

## Trabajo práctico 2.1 - Árboles

### Programación 3 - TUDAI

#### Ejercicio 1.

Implemente la estructura de Árbol Binario para búsquedas.

Métodos:

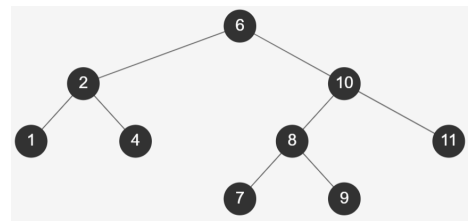
- Integer getRoot(), boolean hasElem(Integer), boolean isEmpty(), void insert(Integer), boolean delete(Integer), int getHeight(), void printPosOrder(), void printPreOrder(), void printInOrder(), List getLongestBranch(), List getFrontera(), Integer getMaxElem(), List getElemAtLevel(int)
- ¿Cuál es la complejidad de cada uno de estos métodos?

#### Ejercicio 2

Dado un árbol binario de búsquedas que almacena números enteros, implementar un algoritmo que retorne la suma de todos los nodos internos del árbol.

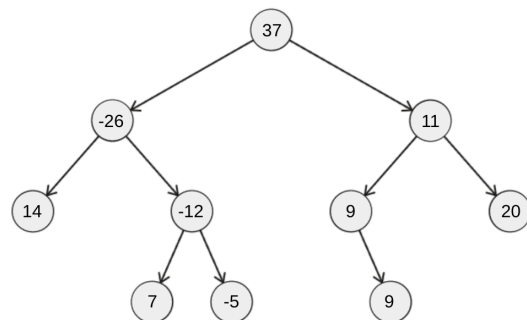
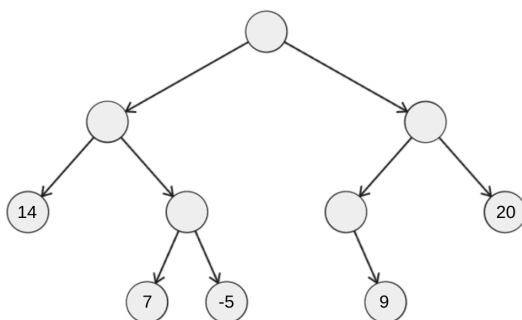
#### Ejercicio 3

Dado un árbol binario de búsqueda que almacena números enteros y un valor de entrada K, implementar un algoritmo que permita obtener un listado con los valores de todas las hojas cuyo valor supere K. Por ejemplo, para el árbol de la derecha, con un valor K = 8, el resultado debería ser [9, 11].



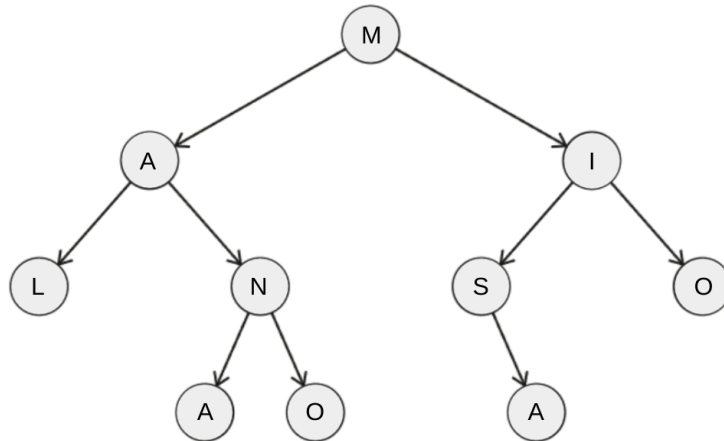
#### Ejercicio 4

Se posee un árbol binario (no de búsqueda), donde los nodos internos están vacíos, mientras que las hojas tienen valores enteros. Se debe implementar un método que recorra el árbol y coloque valores en los nodos vacíos (los nodos internos). El valor de cada nodo interno debe ser igual al valor de su hijo derecho, menos el valor de su hijo izquierdo. En caso de que el nodo tenga un solo hijo, el valor del hijo faltante se reemplaza por un 0. Por ejemplo, tomando como entrada el árbol de la izquierda, el árbol resultante debería quedar con los mismos valores que el de la derecha.



### Ejercicio 5

Dado un árbol binario donde todos los nodos poseen un carácter, de manera que cada rama del árbol contiene una palabra, implementar un algoritmo que busque y retorne todas las palabras que posea exactamente N vocales (ni más ni menos). Por ejemplo, para el siguiente árbol, con una entrada de  $N = 1$ , el algoritmo debería retornar ["MAL"]. En cambio, para un  $N = 2$ , debería retornar ["MANA", "MANO", "MISA"].



### Ejercicio 6

Se desea desarrollar una aplicación para mejorar la atención de una biblioteca en cuanto a la búsqueda de libros dentro del catálogo disponible. Cada libro estará compuesto por un identificador único y datos propios de los libros (título, autor, géneros, año de publicación, cantidad de ejemplares, etc.)

Se sabe, además, que los libros nuevos se agregan al catálogo en horarios fuera de la atención al público.

Se desean proveer los siguientes servicios:

- Obtener la cantidad de ejemplares de un libro dado su identificador único.
- Obtener todos los libros de un género dado.
- Obtener todos los libros publicados entre dos años de publicación dados.

Responda y justifique:

- 1) ¿Qué estructura de datos utilizaría para almacenar todos los libros en memoria dentro de la aplicación?
- 2) ¿Cómo resolvería cada uno de los servicios solicitados? ¿Utilizaría alguna estructura adicional de acceso para mejorar el costo de respuesta de cada servicio?