – в цій функції випадково обирається ціле число від нуля до довжини \_look\_in. Після цього змінній \_result надається значення \_look\_in з індексом випадково обраного числа.

\_check\_requirements – ця функція приймає в якості аргументу посилання на зв’язану структуру. Потім функція проходиться по кожному елементу зв’язаної структури та по кожному елементу масивів, які знаходяться у вузлах. Під час проходу перевіряється змінна result на присутність в кожному масиві зв’язаної структури. Якщо в якомусь масиві такий елемент не зустрівся, то змінній \_result присвоюється значення None, а якщо є, то змінна далі має значення обраного переможця.

\_return\_results – ця функція друкує обраного переможця для користувача й на цьому зупиняється робота системи