# *Estructuras de Datos y Algoritmos*

Año 2023

## **Práctico 4:** Árboles

Licenciatura en Ciencias de la Computación - Profesorado en Ciencias de la Computación (plan viejo

**Ejercicio 1:**

Dada la secuencia de números:

4, 19, -7, 49, 100, 0, 22, 12, -12, 25

**a)** Construir un árbol binario de búsqueda, insertando los elementos en la secuencia dada.

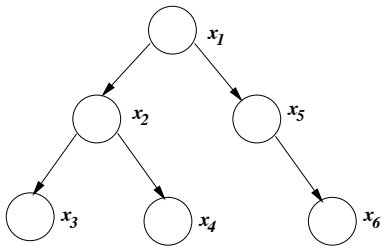
**b)** Insertar el elemento 10 en el árbol obtenido en el punto a).

**c)** Eliminar el elemento 19 del árbol obtenido en el punto b) (utilizar la política: menor de los mayores).

**d)** Insertar el elemento 1 en el árbol obtenido en el punto c).

**Ejercicio 2:**

Dados los elementos del conjunto:$\{x_1, x_2, x_3, x_4, x_5, x_6\}$ ¿Cuántas secuencias de entrada distintas, de todas las posibles secuencias, generarían el siguiente árbol? Mostrar tres de ellas.




**Ejercicio 3:**

Sabiendo que se quieren resolver servicios asociativos sobre una relación almacenada en un árbol binario de búsqueda (*ABB*), se pide graficar el *ABB* que se obtiene, para las relaciones R y S, al incorporar las nuplas en el orden dado por las secuencias.

**i)** $R \subseteq$ *Código-Postal* $\times$ *Ciudad* siendo el asociante *Código-Postal*.
Las nuplas son: *(4700, Catamarca), (3600, Formosa), (5500, Mendoza), (8300, Neuquén), (3400, Corrientes), (5400, San Juan), (4600, Jujuy), (3000, Santa Fe), (4000, Tucumán) , (4200, Santiago del Estero), (9410, Ushuaia), (8500, Viedma).*

**ii)** $S \subseteq$ *Patente* $\times$ *Marca* siendo el asociante *Patente*.
Las nuplas son: *(FUP980, Citroën), (EAX720, Alfa Romeo), (BGG675, Renault Scenic), (CDG890, Volkswagen), (AZT215, Chevrolet Zafira), (BGH287, Toyota), (PYU888, Chrysler), (CDG920, Ford), (PHI789, Peugoet 5004), (MWT567, Chevrolet Astra), (PNY367, Audi), (NRT378, Fiat).*

**Ejercicio 4:**

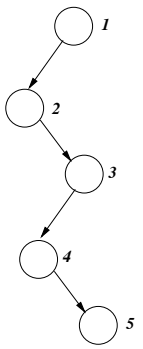
Para uno de los árboles obtenidos en el ejercicio anterior, eliminar los elementos en el mismo orden en el que fueron incorporados. Dejar en claro la política de reemplazo que utiliza.

**Ejercicio 5:**

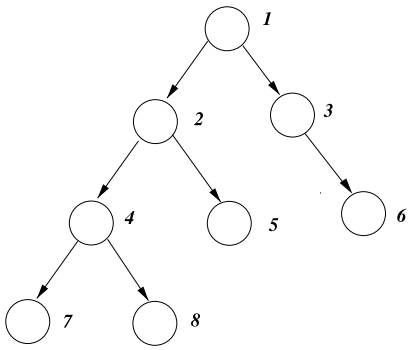
Para cada uno de los los siguientes árboles binarios, dar las secuencias de rótulos que se obtienen luego de realizar sobre ellos los barridos mencionados a continuación:

**a)** barrido pre-orden **b)** barrido post-orden **c)** barrido en orden **d)** barrido por niveles

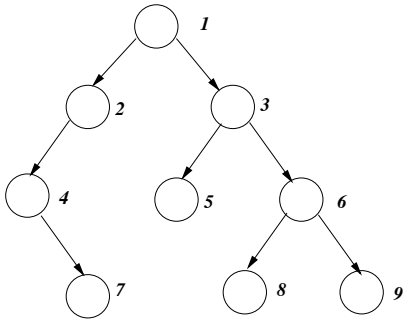
**1)**



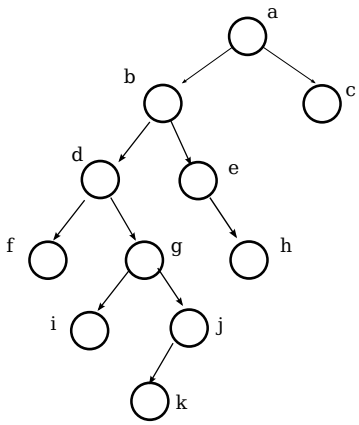

**2)**



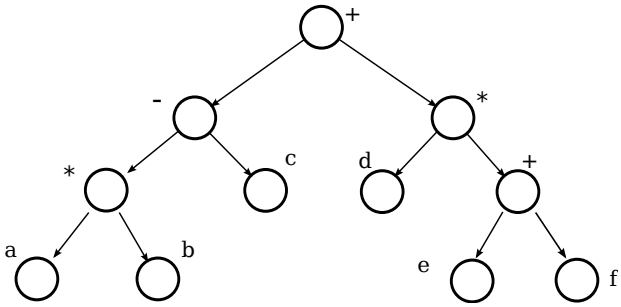

**3)**




**4)**




**5)**




**Ejercicio 6:**

Desarrollar los algoritmos ***iterativos*** para los distintos barridos de un árbol binario de búsqueda.

