

Apuntes

Tema **Arboles - Generalidades**

*"La experiencia no es lo que te sucede, sino lo que haces con lo que te sucede."
Aldous Huxley (1894-1963) Novelista, ensayista y poeta inglés.*

Arboles

¿Qué es un Árbol?



Los árboles representan estructuras no lineales y dinámicas. Un árbol se define como un conjunto finito de elementos llamados nodos que guardan entre ellos una relación jerárquica. Existe un nodo diferenciado llamado **raíz** del árbol, y los demás nodos forman conjuntos diferentes cada uno de los cuales es a su vez un árbol, a estos árboles se los denominó **subárboles**.

Atributos:

NODO: Es cada uno de los elementos del árbol

RAÍZ: Es un nodo especial del que descienden todos los demás nodos. Este nodo es el único que no tiene "padre".

HOJA (o nodo terminal): Es donde termina el árbol, no tiene ningún descendiente (nodos

HIJO: Cada nodo que no es hoja y tiene debajo de él 1 o varios nodos

PADRE: Todo nodo excepto la raíz, tiene asociado un ÚNICO nodo predecesor al que llamaremos padre.

HERMANO: Son los nodos que son hijos de un mismo padre .

NODO INTERNO: Cualquier nodo que no sea una hoja.

NIVEL: Cada nodo de un árbol tiene asignado un número de nivel superior en una unidad a su padre. Arrancando del nivel 1 el nodo raíz. (*propiedad del nodo*).

ALTURA: Es el máximo número de aristas que tiene la rama mas grande. Se puede calcular como el **NIVELMAXIMO - 1** (*propiedad del árbol o sub árbol*).

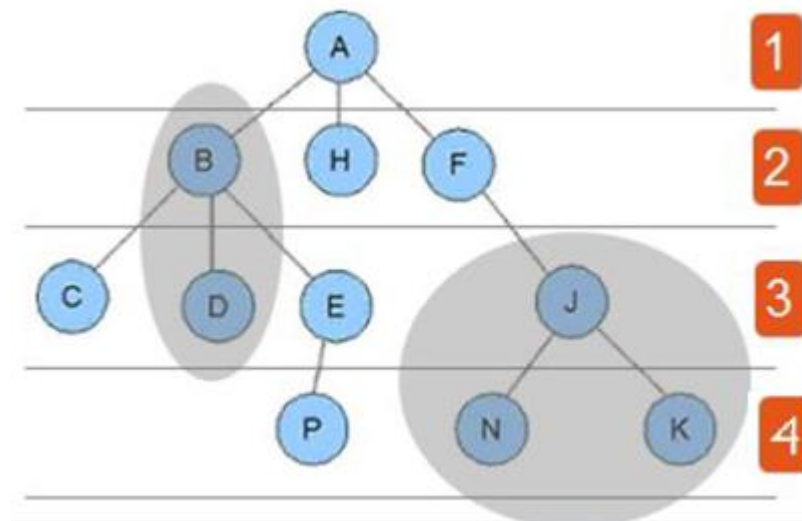
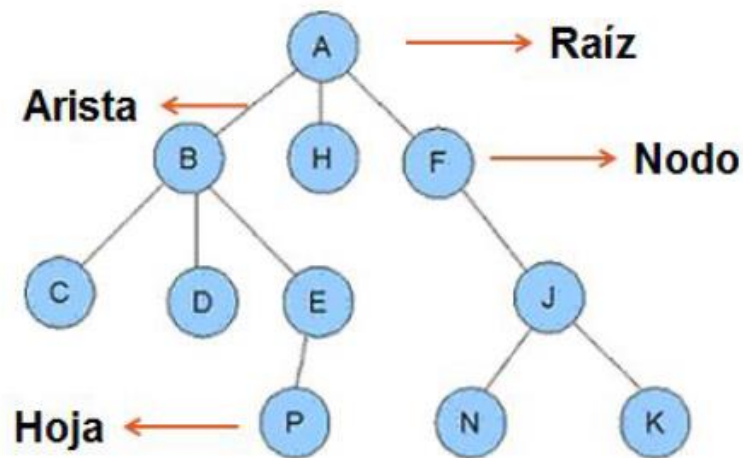
CAMINO: Una sucesión de enlaces que conectan dos nodos.

RAMA: Camino que termina en una hoja.

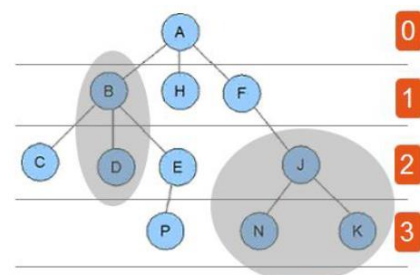
PROFUNDIDAD de un nodo: Es el número de aristas desde la raíz, hasta ese nodo.

PESO DE UN ÁRBOL: Número de hojas de un árbol.

