

**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.
- Las respuestas deben consignarse en el cuadernillo, NO en el enunciado. Las respuestas marcadas en el enunciado no serán tenidas en cuenta.
- Solo podrán presentar el examen los estudiantes que lleguen al salón durante los primeros 15 minutos después de haber empezado la prueba.
- Solo se tendrán en cuenta las respuestas consignadas en el cuadernillo de respuestas.

1. (1.0) A partir del siguiente código, determine cuál enumerador del Árbol Binario de Búsqueda está siendo implementado:

```
using System;
using System.Collections;
using System.Linq;
using System.Text;

namespace estructuras
{
    [Serializable]
    public class Enumerador : IEnumerator
    {
        #region Miembros de IEnumerator

        private int cambios;
        private NodoÁrbol actual;
        private ABB árbol;
        private PilaDinámica pila;

        public Enumerador(int losCambios, NodoÁrbol raíz, ABB elÁrbol)
        {
            cambios = losCambios;
            actual = null;
            árbol = elÁrbol;
            NodoÁrbol nodo = raíz;
            PilaDinámica aux = new PilaDinámica();
            pila = método(raíz, aux);
        }

        public object Current
        {
            get
            {
                if (cambios == árbol.Cambios)
                {
                    if (actual != null)
                        return actual.Información;
                    return null;
                }
                else
                    throw new InvalidOperationException("Operación Inválida.");
            }
        }
    }
}
```

```

public bool MoveNext()
{
    if (cambios == árbol.Cambios)
    {
        if (!pila.IsEmpty())
        {
            actual = (NodoÁrbol)pila.Pop();
            NodoÁrbol raiz = null;
            raiz = (NodoÁrbol)pila.Pop();
            if (raiz != null)
            {
                if (raiz.HijoDer != null && raiz.HijoDer != actual)
                {
                    pila.Push(raiz);
                    pila = método(raiz.HijoDer, pila);
                    if (raiz.HijoDer.HijoDer != null && raiz.HijoDer.HijoDer != actual)
                    {
                        pila = método(raiz.HijoDer.HijoDer, pila);
                    }
                }
                else
                {
                    pila.Push(raiz);
                }
            }
            return true;
        }
        return false;
    }
    else
        throw new InvalidOperationException("Operación Inválida.");
}

public void Reset()
{
    if (cambios == árbol.Cambios)
    {
        actual = null;
        pila.Clear();
        pila = método(árbol.Raíz, pila);
    }
    else
        throw new InvalidOperationException("Operación Inválida.");
}
public PilaDinámica método(NodoÁrbol nodo, PilaDinámica aux)
{
    if (nodo != null)
    {
        aux.Push(nodo);
        if (nodo.HijoIzq != null)
            método(nodo.HijoIzq, aux);
    }
    return aux;
}
#endregion
}
}

```

2. (1.0) Dados los siguientes números: 56, 98, 76, 45, 21, 10, 78, 87, 99; indique cómo quedarían finalmente los siguientes árboles después de la inserción. Tenga en cuenta que los números se insertan en el orden especificado de acuerdo con las restricciones y propiedades de cada árbol. No es necesario indicar el proceso, solo el resultado final.
  - a. (0.5) Para un árbol AVL.
  - b. (0.5) Para un árbol B de orden 2.

