

Actividad 3. Implementación del TAD Árbol Binario

Objetivo

Implementar el TAD árbol binario haciendo uso de una estructura enlazada.

Procedimiento

1. Ver el video o leer la presentación sobre árboles binarios que están disponibles en Moodle, Tema 1/ Sección 1.2 TAD Árbol Binario/ Recursos didácticos.
2. Haciendo uso de una representación enlazada, resolver los ejercicios que se proponen en esta actividad.
3. Para probar el correcto funcionamiento de los métodos de ambos ejercicios se puede hacer uso del tests disponible en Moodle, Tema 1/ Sección 1.2 TAD Árbol Binario/ Actividades Grupo Reducido.

Evaluación

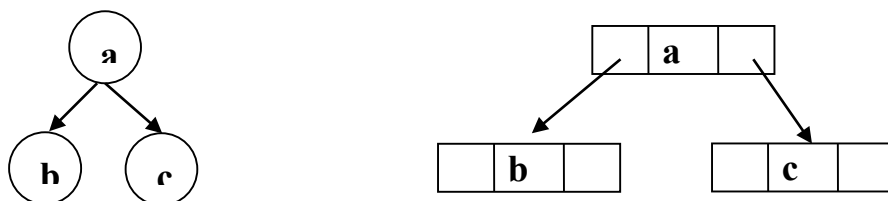
Estos contenidos serán evaluados mediante una prueba individual el 21 de octubre de 2022.

Tiempo estimado

5 horas

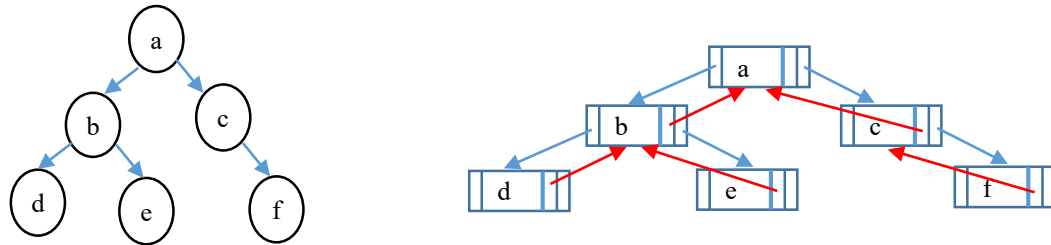
Ejercicio

1.- Una de las formas más habituales de implementar un árbol binario es mediante una estructura enlazada, donde cada nodo (NodoBinario<E>) está compuesto de un elemento de datos y dos enlaces, uno hacia el hijo izquierdo y otro hacia el hijo derecho.



Haciendo uso de la clase `NodoBinario<E>`, se pide crear un proyecto que implemente el TAD Árbol Binario<E>.

2.- Una modificación de la implementación de árboles binarios, que permite efectuar retrocesos sin necesidad de recursividad es manteniendo desde cada nodo un enlace a su padre. La ventaja de esta representación es que permite recorridos fáciles de forma no recursiva y permite obtener caminos ascendentes a partir de cualquier nodo.



ANEXO:

- **TAD Árbol Binario:**

```
public interface ArbolBinario<E>{
    public boolean esVacio();
    public E raiz() throws ArbolVacioExcepcion;
    public ArbolBinario<E> hijoIzq()throws ArbolVacioExcepcion;
    public ArbolBinario<E> hijoDer()throws ArbolVacioExcepcion;
    public boolean esta(E elemento);
    public void setRaiz(E elemRaiz) throws ArbolVacioExcepcion;
    public void setHijoIzq(ArbolBinario<E> hi) throws ArbolVacioExcepcion, NullPointerException;
    public void setHijoDer(ArbolBinario<E> hd) throws ArbolVacioExcepcion, NullPointerException;
    public void suprimir();
}
```