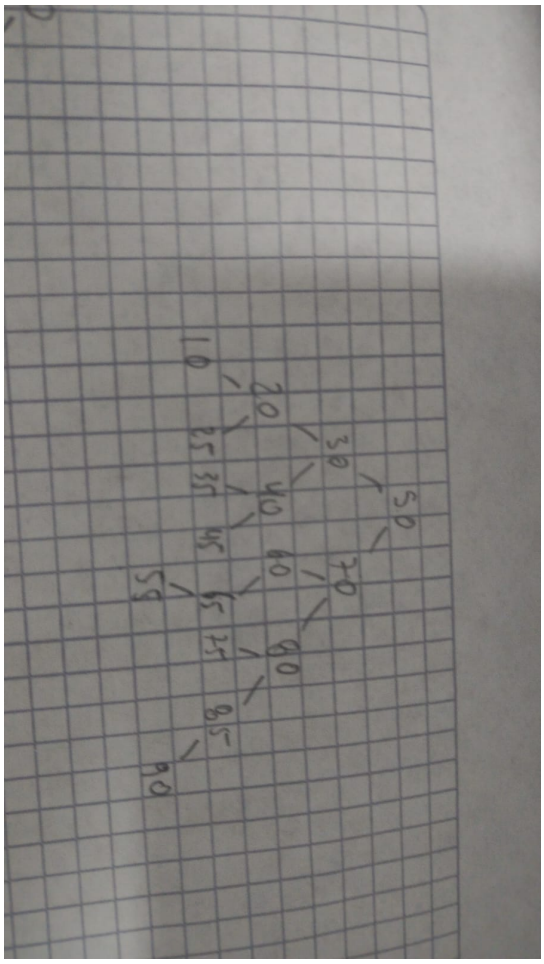
Taller-5: Construcción y Manejo de Árboles AVL y Árboles B

*Javier Eduardo Barreto Rojas Cod: 506231722 Estudiante: Nicolas Saavedra Arciniegas Cod: 506221076

I. Objetivo del taller

Comprender y aplicar los principios de inserción, balanceo y eliminación en árboles AVL.



II. Parte 1: Árboles Binarios de Búsqueda Auto-Balanceables (AVL)

II-A. ***Objetivo:*** *Comprender y aplicar los principios de inserción, balanceo y eliminación en árboles AVL.*

II-B. ***Instrucciones:*** *Para cada uno de los siguientes conjuntos de números, realiza las siguientes tareas:*

III. Conjuntos Dados:

Conjunto de números 1: 50, 30, 70, 20, 40, 60, 80, 10, 25, 35, 45, 55, 65, 75, 85, 95

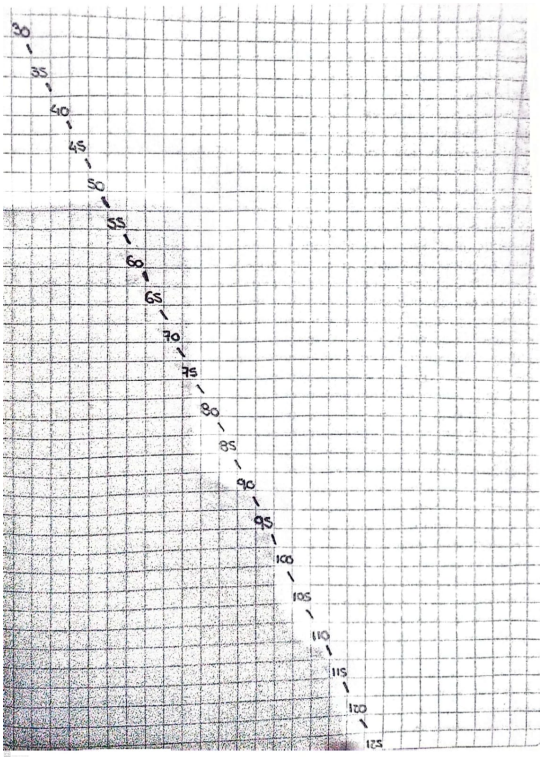
Conjunto de números 2: 30, 35, 40, 45, 50, 55, 60, 65, 70, 75, 80, 85, 90, 95, 100, 105, 110, 115, 120, 125

