

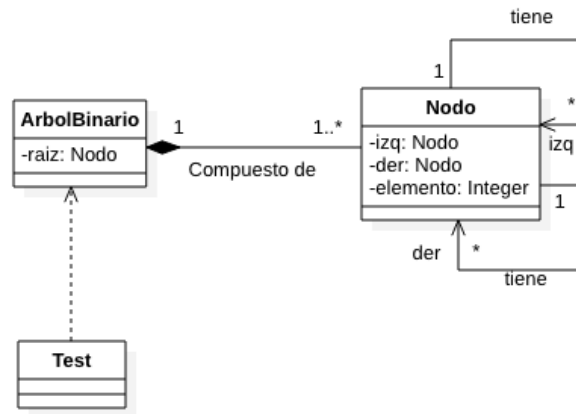


**UNIVERSIDAD DEL CAUCA -
FACULTAD DE INGENIERIA ELECTRONICA Y TELECOMUNICACIONES
PROGRAMA DE INGENIERIA DE SISTEMAS
CURSO DE LABORATORIO DE ESTRUCTURAS DE DATOS II: GRUPO A**

Práctica 4: Árbol Binario

En esta práctica se llevará a cabo la implementación básica en java de un árbol binario. Recuerden que los árboles binarios son estructuras de datos **no lineales**, donde cada nodo del árbol solo puede tener un hijo izquierdo y un hijo derecho, de ahí el nombre **Binario**.

Diagrama de Clases



Funcionalidades para implementar en esta práctica:

1. Crear un método recursivo que permita retornar un nodo del árbol.
2. Crear un método insertar.
3. Crear un método recursivo que permita eliminar un elemento del árbol. En el caso de que el nodo a eliminar sea la raíz se eliminará por completo el árbol binario.
4. Rediseñar la estructura básica del árbol convirtiéndolo en genérico.

Implementación Básica

```
//ArbolBinario.java
package Negocio;
public class ArbolBinario {
    private Nodo raiz;
    public ArbolBinario() {
        this.raiz = null;
    }
    public Nodo getRaiz() {
        return raiz;
    }
    public void setRaiz(Nodo raiz) {
        this.raiz = raiz;
    }

    public boolean buscarElemento(Nodo raiz, int elemento) {
        return buscarElemento(raiz, elemento);
    }
}
```



**UNIVERSIDAD DEL CAUCA -
FACULTAD DE INGENIERIA ELECTRONICA Y TELECOMUNICACIONES
PROGRAMA DE INGENIERIA DE SISTEMAS
CURSO DE LABORATORIO DE ESTRUCTURAS DE DATOS II: GRUPO A**

```
public boolean buscarElemento(Nodo raiz, int elemento) {
    if (raiz == null) {
        return false;
    }
    if (raiz.getElemento() == elemento) {
        return true;
    }
    if (buscarElemento(raiz.getIzq(), elemento)) {
        return true;
    } else if (buscarElemento(raiz.getDer(), elemento)) {
        return true;
    }
    return false;
}

public void mostrarArbolSangrado() {
    mostrarArbolSangrado(raiz, 0);
}

public void mostrarArbolSangrado(Nodo arbol, int nivel) {
    if (arbol != null) {
        mostrarArbolSangrado(arbol.getIzq(), nivel + 1);
        System.out.println(" " + nivel + " " + arbol.getElemento());
        mostrarArbolSangrado(arbol.getDer(), nivel + 1);
    }
}

}

//Nodo.java
package Negocio;
public class Nodo {
    private int elemento;
    private Nodo izq, der;
    public Nodo(int elemento) {
        this.elemento = elemento;
        this.izq = null;
        this.der = null;
    }
    public int getElemento() {
        return elemento;
    }
    public void setElemento(int elemento) {
        this.elemento = elemento;
    }
    public Nodo getIzq() {
        return izq;
    }
    public void setIzq(Nodo izq) {
        this.izq = izq;
    }
    public Nodo getDer() {
```



**UNIVERSIDAD DEL CAUCA -
FACULTAD DE INGENIERIA ELECTRONICA Y TELECOMUNICACIONES
PROGRAMA DE INGENIERIA DE SISTEMAS
CURSO DE LABORATORIO DE ESTRUCTURAS DE DATOS II: GRUPO A**

```
        return der;
    }
    public void setDer(Nodo der) {
        this.der = der;
    }
}

//Test.java
package Presentacion;
import Negocio.*;
public class Test {
    public static void main(String[] args) {
        Arbol ObjetoArbol = new Arbol();
        //crear la raiz
        arbol.setRaiz(new Nodo(3));
        //crear los nodos
        arbol.getRaiz().setDer(new Nodo(1));
        arbol.getRaiz().setIzq(new Nodo(4));
        arbol.getRaiz().getDer().setDer(new Nodo(8));
        arbol.getRaiz().getDer().setIzq(new Nodo(9));
        arbol.getRaiz().getIzq().setDer(new Nodo(5));
        arbol.getRaiz().getIzq().setIzq(new Nodo(2));
        //mostrar el árbol resultante
        ObjetoArbol.imprimirArbolNiveles ();
    }
}
```