BOSQUE: Colección de 2 o más árboles.

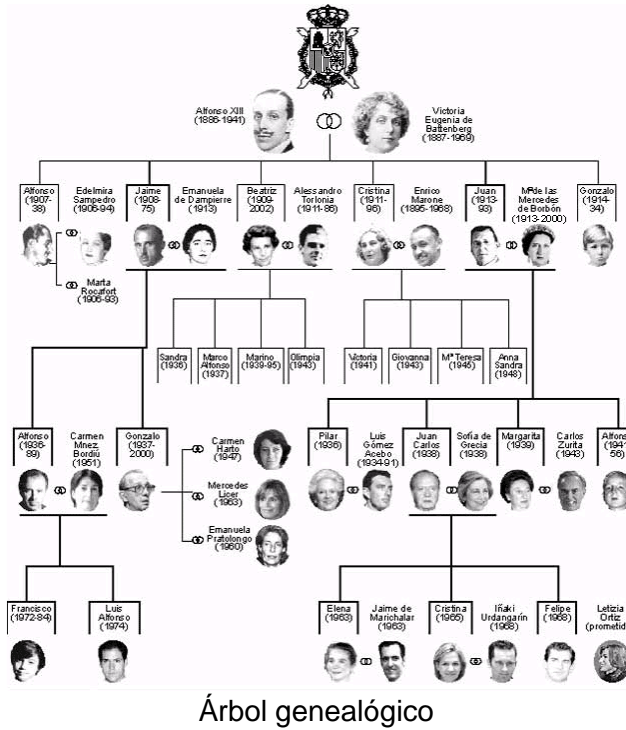


Niveles de un árbol

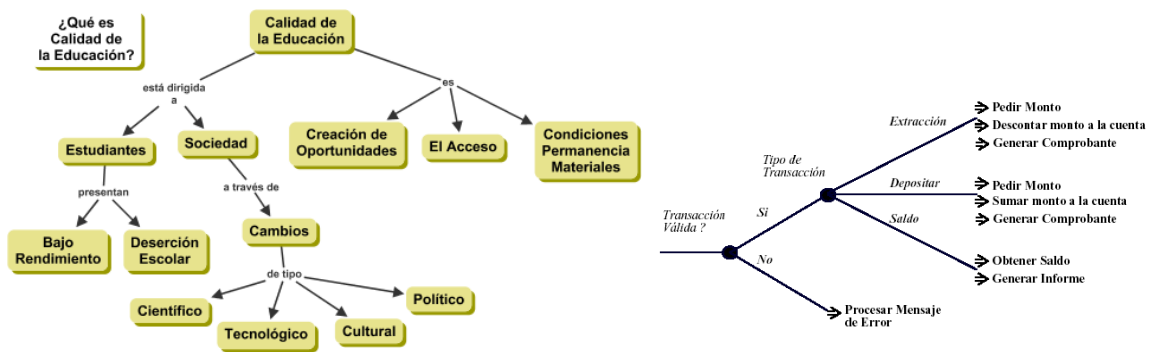


Algunos autores cuentan desde 0

Ejemplos de árboles:



Árbol genealógico



Árboles de decisión

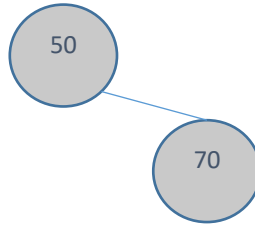
Altura de un árbol

Árbol A

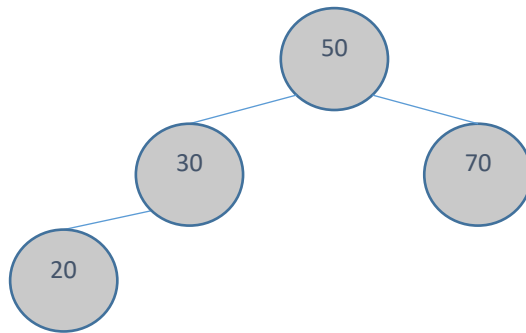
50

ALTURA(A): 0

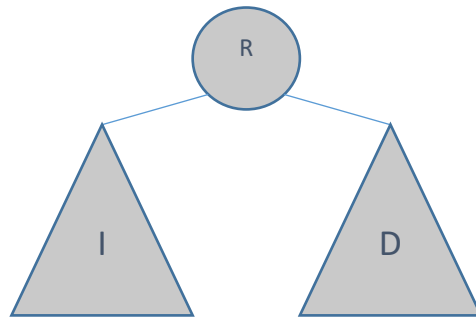
Árbol B

 $ALTURA(B): 1$

Árbol C

 $ALTURA(C): 2$

Árbol D

 $ALTURA(D):$
 $MAX[ALTURA(I), ALTURA(D)]+1$

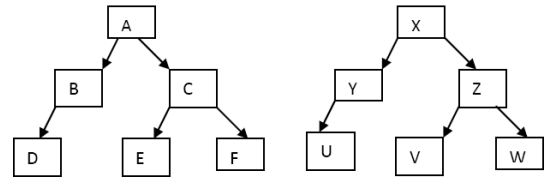
Arboles Binarios

¿Qué es un Árbol binario?

