



Informe

Trabajo practico 2

AlgoGram



✚ Corrector: Martin Buchwald

✚ Alumnos: Fernando Buono – Francisco de Luca Andrea

✚ Mails: fbuono@fi.uba.ar – fdelucaa@fi.uba.ar

✚ Grupo: G36

Distribución del programa

Al momento de comenzar la distribución del programa desarrollado en el TP, decimos dividir el mismo en tres archivos .go. Estos archivos eran, el main, en el cual se leen los archivos de entrada y luego se ejecuta el programa, cuyas funciones principales se encuentran en funciones_app, y por otra parte, en el tercer archivo, funciones_aux, donde simplemente tenemos algunas funciones auxiliares utilizadas en el desarrollo del programa.

TDAs utilizados o creados

Por otro lado, vimos la necesidad de crear cuatro nuevos TDAs específicos para este programa, los cuales llamamos Post, ListaPosts, Usuario y ListaUsuarios, cuyas primitivas utilizamos para lograr la complejidad pedida para cada comando.

Teniendo en cuenta esto, ahora para poder lograr las complejidades pedidas, se nos ocurrió también utilizar dentro de los TDAs creados, también utilizar otros TDAs, los cuales fueron hash, heap y ABB, debido a que estas estructuras nos permiten obtener los resultados buscados. Entonces:

- Por un lado, utilizamos un heap, o cola de prioridad, ya que esta estructura permite obtener los datos, en este caso los posts, los cuales son mas relevantes para el usuario que se encuentra logeado en ese momento. Además, esta estructura es capaz de lograr esto de manera rápida, $O(\log p)$ siendo p la cantidad de posts publicados.
- Por otra parte, el uso de un ABB, o árbol de búsqueda binaria, fue debido a que para poder mostrar los likes de un post en orden alfabético, es decir, de manera ordenada, es más rápido tener una estructura la cual permite insertar datos de manera ordenada, antes que ordenar una estructura cada vez que se inserta un dato o se pide uno. Por esto último, el ABB es ideal para este tipo de situaciones, donde cada vez que un usuario le da like a un post su nombre queda ya ordenado, y el primer dato del ABB va a ser siempre el correspondiente de manera alfabética.
- Por último, en el caso del hash, lo utilizamos dos veces. En el primer caso, el de los usuarios, nos pareció necesario se capaces de encontrar de manera rápida si el usuario, no solo pertenecía a la lista de usuarios, sino también para poder lograr que el mismo realice publicaciones, o likee otras de la manera más rápida posible. Mientras que, en el segundo caso, el hash fue utilizado para guardar los distintos posts realizados por los usuarios, y poder así encontrar un post según su id de manera rápida.