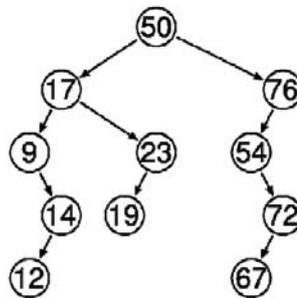


**Reglas del juego:**

- Durante este examen usted no puede pedir ABSOLUTAMENTE nada prestado a sus compañeros, ni hablar con ellos.
- Durante este examen usted no podrá utilizar ningún tipo de dispositivo electrónico, entre otros, calculadoras, celulares, asistentes personales, etc.
- Su nombre y su firma a la aceptación del compromiso de no hacer fraude, en la hoja de respuestas a este examen, deben ir en lapicero. Si emplea más de una hoja márkelas TODAS de igual forma.
- Por ningún motivo puede salir del salón, antes de terminar el examen. De manera que si se retira se considerará que terminó su trabajo.

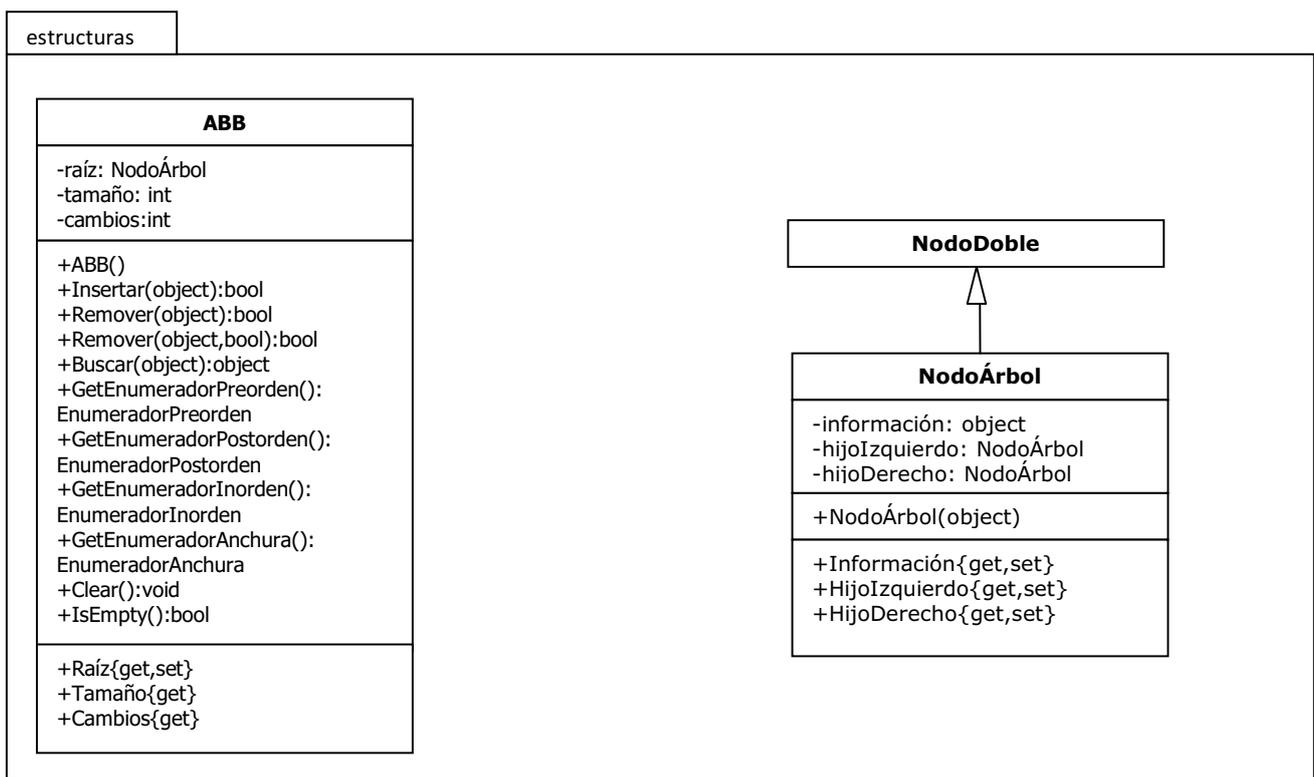
## 1. (1.0) Resuelva los siguientes ejercicios:

- a. (0.5) Este ejercicio fue propuesto por el grupo encargado del Árbol AVL, en las exposiciones de Árboles N-Arios: Balancee el siguiente árbol, de tal forma que quede como un árbol AVL. Indique paso a paso cual es el proceso



- b. (0.5) **Explique** cuál es la complejidad temporal, en el peor de los casos, de la operación de búsqueda en un Árbol AVL.