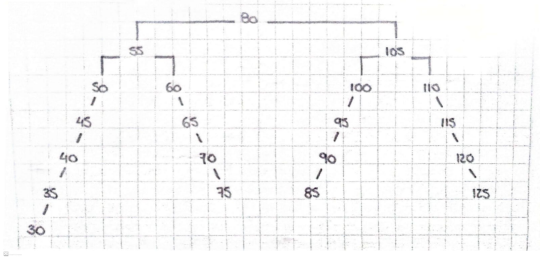
Conjunto de números 3: 40, 55, 65, 35, 70, 25, 15, 75, 30, 10, 45, 85, 60, 20, 50, 80, 90, 95

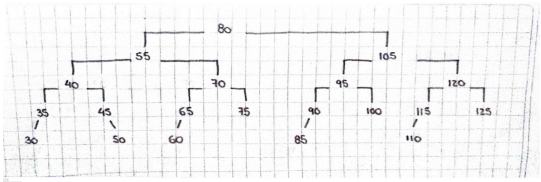
III-A. ***Tareas***

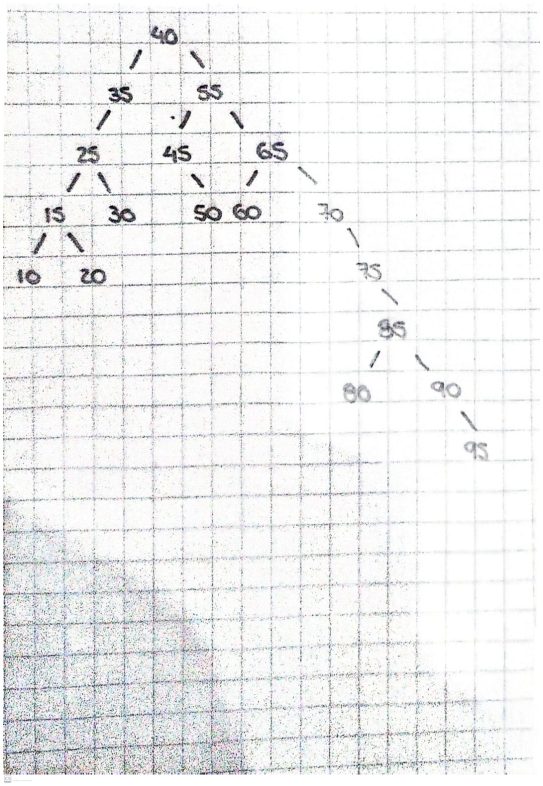
Inserción y Balanceo:

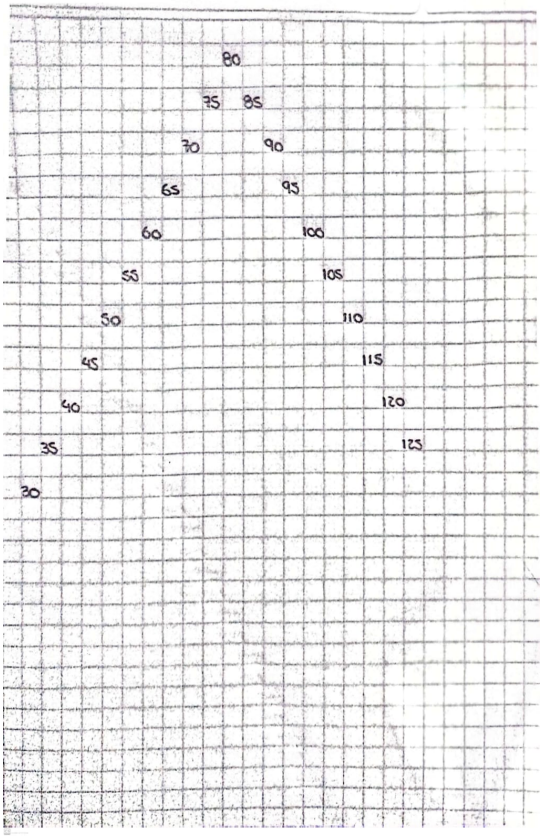
Inserta los números uno por uno en un árbol AVL. Dibuja el árbol después de cada inserción. Verifica si el árbol está balanceado después de cada inserción. Si no lo está, aplica las rotaciones necesarias para balancearlo y documenta el proceso. Explica brevemente por qué cada rotación fue necesaria.

