



UNIVERSIDAD TECNOLÓGICA DE  
PANAMÁ CENTRO REGIONAL DE  
CHIRIQUÍ  
FACULTAD DE INGENIERÍA DE SISTEMAS  
COMPUTACIONALES



**CARRERA:**  
Gestión y Desarrollo de Software

**ACTIVIDAD No. 4**

**LABORATORIO No. 1**

**“Laboratorio 1”**

**ASIGNATURA:** Estructura de Datos II

**DOCENTE:**  
Profa. Nunehar Mondul

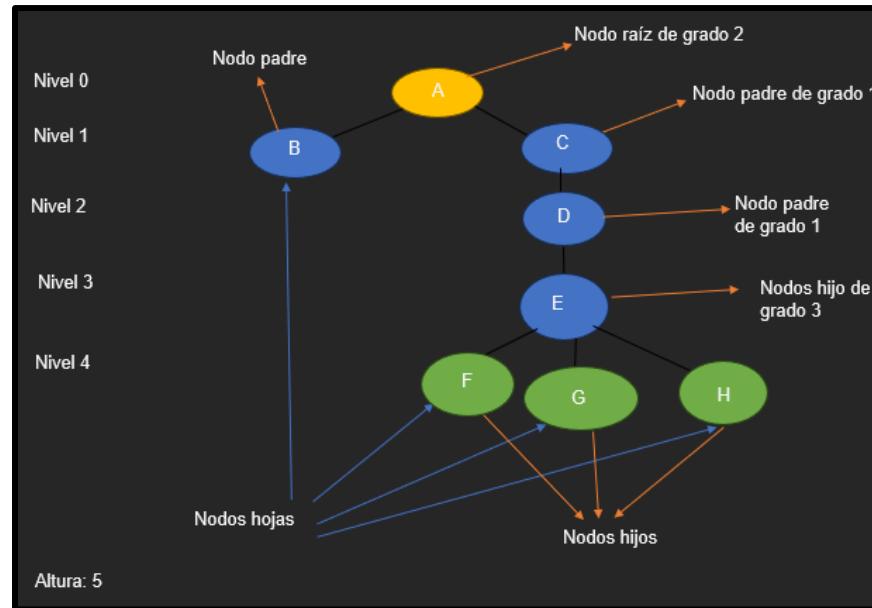
**ESTUDIANTE/s:**  
Jorge Jiménez (4-826-874)  
Briant Arango (4-825-620)  
Jonathán Martínez (4-824-2482)

*I SEMESTRE 2025*

**FECHA:**  
04/15/2025

## Desarrollo

1. "Dibujar un árbol de cuatro niveles que contenga 4 nodos hojas. Mencionar los niveles, altura, grados de los nodos, raíz, nodos padre, nodos hijos.."



2. "Investigar cual es la utilidad de los árboles en la programación."

-Los árboles son estructuras jerárquicas utilizadas en programación para: Organizar datos de forma eficiente, facilitar búsquedas rápidas como en bases de datos y sistemas de archivos, representar estructuras jerárquicas, e implementar algoritmos de inteligencia artificial, compresión de datos, etc.

3. "¿Qué es un árbol binario? Diferencia con un árbol normal y su uso"

-Árbol binario: es un árbol en el que ningún nodo puede tener más de dos sub-árboles. En un árbol binario cada nodo puede tener cero, uno o dos hijos (sub-árboles). Se conoce el nodo de la izquierda como hijo izquierdo y el nodo de la derecha como hijo derecho.

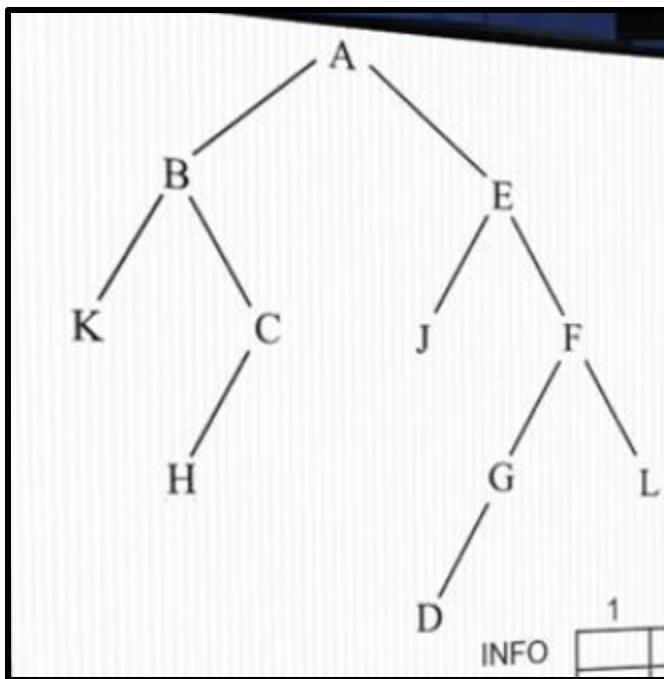
Diferencias:

- El binario limita a dos hijos por nodo.
- Los árboles normales permiten más flexibilidad, pero son menos eficientes para búsquedas.

