



Práctica 2A: Análisis teórico de los TADs AB y ABB

OBJETIVOS

- Diseñar, implementar y usar un TAD árbol.
- Comprender cómo se trabaja con árboles binarios (AB) y con árboles binarios de búsqueda (ABB).

ENUNCIADO

En clase de teoría hemos visto cómo trabajar con una clase que implementa un ABB. También hemos proporcionado dos TADs implementados: **TAD Árbol Binario (AB)** y **TAD Árbol Binario de Búsqueda (ABB)** (en un anexo del tema). Analice la implementación de los TADs anteriores y responda a las siguientes cuestiones:

TAD AB

- a.- ¿Está completa la especificación lógica del TAD AB?. Si considera que falta algo, reescriba de forma correcta la especificación del TAD. Razone su respuesta.
- b.- ¿Cuáles son las operaciones básicas que se pueden realizar con el TAD AB?

TAD ABB

- a.- ¿Está completa la especificación lógica del TAD ABB?. Si considera que falta algo, reescriba de forma correcta la especificación del TAD. Razone su respuesta.
- b.- ¿Cuáles son las operaciones básicas que se pueden realizar con el TAD ABB?

Práctica 2B: TAD AB

PARTE 1

En las clases de teoría hemos estudiado la implementación de las **clases NodoAB y AB**.

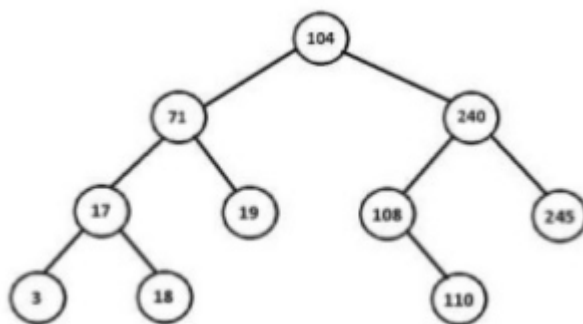
- 1.- Realice la especificación lógica completa del TAD AB.
- 2.- Realice la implementación del TAD AB teniendo en cuenta la especificación lógica anterior.
- 3.- Enriquezca el modelo Árbol Binario considerando la operación **EliminarNodosInferiores()** que se describe a continuación. (Actualice la especificación lógica, incluya la nueva operación **EliminarNodosInferiores()** en la interfaz del AB e implemente el método en la clase AB).



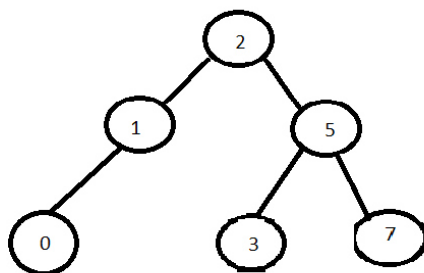
- 4.- Implemente en la clase **AB** un método recursivo **EliminarNodosInferiores()** que elimine los nodos de niveles inferiores a un nivel dado (nivel que el método recibe como parámetro).
- En la memoria de la práctica debe incluir una breve descripción del diseño del algoritmo (idea).
 - Realice la implementación del **método EliminarNodosInferiores()** genérico.

PARTE 2: TAD AB de números enteros.

- 1.- Diseñe una nueva clase llamada **ABEnteros** que represente un AB que almacena elementos de tipo Integer. Comente brevemente el proceso de diseño llevado a cabo.
- 2.- Escriba un método que cree el siguiente árbol AB1 (**CrearAB1()**):



- 3.- Escriba un método que cree el siguiente árbol AB2 (**CrearAB2()**):