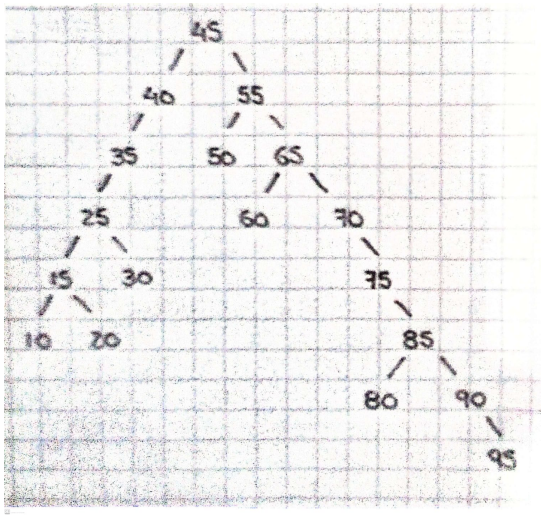

30
35
40
45
50
55
60
65
70
75
80
85
90
95
100
105
110
115
120
125

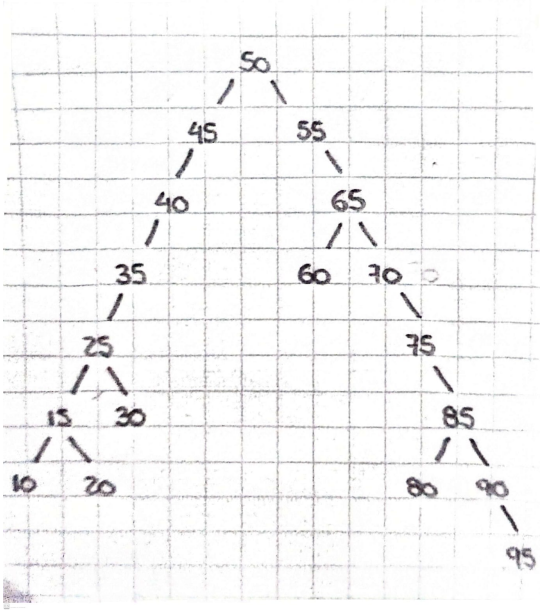

80
55
105
50
60
100
110
45
65
95
115
40
70
90
120
35
75
85
125
30

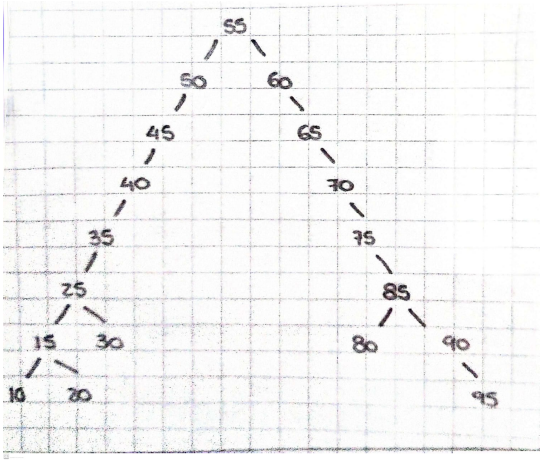

80
55
105
40
70
95
120
35
45
65
75
90
100
115
125
30
50
60
85
110


40
35
55
25
45
65
15
30
50
60
70
10
20
75
95
80
90
95


80
75
85
70
90
65
95
60
100
55
105
50
110
45
115
40
120
35
125
30


45
40
55
35
50
65
25
60
70
15
30
75
10
20
85
80
90
95





IV. Parte 2: Árboles B

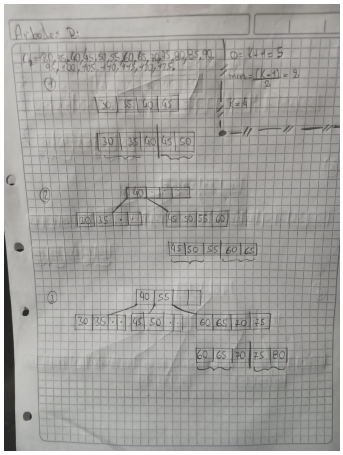
Objetivo Desarrollar habilidades en la construcción y manejo de árboles B, incluyendo inserciones, divisiones y eliminaciones.

