



Juarez Goldemberg Mariana Noelia - 108441
Vita Sanchez María Macarena - 107377
Corrector: Tomás Baldi



ALGOGRAM

TDA's IMPLEMENTADOS

TDA USUARIOS

Se implementó este TDA dado que cada usuario tenía unos determinados comportamientos, entre ellos:

- Postear
- Actualizar su feed
- Ver una próxima publicación.

TDA POSTEOS

Se implementó este TDA dado que cada posteo tiene un comportamiento a la hora de ser utilizado. Entre ellos:

- Dar a conocer su información (orden del usuario que lo publicó y su id)
- Sumar likes
- Obtener la cantidad de likes que posee
- Obtener la lista de personas que le dieron like
- Dar a conocer el contenido (el nombre del usuario que publicó, el contenido de la publicación y el ID de la misma)

DETALLE DE COMANDOS

LOGIN y LOGOUT:

Se utilizó un TDA HASH el cual contiene todos los usuarios posibles. Este TDA tiene la capacidad de permitir guardar y buscar usuarios con una complejidad $O(1)$.

PUBLICAR UN POST:

Para realizar la acción de publicar se utilizó un TDA HEAP, el cual permite a partir de una afinidad, encolar posteos en una complejidad de $O(\log(p))$, repitiendo este proceso la cantidad de usuarios correspondiente, obteniendo una complejidad final de $O(u \log(p))$.

VER EL PRÓXIMO POST DEL FEED:

A partir de la implementación del TDA HEAP en el punto anterior, se logra desencolar en $O(\log(p))$ el posteo siguiente más afín al usuario actual.

DAR LIKE A UN POST:

La utilización del TDA ABB para este comando permite guardar y buscar usuarios que dieron *like* con una complejidad de $O(\log(U_p))$ en cada uno.

MOSTRAR LIKES DE UN POST:

A partir de la implementación del TDA ABB del punto anterior, se aplicó un recorrido IN-ORDER lo que nos permitió mostrar los usuarios en orden alfabético en $O(U_p)$.