



EXAMEN FINAL: ESTRUCTURAS DE DATOS

Solorzano Rodas, Julio Andres

UDV ED1



Explicación de la estructura de clases (Node , Mostrar inserciones paso a paso. AVLTree , Main)

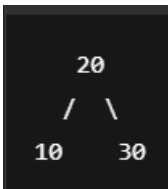
Breve descripción del problema resuelto.

Instrucciones para ejecutar el proyecto

Explicación del Proyecto AVL en Java

Breve descripción del problema resuelto

Este programa permite construir un **Árbol AVL dinámico** desde consola, insertando números uno por uno. Cada vez que se inserta un número, el árbol se **actualiza automáticamente y se balancea**, aplicando rotaciones simples o dobles si es necesario. Luego, se imprime la estructura del árbol en formato gráfico tipo:



Estructura de Clases

◆ Node.java

Clase que representa un nodo del árbol.

- **Atributos:**
 - int valor: el número guardado.
 - Node izquierda: subárbol izquierdo.
 - Node derecha: subárbol derecho.
 - int altura: altura del nodo para calcular balance.

◆ AVLTree.java

Clase que contiene toda la lógica del Árbol AVL.

- **Atributos:**
 - Node raiz: raíz del árbol.
- **Métodos principales:**

- insertar(Node nodo, int valor): inserta un nodo como BST, luego recalcula balance y aplica rotaciones si es necesario.
- getAltura(Node nodo): retorna la altura del nodo.
- getFactorBalance(Node nodo): calcula el factor de balance de un nodo.
- rotarIzquierda(Node x) y rotarDerecha(Node y): realizan rotaciones simples para balancear.
- printTree(Node raiz): imprime gráficamente el árbol en consola usando espacios y ramas / \.

◆ Main.java

Clase principal que ejecuta el programa.

- Solicita números desde consola.
- Inserta cada número en el árbol.
- Imprime el árbol actualizado.
- Finaliza si el usuario escribe "exit" o -1.

✂ Instrucciones para ejecutar el proyecto

Requisitos

- Java instalado (Java 8 o superior)
- Editor de código o IDE (como IntelliJ, VS Code o NetBeans)
- Consola o terminal

Pasos para correr el programa

1. Guarda los archivos en una carpeta:
 - Node.java
 - AVLTree.java
 - Main.java
2. Abre una terminal en esa carpeta.
3. Compila los archivos:

bash

```
javac *.java
```

Ejecuta el programa: `java Main`

- ❓ En consola, escribe un número entero y presiona Enter.
- ❓ El árbol AVL se imprimirá automáticamente. Escribe más números para ver cómo se balancea.
- ❓ Para salir del programa, escribe "exit" o -1.

Anexos

Capturas de:

Entradas (lista de números insertados).

Ejemplos:

Secuencia de entrada: [10, 20, 30]

Secuencia de entrada: [50, 30, 70, 20, 40, 60, 80, 35]

Salida (dibujo del árbol AVL final).

Ejemplo gráfico del árbol AVL generado.

```
Inserta números enteros al árbol AVL (escribe 'exit' o -1 para terminar):
Número: 10
Árbol AVL actualizado:
10

Número: 20
Árbol AVL actualizado:
  10
   \
    20

Número: 30
Árbol AVL actualizado:
    20
   / \
  10  30

Número: -1
Programa terminado.
```

Número: 50
Árbol AVL actualizado:
50

Número: 30
Árbol AVL actualizado:
50
/
30

Número: 70
Árbol AVL actualizado:
50
/ \
30 70

Número: 20
Árbol AVL actualizado:
50
/ \
/ \
30 70
/
20

Número: 40
Árbol AVL actualizado:
50
/ \
/ \
30 70
/ \
20 40

Número: 60
Árbol AVL actualizado:
50
/ \
/ \
30 70
/ \ /
20 40 60

Número: 80
Árbol AVL actualizado:
50
/ \
/ \
30 70
/ \ / \
20 40 60 80

Número: 35
Árbol AVL actualizado:
50
/ \
/ \
/ \
30 70
/ \
/ \
/ \
20 40 60 80
/
35

Número: -1
Programa terminado.