Instrucciones Utiliza el **Conjunto de números 2 y 3** para construir árboles B, siguiendo estas pautas:

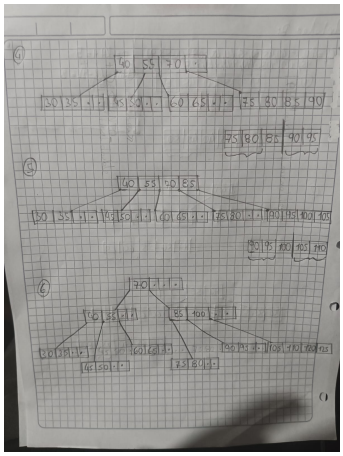
Construcción de Árboles B y Eliminacion de Claves:

V. Entregables

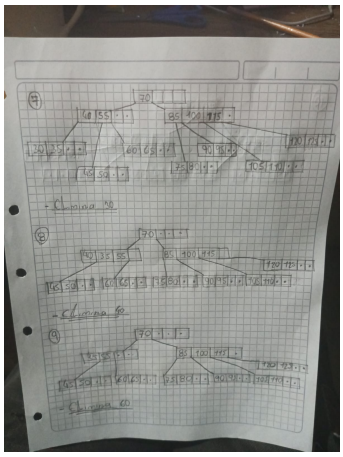
VI. Arbol K=4 - Conjunto 2



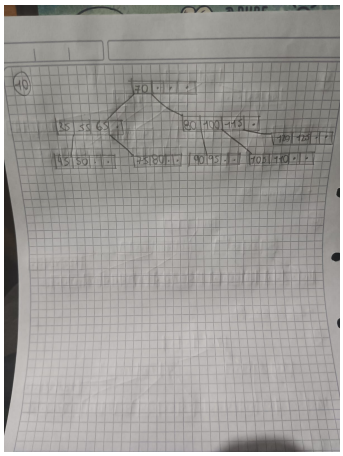
En este fragmento del primer arbol se hallo la raiz, como su cantidad maxima de hojas en la hoja mas inferior de este y asi mismo tambien la cantidad maxima de elementos en estas hojas.



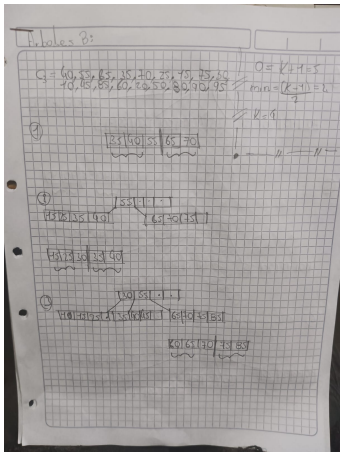
Luego en este fragmento del arbol se realiza la agregacion de los elemntos 95, 105 y 125, para continuar ordenando el arbol de forma que este se mantenga balanceado para poder continuar con el proceso de agregacion.



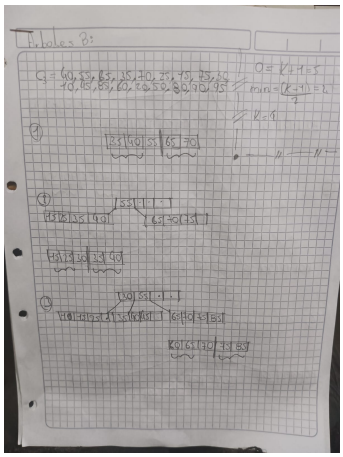
Aqui, igualmente que en el paso anterior se realizan las validaciones correspondientes para continuar con el proceso de agregacion de elementos de la cadena al arbol. Finalmente se procede a eliminar los elementos 30, 40 y 60.



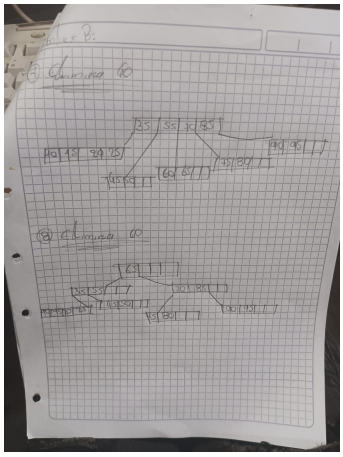
VII. Arbol K=4 - Conjunto 3



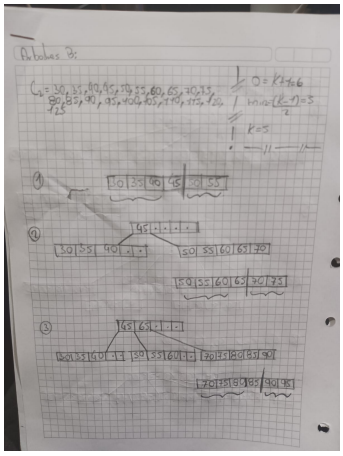
En este fragmento del primer arbol se hallo la raiz, como su cantidad maxima de hojas en la hoja mas inferior de este y asi mismo tambien la cantidad maxima de elementos en estas hojas.



