



Informe Taller 02 Estructuras lineales

Facultad de Ingeniería

Estructuras de datos

John Corredor

David Estevan Rodríguez Jurado

Julian Perez

Juan Sebastian Rodriguez

Daniel Galvis Jaramillo

23 de septiembre de 2024

Implementación

La implementación del programa está diseñada para facilitar la lectura y manipulación de un archivo de texto de manera estructurada. Se decidió utilizar un **vector<std::list<Palabra>>** para almacenar cada línea del archivo, dividiendo el contenido en palabras y asociándolas con su número de línea. Esto permite que las búsquedas de palabras sean más específicas y eficientes, ya que cada palabra se encuentra organizada según la línea en la que aparece.

La clase **Palabra** encapsula tanto la palabra en sí como el número de línea en el que se encuentra, lo que le otorga una mayor modularidad al manejo del texto. La clase **ArchivoTexto** se encarga de todas las operaciones relacionadas con las palabras del archivo, como las búsquedas, garantizando que el manejo del texto esté separado de la lógica principal del programa.

Para la separación de responsabilidades, el **main()** se limita a la lógica de ejecución principal, mientras que las operaciones complejas, como las búsquedas, están encapsuladas en funciones dentro de las clases. Esto favorece la mantenibilidad del código, que es legible y fácil de modificar si es necesario.

Guía de compilación y ejecución

Para compilar el programa usando g++, sigue estos pasos:

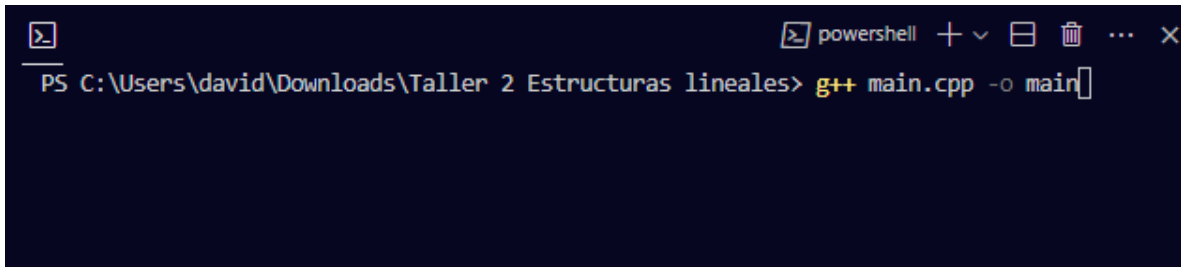
Compilar el código: Abre una terminal y ejecuta el siguiente comando para compilar el archivo **main.cpp** junto con el archivo **Clases.h** “**g++ main.cpp -o main**”.

Ejecutar el programa: Una vez compilado, puedes ejecutar el programa pasándole el archivo de texto como parámetro “**./main archivo.txt**”.

Asegúrate de que **archivo.txt** esté en el mismo directorio que el ejecutable o proporciona la ruta completa al archivo.

Compilación y ejecución

Para la compilación con g++ utilizamos el commando “**g++ main.cpp -o main**” y esperamos a que se compile el programa.

A screenshot of a PowerShell terminal window. The title bar shows 'powershell' and standard window controls. The command prompt shows the path 'C:\Users\david\Downloads\Taller 2 Estructuras lineales' followed by the command 'g++ main.cpp -o main' with a cursor at the end.

```
PS C:\Users\david\Downloads\Taller 2 Estructuras lineales> g++ main.cpp -o main
```

Figura 1. Comando para compilación.

Para comprobar que si se compiló y creó el ejecutable revisaremos el directorio y buscaremos el ejecutable con el nombre que creamos.



Figura 2. Buscar ejecutable.

Para ejecutar nuestro programa escribiremos el commando “**./main entrada1.txt**”, ya que haremos la prueba con entrada1.txt que esta en nuestro directorio, recordando que la estructura del archive que se da como paranmetro tiene que tener como primera linea el numero de lineas del archive de texto sin contarse a ella misma y la segunda linea es la subcadena con la que se hara el proceso de busqueda de palabras en el archivo de texto.

```
≡ entrada1.txt
1  5
2  co
3  solo con porro pongo loco
4  loco monto moto, corro, todo borro
5  borro todo tonto, no compongo, solo compro
6  yo, mono loco, no conozco otro modo!
```

Figura 3. Captura del archivo “entrada1.txt”.

```
PS C:\Users\david\Downloads\Taller 2 Estructuras lineales> ./main entrada1.txt

Hay 3 palabras que empiecen con: co
línea 4: corro,
línea 5: compongo,
línea 5: compro

Hay 5 palabras que contienen: co
línea 3: loco
línea 4: loco
línea 4: corro,
línea 5: compongo,
línea 5: compro

Hay 2 palabras que contienen: oc
línea 3: loco
línea 4: loco
```

Figura 4. Captura de la ejecución del programa.

Si hacemos este proceso a mano veremos que nos dara exactamente el mismo resultado, siempre y cuando pasemos como parametro desde la terminal un archivo de texto valido para el programa.