Usos del árbol binario:

- Árbol binario de "búsqueda" (BST) para búsquedas eficientes.
- Árboles de "expresión matemática".
- Árboles "AVL" y "Red-Black" para estructuras balanceadas.

4. "Dibujar tabla de asignación de memoria para el siguiente árbol binario."

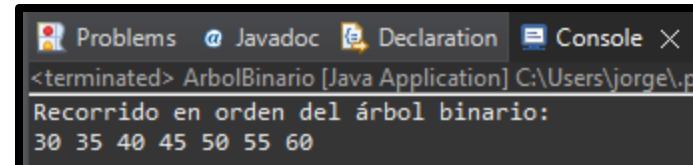


	1	2	3	4	5	6	7	8	9	10	11	12
N	C	F	D		A	L	H	E	J	G	B	K
L	7	10	0	0	H	0	0	9	0	0	12	0
R	0	6	0	0	8	6	0	2	0	3	1	0

5. "Para el siguiente código documentar el código y cambiar la inserción de los nodos para que quede con este formato in orden 30, 35, 40, 45, 50, 55, 60.  
(Captura de pantalla de salida)."

```
arbol.java X
1 package arbol;
2 class Nodo { // Esta es una clase que representa un nodo del árbol
3     int valor;
4     Nodo izquierdo, derecho;
5
6     public Nodo(int item) {
7         valor = item;
8         izquierdo = derecho = null;
9     }
10}
11
12 class ArbolBinario { // Esta es la clase principal del árbol binario
13     Nodo raiz;
14
15     ArbolBinario() { // El constructor que crea un árbol vacío
16         raiz = null;
17     }
18
19     void insertar(int valor) { // El método público para insertar un valor
20         raiz = insertarRec(raiz, valor);
21     }
22
23     Nodo insertarRec(Nodo raiz, int valor) { // El método recursivo que inserta un
24         if (raiz == null) {                         // nodo en el lugar correcto
25             raiz = new Nodo(valor);
26             return raiz;
27         }
28
29         if (valor < raiz.valor) {
```

```
30         raiz.izquierdo = insertarRec(raiz.izquierdo, valor);
31     } else if (valor > raiz.valor) {
32         raiz.derecho = insertarRec(raiz.derecho, valor);
33     }
34
35     return raiz;
36 }
37
38 void enOrden() { // El método para recorrer el árbol en orden.
39     enOrdenRec(raiz);
40 }
41
42 void enOrdenRec(Nodo raiz) { // Recorrido en orden de: izquierda, raíz, derecha
43     if (raiz != null) {
44         enOrdenRec(raiz.izquierdo);
45         System.out.print(raiz.valor + " ");
46         enOrdenRec(raiz.derecho);
47     }
48 }
49
50 public static void main(String[] args) {
51     ArbolBinario arbol = new ArbolBinario();
52
53     arbol.insertar(45); // Se insertan los valores en orden para que
54     arbol.insertar(35); // el recorrido en orden sea igual.
55     arbol.insertar(30);
56     arbol.insertar(40);
57     arbol.insertar(50);
58     arbol.insertar(55);
59     arbol.insertar(60);
60
61     System.out.println("Recorrido en orden del árbol binario:");
62     arbol.enOrden();
63 }
64 }
```



A screenshot of a Java IDE's console window. The tabs at the top are 'Problems', 'Javadoc', 'Declaration', and 'Console'. The status bar at the bottom shows '<terminated> ArbolBinario [Java Application] C:\Users\jorge\p'. The main area of the window displays the output of the program: 'Recorrido en orden del árbol binario:' followed by the values '30 35 40 45 50 55 60' on a single line.

```
Problems Javadoc Declaration Console
<terminated> ArbolBinario [Java Application] C:\Users\jorge\p
Recorrido en orden del árbol binario:
30 35 40 45 50 55 60
```