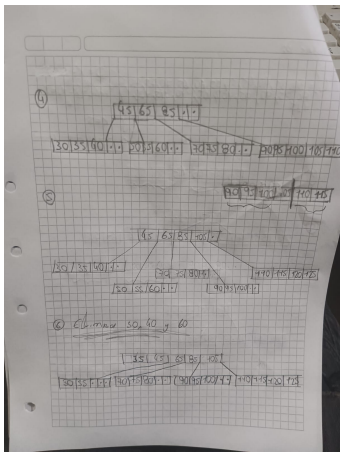
Luego en este fragmento del arbol se realiza la agregacion y ordenacion de algunos elementos, luego se ordena el arbol y finalmente se eliminan los elementos 30, 40 y 60, en cada una de estas eliminaciones validando que el arbol quede ordenado en cada inciso.



VIII. Arbol K=5 - Conjunto 2



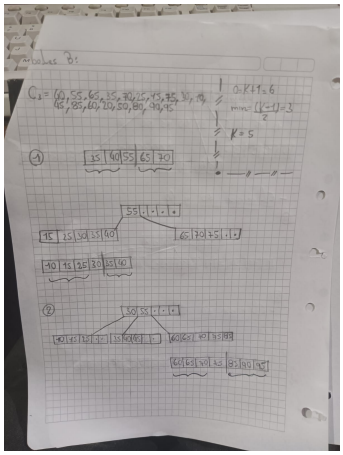
En este fragmento del primer arbol se hallo la raiz, como su cantidad maxima de hojas en la hoja mas inferior de este y asi mismo tambien la cantidad maxima de elementos en estas hojas.



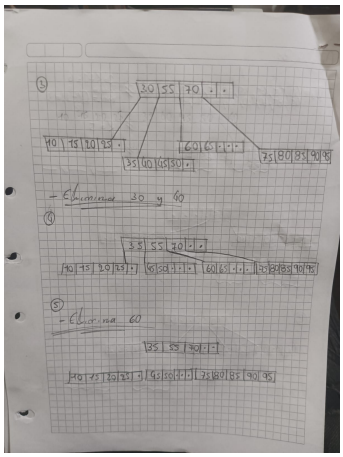
En esta parte se realizan algunas agregaciones de elementos al arbol de manera que, continue balanceado en su estructura provocando algunos descuelgues de datos para poder realizar

este balance de una forma mas efectiva, finalmente se realiza un proceso de eliminacion mediante el cual se eliminan los valores correspondientes al 30, 40 y 60, evitando cualquier desbalance en la estructura del arbol.

IX. Arbol K=5 - Conjunto 3



En este fragmento del primer arbol se hallo la raiz, como su cantidad maxima de hojas en la hoja mas inferior de este y asi mismo tambien la cantidad maxima de elementos en estas hojas.



Mediante este fragmento se muestra como en el arbol se realiza un proceso de ordenamiento del mismo, luego se procede a eliminar los valor 30, 40 y 60, manteniendo de esta forma su balance

Recorridos del Árbol:

Realiza recorridos en preorden, inorden y postorden sobre los árboles AVL construidos con el conjunto de números 2 y 3. **(Documenta el resultado de cada recorrido.)**

Pre-Order:

- **Conjunto 1:** [50,30,20,10,25,40,35,45,70,60,65,55,80, 75,85,90].
- **Conjunto 2:** [80,55,40,35,30,45,50,70,65,60,75,105,95, 90,85,100,120,115,110,125].
- **Conjunto 3:** [50,45,40,35,25,15,10,20,30,55,65,60,70, 75,85,80,90,95].

