

Práctica 02

DOCENTE	CARRERA	CURSO
Vicente Machaca Arceda	Maestría en Ciencia de la Computación	Algoritmos y Estructura de Datos

PRÁCTICA	TEMA	DURACIÓN
02	Estructura de datos	–

1. Datos de los integrantes

- Grupo: **2**
- Integrantes:
 - EDER ALONSO AMPUERO ATAMARI
 - HOWARD FERNANDO ARANZAMENDI MORALES
 - JOSE EDISON PEREZ MAMANI
 - HENRRY IVAN ARIAS MAMANI

2. Repositorio de GITHUB

Url Github: Práctica 2

3. Estructuras de Datos

3.1. BTree

ZZZZZZZZZZ

3.2. AVL

El árbol AVL recibe su nombre de las iniciales de sus inventores, Georgii Adelson-Velskii y Yevgeniy Landis. Dieron a conocer esto mediante la publicación de un artículo en 1962, “Un algoritmo para organizar la información” (Un algoritmo para organizar la información”).

Un árbol AVL es un árbol de búsqueda binaria que tiene una altura equilibrada: para cada nodo x , las alturas de los subárboles izquierdo y derecho de x difieren en 1 como máximo. Para implementar un árbol AVL, mantenemos un atributo adicional en cada nodo: $x.h$ es la altura del nodo x . Como para cualquier otro árbol binario de búsqueda T , asumimos que $T.root$ apunta al nodo raíz.

El árbol AVL siempre está equilibrado de modo que para todos los nodos, la altura de la rama izquierda no difiera en más de una unidad de la altura de la rama derecha o viceversa. Gracias a esta forma de equilibrio, la complejidad de una búsqueda en uno de estos árboles se mantiene siempre en el orden de complejidad $O(\log n)$. El factor de equilibrio puede almacenarse directamente en cada nodo o calcularse a partir de la altura de los subárboles.

Para lograr este equilibrio, la inserción y eliminación de nodos debe realizarse de manera especial. Si la condición de equilibrio se rompe al realizar una operación de inserción o eliminación, se debe realizar una serie de rotaciones de nodos.

3.2.1. Resultados del experimento

- En la figura 1 mostramos nuestro árbol inicial con 11 nodos.
- En la figura 1 mostramos nuestro árbol inicial con 11 nodos.
- En la figura 1 mostramos nuestro árbol inicial con 11 nodos.
- En la figura 3 mostramos nuestro árbol inicial con 11 nodos.
- En la figura 4 mostramos el árbol luego de insertar un nodo nuevo.
- En la figura 5 mostramos luego de eliminar un nodo.
- En la figura 6 mostramos luego de eliminar el nodo raíz.

```
Buscando nodo 22
No encontrado
Buscando nodo 21
Se encontro el nodo
Resultado: 21
valor Mínimo: 6
valor Máximo: 63
```

Figura 1: Árbol AVL búsqueda variada

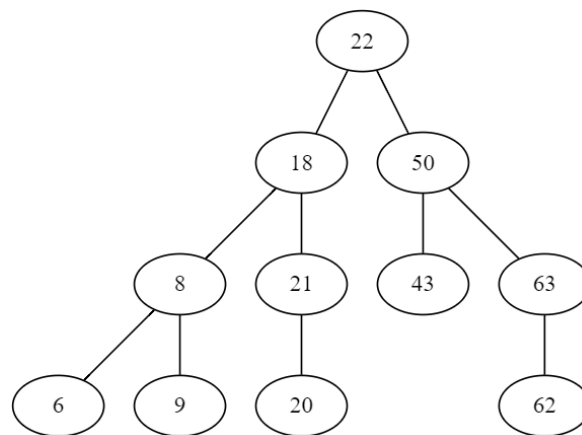


Figura 2: Árbol AVL inicial

4. Conclusiones

- SSSS
- SSSS
- Un árbol AVL se mantiene ordenado, pero hay mas rotaciones en las inserciones que en las eliminaciones, su costo para buscar, eliminar, insertar es de $O(\log n)$.
-

