

TAD Arbol Binario

Práctica 11 Arboles Binarios

Objetivo

Esta práctica tiene como objetivo que el alumno repase sus conocimientos sobre la estructura de datos Árbol Binario mediante la implementación de métodos para ejemplificar algunas definiciones sobre árboles binarios y para realizar recorridos sobre éstos.

Descripción general

Dentro de la clase `ArbolBinario` se hará uso de la clase `NodoArbol` para representar un árbol binario y se implementarán los siguientes métodos que permitan conocer datos del árbol binario:

1. **Tamaño:** Cantidad de nodos en el árbol binario.
2. **Altura:** La altura de un árbol es el número máximo de niveles que tiene. Por definición la raíz de un árbol está en el nivel 1 y cualquier nodo está a un nivel más que el de su padre.
3. **Número de hojas:** Un nodo que no tiene hijos se conoce como hoja o nodo externo.
4. **Número de nodos internos:** Los nodos que tienen padre y al menos un hijo se denominan nodos internos.

También se implementarán los tres recorridos básicos de un árbol binario:

1. **Pre-orden:** se visita la raíz del árbol, luego el subárbol izquierdo y finalmente el subárbol derecho. Cada subárbol se visita a su vez en pre-orden.
2. **In-Orden:** se visita el subárbol izquierdo, luego la raíz y finalmente el subárbol derecho. Cada subárbol se visita a su vez en in-orden.
3. **Post-orden:** se visita el subárbol izquierdo, luego el subárbol derecho y finalmente la raíz. Cada subárbol se visita a su vez en post-orden.

Material

El material de esta práctica consta de los siguientes archivos:

- **NodoArbol.class** clase que implementa los nodos de un árbol binario.

- **ArbolBinario.java** implementará los métodos para trabajar con árboles binarios.
- **PruebaArbolBinario.class** programa para probar la clase `ArbolBinario`.
- Documentación:
 - `NodoArbol.html` documentación de la clase `NodoArbol`.

Desarrollo

1. Descargar los archivos `NodoArbol.class`, `ArbolBinario.java` y `PruebaArbolBinario.class` en el directorio donde se va a desarrollar la práctica.
2. Leer la documentación y revisar el material de la práctica. Recordar que las declaraciones y la firma de los métodos proporcionados, así como de los requeridos en el desarrollo de la práctica, no podrán ser modificados.
3. En la clase `ArbolBinario` implementar los siguientes métodos:
 - a. **tamano()** Determina la cantidad de nodos del árbol binario.
 - b. **altura()** Determina la altura del árbol binario.
 - c. **hojas()** Devuelve el número de hojas que contiene el árbol binario.
 - d. **nodosInternos()** Devuelve el número de nodos internos que contiene el árbol binario.
 - e. **imprimir(NodoArbol)** Imprime en pantalla el dato del nodo indicado como parámetro en caso de que este no sea nulo.
 - f. **preOrden()** Realiza un recorrido del árbol binario en pre-orden imprimiendo su contenido.
 - g. **inOrden()** Realiza un recorrido del árbol binario en in-orden imprimiendo su contenido.
 - h. **postOrden()** Realiza un recorrido del árbol binario en post-orden imprimiendo su contenido.
4. Probar la implementación de los métodos ejecutando el programa `PruebaArbolBinario`. Si el programa funciona adecuadamente se verán los siguientes mensajes:

```
----- Arbol Binario 1 -----
Tamano: 5
Altura: 3
Numero de hojas: 2
Numero de nodos internos: 2
```

Recorrido en pre-orden: A B C E G
Recorrido en in-orden: C B A E G
Recorrido en post-orden: C B G E A

----- Arbol Binario 2 -----

Tamano: 7

Altura: 3

Numero de hojas: 4

Numero de nodos internos: 2

Recorrido en pre-orden: Rosario Liliana Ernesto Nieves Juan Rosa
Luis

Recorrido en in-orden: Ernesto Liliana Nieves Rosario Rosa Juan
Luis

Recorrido en post-orden: Ernesto Nieves Liliana Rosa Luis Juan
Rosario