- 4.- Implemente en la clase **ABEnteros** un método recursivo booleano **EsABB()** que devuelva true si el AB de claves enteras considerado es ABB y false en caso contrario.
- 5.- Implemente un método recursivo booleano **RaizIgualNodosInternos()** que trabaje sobre un AB de claves enteras y devuelva true si el valor de la clave contenida en el nodo raíz del AB es igual al número de nodos internos del árbol, en caso contrario, devolverá false.
- 6.- Implemente un método recursivo **MinimoValorNivel()** que calcule el valor mínimo que existe en un nivel dado (n) en un AB de claves enteras.
- Comente brevemente el diseño del algoritmo (idea).
 - Antes de invocar al método debe comprobar que existen al menos n niveles en el árbol. Para ello, implemente un método que calcule el número de niveles que tiene el



árbol **NivelesAB()**. El algoritmo debe contemplar la posibilidad de que no existan n niveles en el árbol en cuyo caso se lanzará un mensaje del tipo “ERROR: no existen 4 niveles en el árbol. El árbol tiene 3 niveles”.

- Tenga en cuenta que consideramos que el nivel del nodo raíz es el nivel 0.
- Realice la implementación del método **MinimoValorNivel()**.

7.- Para probar los métodos anteriores debe crear la **clase PruebaABEnteros** donde realizará las siguientes tareas:

- Crear AB1 y AB2 anteriores.
- Mostrar un listado en pantalla de las claves del AB1 en PreOrden
- Mostrar un listado en pantalla de las claves del AB2 en InOrden
- Comprobar si los árboles AB1 y AB2 son ABB y mostrar en pantalla el resultado obtenido en cada caso.
- Ejecutar el método RaizIgualNodosInternos() sobre el árbol AB1 y sobre el árbol AB2 y mostrar en pantalla el resultado obtenido en cada caso.
- Eliminar en el árbol AB1 los nodos que se encuentran por debajo del nivel 2. Después, realizar un listado de claves en InOrden de AB1.
- Eliminar en el árbol AB2 los nodos que se encuentran por debajo del nivel 3. Después, realizar un listado de claves en InOrden de AB2.
- Pedir al usuario un número entero (nivel n) e invocar al método MinimoValorNivel() mostrando finalmente el resultado obtenido. Los mensajes que se mostrarán por pantalla serán como los que aparecen en los siguientes ejemplos:

“Árbol AB1: El valor mínimo obtenido en el nivel 1 es 71”

“Árbol AB2: El valor mínimo obtenido en el nivel 2 es 0”

“ERROR: no existen 5 niveles en el árbol. El árbol tiene 3 niveles”.

** Para facilitar las pruebas, se proporcionan a continuación la secuencia de claves obtenidas al realizar los diferentes recorridos en profundidad solicitados sobre los árboles anteriores.

Árbol AB1

Recorrido en PreOrden : 104,71,17,3,18,19,240,108,110,245

Recorrido en InOrden : 3,17,18,71,19,104,108,110,240,245

Recorrido en PostOrden : 3,18,17,19,71,110,108,245,240,104 [no se pide]

Árbol AB2

Recorrido en InOrden : 0,1,2,3,5,7



PARTE 3

Si hubiésemos implementado los métodos de la práctica 2B en un ABB.. ¿En qué medida cambiaría el código fuente que ha desarrollado?.. ¿Por qué?.. Justifique de forma adecuada su respuesta.

NORMAS DE REALIZACIÓN DE LA PRÁCTICA:

- Elaborar una memoria (en papel) en la que se responda **de forma razonada** a todas las cuestiones que se plantean en la práctica. También deberá contener un listado de todo el código fuente desarrollado.
- Además, el código fuente deberá ser entregado a través del aula virtual teniendo en cuenta las instrucciones que en él se detallan.

FECHAS:

Publicación de la práctica: Martes, 2 de Mayo de 2017

Entrega de la práctica: Lunes, 29 de Mayo de 2017 – 23:00 h(Campus Virtual)

Nota: Los alumnos que cursan la asignatura desde Segovia, deberán entregar además la memoria impresa de la práctica al inicio del examen de teoría de PED que se celebrará el 30 de Mayo de 2017 a las 16 h.

**** Pasada esta fecha, NO se recogerán más prácticas.**