

Resumen

POR HACER

Keywords: C, Grafos, Red Social, Simulación, Estructuras de Datos

Índice

1	Introducción	1
2	Objetivo Principal	1
2.1	Objetivos secundarios	1
3	Planteamiento del desarrollo del proyecto	1
4	Implementación	1
4.1	Estructura de directorios	2
5	Gestión del equipo de trabajo	2
5.1	Normas de codificación	2
5.2	Comunicación	2
5.3	División del trabajo	2
6	Posibles mejoras	3
7	Ejemplo de uso	3
8	Conclusiones	3
9	Referencias	3

1. Introducción

POR HACER

2. Objetivo Principal

El objetivo principal del proyecto es crear una simulación de una red social eficiente y rápida, combinando armónicamente los distintos tipos de estructuras de datos vistos durante el curso de Estructuras de Datos (grafos, listas, hash tables, colas de prioridad, etc.).

2.1. Objetivos secundarios

- Implementar algoritmos de búsqueda eficientes, priorizando el tiempo de respuesta
- Implementar buenas prácticas de codificación
- Reforzar habilidades de trabajo en equipo y coordinación de tareas entre pares

3. Planteamiento del desarrollo del proyecto

POR HACER

4. Implementación

La implementación del proyecto fue realizada en lenguaje C, utilizando librerías estándar. El código fue escrito de manera modular, o sea, las funciones fueron separadas en diferentes archivos según su funcionalidad. Esto permitió una mayor manera de organización y mantenimiento del código, así como una mejor comprensión del funcionamiento del programa.

4.1. Estructura de directorios

En el directorio principal se encuentran los archivos `Makefile`, con el cual es posible compilar el programa de manera automática, y el archivo `README.MD`, que contiene información sobre el proyecto y su funcionamiento. Además, se encuentra el archivo `subtopics`, que contiene una lista de subtópicos modificable, que se utilizará para el funcionamiento de la aplicación.

Dentro del directorio `src` se encuentran los archivos fuente del programa, que son los siguientes:

- `main.c`: Contiene el flujo principal del programa.
- `database.c`: Contiene las funciones para la gestión y guardado de la base de datos.
- `graph.c`: Contiene las funciones para gestionar los grafos y relaciones entre usuarios.
- `hash_table.c`: Contiene las funciones para gestionar tablas hash.
- `users.c`: Contiene las funciones para gestionar usuarios y publicaciones.
- `utilities.c`: Contiene las funciones de utilidad y testing para el programa.
- `heaps.c`: Contiene las funciones para gestionar las colas de prioridad.

Por otro lado, en el directorio `incs` se encuentran los archivos de cabeceras correspondiente a cada uno de los archivos fuente.

5. Gestión del equipo de trabajo

El equipo de trabajo constó de 5 personas, uno de ellos designado por el profesor como líder del grupo, con la responsabilidad de coordinar el trabajo y la gestión del proyecto. Debido a la cantidad de personas involucradas, fue fundamental establecer mecanismos de comunicación, coordinación y orden para asegurar la colaboración y la coordinación de los diferentes miembros del equipo. Algunos de estos mecanismos fueron:

5.1. Normas de codificación

Normas de codificación fueron establecidas para el proyecto, con el objetivo de garantizar la coherencia y la consistencia en la codificación. Estas normas incluían:

- **Constantes:** Para las constantes se utilizó la convención de `SCREAMING_SNAKE_CASE`, con el objetivo de diferenciarlas de las variables y funciones.
- **Variables:** Para las variables comunes se utilizó la convención `camelCase`.
- **Funciones:** Para las funciones se utilizó la convención `snake_case`, así diferenciándose de las variables y constantes.
- **Tipos de datos:** Para los tipos de datos creados con `typedef` se utilizó la convención `PascalCase`, con el objetivo de diferenciarlos de los tipos de datos nativos de C.
- **Llaves de apertura:** Se estableció la convención de que las llaves de apertura se escribieran en la misma línea que el código que las contiene.

```

1      #define MAX_CHAR 256 // Constante
2      typedef _User *User; // Tipo de dato
3      Graph newGraph; // Variable
4      void print_all_users(Graph graph); // Funcion
5

```

Código 1. Ejemplo de código, utilizando las normas de codificación

5.2. Comunicación

Un aspecto de suma importancia para el proyecto es la comunicación entre los miembros del equipo, ya que sin ella, no sería posible garantizar la colaboración y la coordinación en el desarrollo del proyecto. Debido al tamaño del equipo y la complejidad del proyecto esto fue un desafío, debido a las diferencias en horarios y responsabilidades académicas de cada miembro. Para resolver este problema fue esencial establecer un medio de comunicación remoto, que permitiera a cada uno comunicarse de manera rápida y en el lugar en que estuviese.

Se creó un chat en línea con cada integrante del equipo, este chat utilizándose para discutir sobre el desarrollo del proyecto y su organización, así como para compartir información y recursos.

Esto resultó exitoso, ya que las reuniones presenciales resultaron ser extremadamente difíciles de coordinar, la comunicación remota permitiendo a los miembros ver y discutir los detalles del proyecto en el momento del día en que estuviesen disponibles. Sin embargo, esto también implicó consecuencias, ya que dio paso a que fuera fácil que algún miembro del equipo "desapareciese"; sin embargo, estas situaciones fueron manejadas a través de la comunicación por privado con este para consultar su situación.

5.3. División del trabajo

Al inicio del proyecto se decidió establecer **objetivos a corto plazo** para cada miembro del equipo, según las estructuras de datos que se deseaban implementar en el proyecto. A cada uno se le asignaron según afinidad, capacidad y gusto propio.

Una vez las estructuras de datos hechas, se decidió crear una **lista de tareas** a realizar, cada miembro con al menos una tarea a realizar. Esta lista fue creada en base a las ideas discutidas por los integrantes del equipo, y se fue actualizando constantemente en función de las necesidades del proyecto.

6. Posibles mejoras

POR HACER

7. Ejemplo de uso

POR HACER

8. Conclusiones

POR HACER

9. Referencias