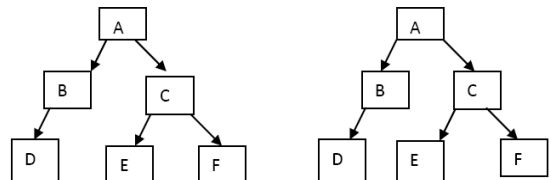
Un árbol binario es una estructura de datos en la cual cada nodo siempre tiene un hijo izquierdo y un hijo derecho. No pueden tener más de dos hijos (*de ahí el nombre "binario"*). Si algún hijo tiene como referencia a **null**, es decir que no almacena ningún dato, entonces este es llamado un nodo externo. En el caso contrario el hijo es llamado un nodo interno. Usos comunes de los árboles binarios son los árboles binarios de búsqueda, los montículos binarios y Codificación de Huffman.

Tipos de árboles Binarios

- Árboles binarios similares: Son aquellos cuya estructura es igual



- Árboles binarios equivalentes o copias: son aquellos cuya su estructura es igual y los datos contenidos son iguales.



- Árboles binarios equilibrados: Sus alturas o profundidades se diferencian como máximo en 1 unidad.
- Árboles binarios llenos: Un árbol binario lleno es un árbol en el que cada nodo tiene cero o dos hijos, es decir su factor de equilibrio es 0.
- Árboles binarios perfecto: Un árbol binario perfecto es un árbol binario lleno en el que todas las hojas (vértices con cero hijos) están a la misma profundidad (distancia desde la raíz, también llamada altura).

Árboles y recursividad

Los árboles son una estructura de datos recursiva, compuesta **por un elemento denominado raíz y por árboles asociados**, los cuales se denominan **sub-árboles**. El hecho de definir la estructura de datos en términos de sí misma es lo que genera que se denomine recursiva. Es por eso que resulta indispensable poder comprender los siguientes conceptos de recursividad:

¿Qué es recursividad o recursión?: Según se define en los fundamentos de programación: “Es una herramienta muy potente y útil para la resolución de problemas”.

Estos problemas están basados en su propia definición.

¿Qué es una estructura recursiva?: Es una estructura de datos en la que los elementos se definen a sí mismos.

¿Qué es un método recursivo?: Es un método que directa o indirectamente se hace una llamada a sí mismo.

¿Cuáles son los problemas de la recursión?: Ciclo infinito o grande, disparando la necesidad recursos, sobre todo espacio.

¿Solución?: Asegurarse llamadas con instancias diferentes de menor complejidad.

Representación de árboles binarios en memoria

- Representación enlazada o de punteros
- Representación secuencial o con arrays

El principal requerimiento para cualquier representación es que para cualquier árbol T se tenga acceso directo a la raíz de dicho árbol y dado un nodo N del árbol se tenga acceso directo a sus hijos.

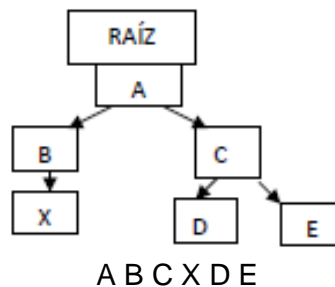
Representación enlazada o de punteros:

El árbol se representa por medio de una lista enlazada especial en la que cada nodo tiene 3 campos:

- IZQ; campo enlace o puntero al hijo izquierdo del nodo, si dicho hijo no existe, contiene NIL.
- DER; campo enlace o puntero al hijo derecho.
- RAIZ; variable que apunta al nodo raíz del árbol.

Un árbol estará vacío cuando $RAIZ \leftarrow NIL$.

Para las inserciones y borrados utilizaremos la lista DISP, va a ser una lista enlazada normal en la que cada nodo tendrá los 3 campos siendo el IZQ el campo enlace.



Recorrido de árboles binarios

Recorrido de un árbol es el proceso que consiste en acceder una sola vez a cada nodo del árbol; hay 3 maneras estándar de recorrer un árbol y en las tres se dan los mismos pasos y también coinciden en que se recorre primero su árbol izquierdo y luego el derecho.

Las tres maneras se diferencian en el momento que se accede al nodo raíz.

- 1. – PREORDEN:**
 - 1º se procesa la raíz.
 - 2º subárbol izquierdo.
 - 3º subárbol derecho.
- 2. – INORDEN:**
 - 1º subárbol izquierdo.
 - 2º se procesa la raíz.
 - 3º subárbol derecho.
- 3. – POSTORDEN:**
 - 1º subárbol izquierdo.
 - 2º subárbol derecho.
 - 3º se procesa la raíz