In-Order:

- **Conjunto 1** [10,20,25,30,35,40,45,50,60,55,65,70,75, 80,85,90].
- **Conjunto 2:** [30,35,40,45,50,55,60,65,70,75,80,85,90, 95,100,105,110,115,120,125].
- **Conjunto 3:** [10,15,20,25,30,35,40,45,50,55,60,65,70, 75,80,85,90,95].

Post-Order:

- **Conjunto 1:** [10,25,20,35,45,40,30,55,65,60,75,90,85, 80,70,50].
- **Conjunto 2:** [30,35,50,45,40,60,65,75,70,55,85,90,100, 95,110,115,125,120,105,80].
- **Conjunto 3:** [10,20,15,30,25,35,40,45,60,80,95,90,85, 75,70,65,55,50].

X. Entregables

Diferencias Árboles Discusión sobre las diferencias observadas en el comportamiento entre los árboles AVL y los árboles B con diferentes órdenes.

Árboles-AVL:

- Un árbol AVL es un tipo de árbol binario de búsqueda autoequilibrado.
- Cada nodo del árbol tiene un factor de equilibrio, que es la diferencia entre la altura del subárbol izquierdo y la altura del subárbol derecho.
- El factor de equilibrio se utiliza para mantener el árbol balanceado, lo que garantiza que las operaciones de búsqueda, inserción y eliminación se realicen de manera eficiente.
- Las operaciones de inserción y eliminación pueden provocar desequilibrios en el árbol, por lo que se realizan rotaciones para restaurar el equilibrio.
- La altura del árbol AVL es relativamente baja, lo que reduce el tiempo de búsqueda.

Árboles-B:

- Un árbol B es un tipo de árbol de búsqueda autoequilibrado que utiliza un conjunto de nodos que contienen claves y punteros a otros nodos.
- Cada nodo del árbol B tiene un número fijo de claves y punteros a otros nodos.
- La clave más pequeña de un nodo se encuentra en la raíz del subárbol izquierdo, y la clave más grande se encuentra en la raíz del subárbol derecho.
- Los árboles B se utilizan comúnmente en bases de datos y sistemas de archivos para almacenar grandes cantidades de datos.
- La altura del árbol B es relativamente baja, lo que reduce el tiempo de búsqueda.

XI. Diferencias observadas en el comportamiento entre los árboles AVL y los árboles B con diferentes órdenes

Orden: Los árboles AVL tienen un orden fijo de 2, lo que significa que cada nodo tiene un máximo de 2 hijos. Los árboles B, por otro lado, tienen un orden variable, lo que significa que cada nodo puede tener un número variable de hijos.

Balanceo: Los árboles AVL se balancean mediante rotaciones, mientras que los árboles B se balancean mediante la reorganización de los nodos y la redistribución de las claves.

Altura: La altura de los árboles AVL es generalmente más baja que la de los árboles B, especialmente para conjuntos de datos grandes.

Inserción y eliminación: La inserción y eliminación de nodos en árboles AVL es más compleja que en árboles B, ya que requiere rotaciones y ajustes en el factor de equilibrio.

Características

Caracteristicas:	Arboles AVL:	Arboles B:
Orden	Fijo{2}	Variable
Balanceo	Rotaciones	Reorganización de nodos y redistribución de claves
Altura	Relativamente baja	Relativamente alta
Inserción y eliminación	Compleja	Simple
Uso	Búsqueda y ordenamiento	Almacenamiento de grandes cantidades de datos

Similitudes

- Ambos tipos de árboles están diseñados para optimizar el tiempo de búsqueda, inserción y eliminación.
- Ambos tipos de árboles utilizan algoritmos de balanceo para mantener un buen rendimiento.
- Ambos tipos de árboles son recursivos en su estructura y algoritmos.

Diferencias

- Los árboles AVL son binarios, mientras que los árboles B pueden tener múltiples claves y punteros por nodo.
- Los árboles AVL se reequilibran después de cada operación, mientras que los árboles B solo se reequilibran después de inserciones o eliminaciones.
- Los árboles AVL tienen una altura logarítmica fija, mientras que la altura de los árboles B depende del orden del árbol.
- Los árbolárboles AVL son más adecuados para aplicaciones en memoria, mientras que los árboles B son más adecuados para aplicaciones en disco.

Referencias

[1] *Taller 5*, GitLab. [En línea]. Disponible en: https://gitlab.com/konrad_lorenz/data-structures-2024-1/taller-5. [Consultado: 19-may-2024].