

## TRABAJO PRÁCTICO Nº 5

### Árboles Binarios de Búsqueda y AVL

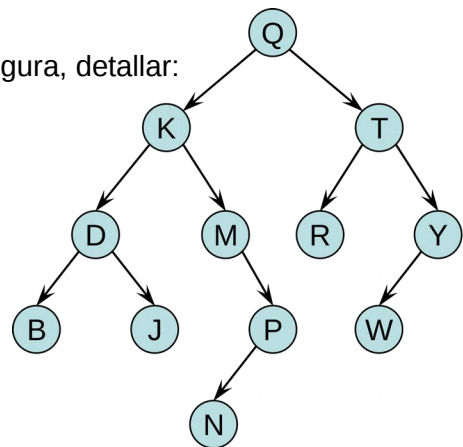
#### Objetivo:

Comprender los conceptos generales sobre árboles e implementar sus operaciones básicas y métodos existentes para el recorrido de los elementos de un árbol.

#### Ejercicios:

1) Utilizando el árbol con raíz Q, que se muestra en la figura, detallar:

- Nodos hojas.
- Nodos no terminales o interiores.
- Ramas.
- Descendientes de “K”.
- Ancestros de “J”.
- Camino y longitud de camino de “K” a “M”.



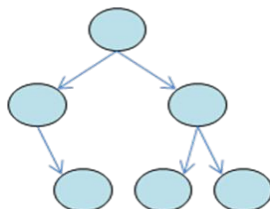
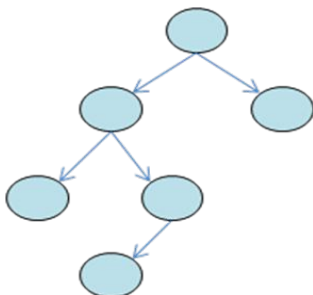
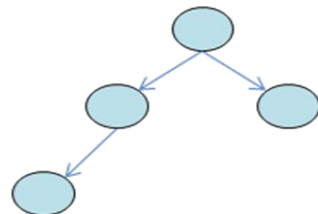
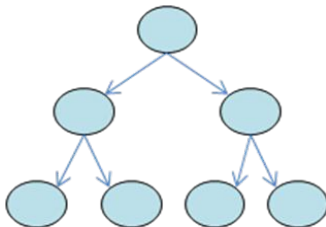
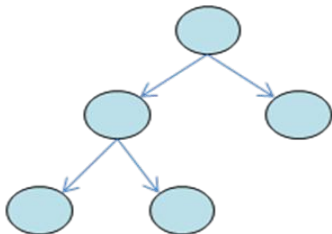
2) Mostrar como quedaría el árbol anterior después de los siguientes cambios.

- Añadiendo nodo C.
- Añadiendo nodo Z.
- Añadiendo nodo X.
- Suprimiendo nodo M.
- Suprimiendo nodo Q.
- Suprimiendo nodo R.

3) Mostrar en que orden se muestran los elementos del árbol correspondiente al ejercicio **(1)**, de acuerdo a los siguientes recorridos:

- Un recorrido *inorden*.
- Un recorrido en *postorden*.
- Un recorrido en *preorden*.
- Un recorrido *por niveles*.

- 4) ¿Cuántos nodos puede llegar a tener un árbol binario con 5 niveles?
- 5) ¿Cuál es el número mínimo de nodos de un árbol binario completo con 5 niveles?
- 6) Dibuje el árbol binario que da lugar a los siguientes recorridos en preorden, inorden y postorden:
- (a) Preorden: H, J, I, E, C, D, B, A, G, F  
(b) Inorden: I, J, H, D, C, A, B, G, E, F  
(c) Postorden: I, J, D, A, G, B, C, F, E, H
- 7) Investigue los conceptos relacionados a árboles: **isomorfos**, **semejantes** y **espejos** o **iguales**. Explique cuando se aplican dichos conceptos y de ejemplos gráficos.
- 8) Analice las siguientes gráficas y para cada una clasifique, según corresponda, en: **no equilibrado**, **equilibrado**, **completo** ó **lleno**.



- 9) Mostrar gráficamente todas las rotaciones que implica la inserción de cada uno de los siguientes elementos: 10, 100, 20, 80,40, 70 en un árbol AVL. Mostrar en cada inserción el árbol resultante y sus rotaciones.
- 10) En base al árbol de ejercicio anterior, se pide eliminar los valores 10 y 20. Mostrar para cada elemento eliminado el árbol resultante y todas las rotaciones implicadas.
- 11) Cree la clase **ArbolBinarioEnlazado<T>** implementando todos los métodos de la interfaz **BinarySearchTreeADT** la cual a su vez extiende de **BinaryTreeADT**. Ambas interfaces y sus métodos se describen en los capítulos 12.4 y 13.2 del libro [Lewis and Chase].
- 12) Cree el método **esSemejanteA (BinarySearchTree<T> otroArbol)**, el cual deberá determinar si el árbol actual es semejante al árbol ingresado como parámetro.
- 13) Cree el método **existeCamino(T v1, T v2)**, el cual deberá determinar si existe o no un camino entre el nodo que contiene el valor **v1** y el nodo que contiene el valor **v2**.
- 14) Cree el método **esAncestro(T v1, T v2)**, el cual deberá determinar si el nodo que contiene a **v1** es ancestro del nodo cuyo contenido es **v2**.