

# Estructuras de Datos 2022-1

## Práctica 5: Árboles AVL

Pedro Ulises Cervantes González  
confundeme@ciencias.unam.mx

Emmanuel Cruz Hernández  
emmanuel\_cruzh@ciencias.unam.mx

Yessica Janeth Pablo Martínez  
yessica\_j\_pablo@ciencias.unam.mx

América Montserrat García Coronado  
ame\_coronado@ciencias.unam.mx

Adrián Felipe Vélez Rivera  
adrianf\_velez@ciencias.unam.mx

Fecha límite de entrega: 15 de diciembre de 2021  
Hora límite de entrega: 23:59:59 hrs

### 1. Objetivo

Implementar y conocer el funcionamiento interno de la estructura de datos conocida como Árbol AVL.

Esta estructura se basa en un orden jerárquico y necesita un conjunto de elementos comparables para darle un lugar específico a cada elemento dentro de un árbol binario. Además mantiene la propiedad de ser ordenado, tomando la altura como base para lograrlo.

### 2. Actividad

Implementa las operaciones de la interfaz *TDABinarySearchTree* en una clase llamada *AVLTree*. El objetivo es programar el comportamiento de un árbol AVL.

#### 2.1. Retrieve (0.5 puntos)

Implementa el método `retrieve(K)`, que dada una clave  $k$ , regresa el elemento asociado a dicha clave o null si no existe.

#### 2.2. Insert 0.5 puntos)

Implementa el método `insert(T, K)`, que dada una clave  $k$  y un elemento  $t$ , inserta el elemento  $t$  con clave  $k$  en el árbol, considerando que en un nodo  $p$  con elemento  $e(p)$ , tiene en el subárbol izquierdo a todos los elementos cuya clave es menor a  $e(p)$  y como subárbol derecho a todos los elementos con clave mayor a  $e(p)$ .

Una vez insertado, este nodo debe rebalancer y actualizar las alturas a partir del nodo insertado hasta la raíz del árbol.

#### 2.3. Delete (0.5 puntos)

Implementa el método `delete(K)`, que dada una clave  $k$  elimina su elemento asociado en el árbol, además de regresar el elemento. En caso de no existir, se regresan null.

Una vez eliminado, este nodo debe rebalancer y actualizar las alturas a partir del nodo que toma el lugar del eliminado hasta la raíz del árbol.

## 2.4. FindMin (0.5 puntos)

Implementa el método `findMin()` que regresa el elemento con clave mínima en el árbol o `null` si el árbol es vacío.

## 2.5. FindMax (0.5 puntos)

Implementa el método `findMax()` que regresa el elemento con clave máxima en el árbol o `null` si el árbol es vacío.

## 2.6. isEmpty (0.5 puntos)

Implementa el método `isEmpty()` que regresa *true* si el árbol es vacío y *false* en otro caso.

## 2.7. rebalancea (6 puntos)

Implementa un método que permita rebalancear el árbol a partir de un nodo *v* hasta la raíz del árbol, que además actualice la altura de los nodos involucrados en el rebalanceo.

## 2.8. Recorridos (0.5 puntos)

Implementa los recorridos siguientes sobre el árbol AVL.

- Preorden
- Inorden
- Postorden

Visitar en este caso se tomará como imprimir el elemento actual del árbol.

## 2.9. Main (0.5 puntos)

Crea un método *main* en la clase **AVLTree** con un menú que permita hacer todas las operaciones definidas, así como salir del mismo.

## 3. Extra (1 Punto)

Implementa las operaciones de la interfaz *TDABinarySearchTree* en una clase llamada *RedBlackTree*. El objetivo es programar el comportamiento de un árbol rojinegro. Para la evaluación del punto, se considerarán los métodos siguientes

- insert (0.5 puntos)
- delete (0.5 puntos)

## 4. Recursos de apoyo

- Presentación:

[https://docs.google.com/presentation/d/1n3zvm0yqUTIcNYxqnc7iIpcLZ-qqV\\_eBTdZas-o8Cr8/edit?usp=sharing](https://docs.google.com/presentation/d/1n3zvm0yqUTIcNYxqnc7iIpcLZ-qqV_eBTdZas-o8Cr8/edit?usp=sharing)

## 5. Reglas Importantes

- No se recibirán prácticas en las que estén involucrados más de dos integrantes.
- Cumple con los lineamientos de entrega.
- Todos los archivos deberán contener nombre y número de cuenta.
- Tu código debe estar comentado. Esto abarca clases, atributos, métodos y comentarios extra.
- Utiliza correctamente las convenciones para nombrar variables, constantes, clases y métodos.
- No se pueden usar implementaciones de estructuras de datos ya hechas por Java. En caso de usarse, se evaluará con 0 la actividad.
- En caso de no cumplirse alguna de las reglas especificadas, se restará 0.5 puntos en tu calificación obtenida.