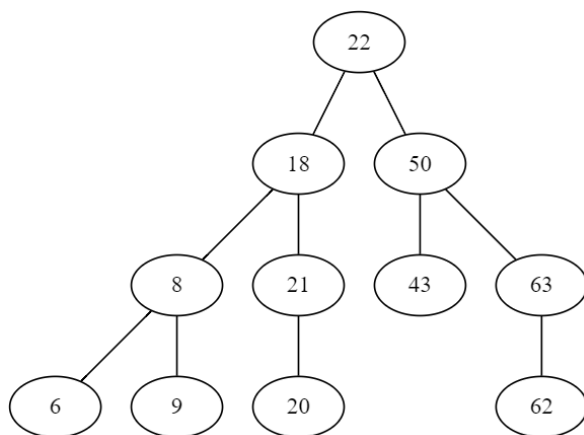


Figura 3: Árbol AVL inicial

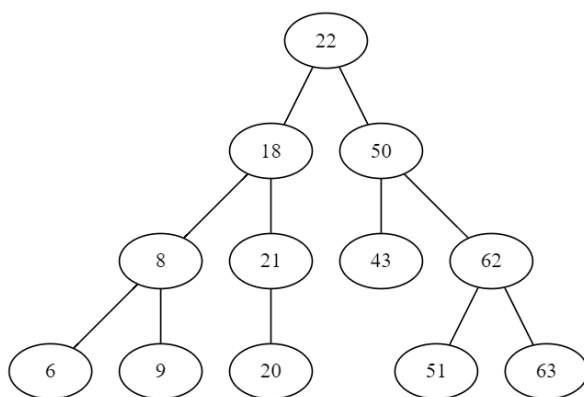


Figura 4: Árbol AVL con inserción

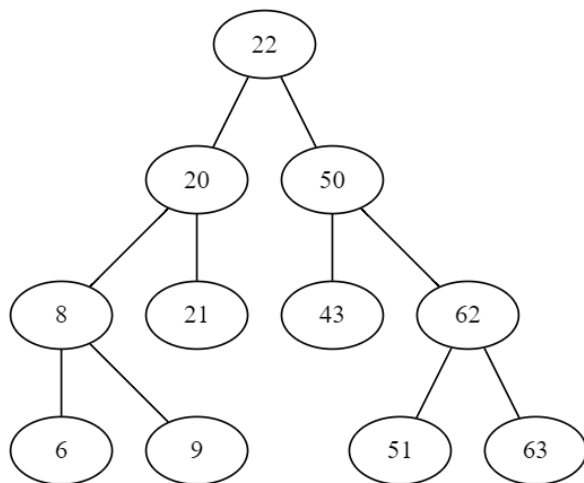


Figura 5: Árbol AVL con eliminación nodo

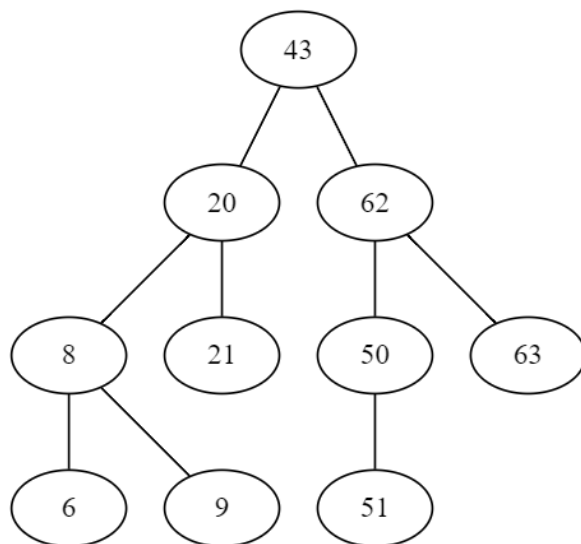


Figura 6: Árbol AVL con eliminación de la raíz

5. Referencias

1. 1
2. 1
3. 1
4. 1
5. 1
6. 1
7. 1
8. 1
9. 1
10. 1