## 2. (3.0) Usted cuenta con la clases ABB y NodoÁrbol que se presentan en el siguiente diagrama:



Se ha considerado necesario, incluir los siguientes métodos en la clase ABB:

**a. (1.0) Un método a partir del cual se pueda obtener el abuelo de un nodo**

A Sandcastle Documented Class Library

### Abuelo Method (información)

Namespaces ► estructuras ► ABB ► Abuelo(Object) C# ▼

Dado un objeto información, si éste se encuentra en el Árbol Binario de Búsqueda, retorna la información del nodo abuelo. Por ejemplo, para el siguiente ABB:  Si se invoca al método con el objeto 6 como información, retornará el objeto 7 como información resultado.

**Declaration Syntax**

C# Visual Basic Visual C++

```
public Object Abuelo(  
    Object información  
)
```

**Parameters**

**información (Object)**  
El objeto información a partir del cual se buscará la información de nodo Abuelo.

**Return Value**

La información del nodo Abuelo. Si el nodo que se recibe por parámetro no existe en el árbol, retorna null. Si el abuelo no existe, retorna null.

---

Assembly: estructuras (Module: estructuras) Version: 1.0.0.0 (1.0.0.0)

**b. (1.0) Un método a partir del cual se pueda obtener el tío de un nodo**

A Sandcastle Documented Class Library

### Tío Method (información)

Namespaces ► estructuras ► ABB ► Tío(Object) C# ▼

Dado un objeto información, si éste se encuentra en el Árbol Binario de Búsqueda, retorna la información del nodo tío. Por ejemplo, para el siguiente ABB:  Si se invoca al método con el objeto 8 como información, retornará el objeto 5 como información resultado.

**Declaration Syntax**

C# Visual Basic Visual C++

```
public Object Tío(  
    Object información  
)
```

**Parameters**

**información (Object)**  
El objeto información a partir del cual se buscará la información del nodo Tío.

**Return Value**

La información del nodo Tío. Si el nodo que se recibe por parámetro no existe en el árbol, retorna null. Si el tío no existe, retorna null.

---

Assembly: estructuras (Module: estructuras) Version: 1.0.0.0 (1.0.0.0)

**c. (1.0) Un método a partir del cual se puedan obtener los nietos de un nodo**

A Sandcastle Documented Class Library

### Nietos Method (información)

Namespaces ► estructuras ► ABB ► Nietos(Object) C# ▼

Dado un objeto información, si este se encuentra en el Árbol Binario de Búsqueda, retorna una Cola Dinámica que contiene la información de todos los nietos. Por ejemplo, para el siguiente ABB:  Si se invoca al método con el objeto 7 como información, retornará una cola que contendrá la siguiente información: 4-6-8-10. El orden puede ser diferente de acuerdo a como se adicione la información de los nietos en la estructura

**Declaration Syntax**

C#  Visual Basic  Visual C++

```
public ColaDinámica Nietos(
    Object información
)
```

**Parameters**

**información (Object)**  
El objeto información a partir del cual se obtendrán los nietos

**Return Value**

La ColaDinámica que contiene la información de los nietos. Si el nodo que se recibe por parámetro no existe en el árbol, retorna null. Si el nodo no tiene nietos retorna null

---

Assembly: estructuras (Module: estructuras) Version: 1.0.0.0 (1.0.0.0)

Elabore el código en C# de estos métodos, cumpliendo con la descripción dada para el mismo. Tenga en cuenta que si necesita métodos auxiliares de la clase ABB, debe escribir el código correspondiente a dichos métodos. Si requiere utilizar métodos auxiliares de otras estructuras distintas a ABB, no debe escribir su código, simplemente úselos y dé una breve descripción de lo que haría cada uno.

3. (1.0) Elabore la clase EnumeradorProfundidad, que permita recorrer en profundidad un grafo dirigido, implementado mediante conjunto de adyacencias. Esta clase debe implementar System.Collections.IEnumerator, por lo tanto, debe contener los métodos definidos en la interfaz. Para la solución propuesta usted podrá utilizar únicamente colas o pilas como estructuras auxiliares.

```
using System;
using System.Collections;
using System.Linq;
using System.Text;
namespace estructuras
{
    public class EnumeradorProfundidad : IEnumerator
    {
        #region Miembros de IEnumerator
        public object Current
        {
        }
        public bool MoveNext()
        {
        }
        public void Reset()
        {
        }
        #endregion
    }
}
```