

**Taller #3 – Estructuras de Datos – Árbol AVL**



Pontificia Universidad  
**JAVERIANA**  
Colombia

**Andrés Vélez  
Echeverry**

**Felipe Londoño  
Santiago Fandiño**

**Profesor: Alejandro Martínez Castro.**

**Pontificia Universidad Javeriana Facultad de ingeniería.**

## **Indicaciones de Compilación.**

- **Compilación:** g++ -std=c++11 ArbolAVL.hxx NodoBinario.hxx main.cpp -o programa
- **Ejecución:** ./programa

## **TAD Nodo Binario.**

### **Datos Mínimos.**

- dato, tipo T (tipo de dato que se especifique), representa el dato que almacena el nodo.
- hijoIzq, Nodo Binario, representa el nodo de la izquierda.
- hijoDer, Nodo Binario, representa el nodo de la derecha.

### **Operaciones.**

- NodoBinario(), constructor predeterminado, crea el nodo.
- NodoBinario (T val), constructor que asigna un valor inicial al nodo.
- ~NodoBinario(), es un destructor.
- ObtenerDato(), retorna el dato que contiene el nodo.
- FijarDato(T val), cambio el dato del nodo por val.
- ObtenerHijoIzq(), retorna el hijo izquierdo de un nodo.
- ObtenerHijoDer(), retorna el hijo derecho de un nodo.
- FijarHijoIzq(NodoBinario izq), cambia el contenido del nodo izquierdo del padre por izq.
- FijarHijoDer(NodoBinario der), cambia el contenido del nodo derecho del padre por der.
- esHoja(), indica si el nodo actual es hoja o no.
- altura(), retorna la altura del nodo.
- preOrden(), representa el recorrido en preOrden.
- posOrder(), representa el recorrido del árbol en posOrden.
- inOrder(), representa el recorrido del árbol en inOrden.
- tamaño(), devuelve el tamaño del subárbol con raíz del nodo actual.

## **TAD Árbol AVL.**

### **Datos Mínimos.**

- raíz, Nodo Binario, representa la raíz del árbol.
- datoRaiz, tipo T (tipo de dato que se especifique), representa el dato que almacena el nodo raíz.

### **Operaciones.**

- ArbolAVL(), constructor predeterminado, crea el árbol.
- ~ArbolAVL(), es un destructor.
- esVacio(), retorna si el árbol tiene o no nodos.
- altura(), retorna la altura del árbol.
- tamano(), retorna el tamano del árbol.
- insertar(T val), dado un valor este se agrega como nodo al árbol.
- insertarRekursivo, representa una función auxiliar que permite la inserción de manera recursiva.

- eliminar(T val), dado un valor este se eliminar como nodo del árbol.
- eliminarRecursivo, representa una función auxiliar que permite la inserción de manera recursiva.
- Balancear, permite luego de cada ejecución realizar las rotaciones necesarias al árbol.
- buscar, permite buscar en el árbol un nodo específico.
- preOrden(), representa el recorrido en preOrden.
- posOrder(), representa el recorrido del árbol en posOrden.
- inOrder(), representa el recorrido del árbol en inOrden.
- nivelOrden(), representa el recorrido del árbol en nivelOrden.
- rotaciónizquierdaDerecha, Nodo Binario, representa la operación de balanceo en este caso la rotación por izquierda y luego por derecha.
- rotaciónderechaIzquierda, Nodo Binario, representa la operación de balanceo en este caso la rotación por derecha y luego por izquierda.
- rotacionDerecha, Nodo Binario, representa la operación de balanceo en este caso la rotación por derecha.
- rotacionIzquierda, Nodo Binario, representa la operación de balanceo en este caso la rotación por izquierda.

### Diagrama de Relación de Tads.