**Ejercicio 7:**

Sabiendo que el recorrido en *preorden* de un determinado árbol binario es: **GEAIBMCLDFKJH** y el recorrido en *inorden* es: **IABEGLDCFMKHJ**.

**a)** Dibujar el árbol binario que produjo esos barridos.

**b)** Dar el recorrido en postorden del mismo.

**Ejercicio 8:**

Considerando una relación $R \subseteq X \times Y$ almacenada en un árbol binario de búsqueda (*ABB*), se pide:

**a)** Desarrollar los operadores *LOCALIZACIÓN*, *EVOCACIÓN*, *ALTA*, *BAJA* y *MODIFICACIÓN*.

**b)** Desarrollar las evocaciones extremales NO destructivas: *Recuperar MÍNIMO* y *Recuperar MÁXIMO*.

**Ejercicio 9:**

Para cada uno de los árboles obtenidos en el **Ejercicio 3**, obtener el esfuerzo medio y máximo de localización exitosa y no exitosa **a posteriori**. Agregue las hipótesis que considere necesarias.

**Ejercicio 10:**

Diseñar la función:

```
int nodosCaminoMin(a:arbol)
```

que dado un árbol binario no vacío determine y devuelva el número de nodos existentes en el camino más corto desde la raíz a una hoja. ¿Cuál sería, en notación asintótica, el costo máximo a priori de este algoritmo, si se considera como función de costo: celdas consultadas?

**Ejercicio 11:**

Implementar una función booleana que, dados dos árboles binarios con igual cantidad y disposición de nodos; es decir, tienen la misma forma, responda *True* si el primero es menor al segundo y *False* en otro caso. *Diremos que un árbol binario A es menor a otro B, si los elementos de A son menores que los de B, en los nodos coincidentes en posición .*

**Ejercicio 12:**

Sea $X$ un conjunto de números reales, se define el *rango* del conjunto $X$ como *la diferencia entre el mayor y el menor valor de* $X$. Suponiendo que el conjunto $X$ se encuentra almacenado en un *ABB*, se pide:

**a)** Desarrollar en pseudo-código un algoritmo que permita calcular el rango de $X$.

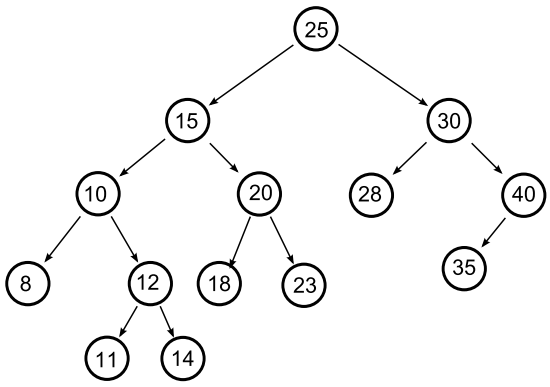
**b)** Obtener el esfuerzo máximo a priori, en notación asintótica, de realizar dicha operación considerando el algoritmo desarrollado por Ud. en a). Agregue las hipótesis que considere necesarias.

**Ejercicio 13:**

Sea el conjunto $X \subseteq \mathbb{N}$ almacenado en un *ABB* se pide:

**a)** Desarrollar, en pseudocódigo, una rutina que devuelva los elementos de $X$ almacenados en el *ABB* que caigan en el intervalo $[a, b]$.

**b)** Ejecute la rutina desarrollada en a) para los intervalos $[12, 30]$ y $[8, 20]$ sobre el siguiente *ABB*.



**Ejercicio 14:**

Dados $N$ elementos de un conjunto almacenado en una *LSD*, se pide desarrollar un algoritmo que realice su ordenamiento, de *mayor a menor*, utilizando como estructura auxiliar un *ABB* (*Treesort*). Suponga para ello que cuenta con todos los operadores básicos sobre las estructuras ya programados.

¿Cambiaría algo de lo programado por usted si se decidiera ordenar el conjunto de *menor a mayor*?

**Ejercicio 15:**

**a)** Dar el árbol de expresión para cada una de las siguientes expresiones:

**i)** $n - (\log y + \log x)$

**ii)** $b + (b^2 - 4 * a * c)/2 * a$

**iii)** $(\neg(p_1 \vee p_3) \Rightarrow (p_2 \wedge \neg p_2))$

**b)** Usando cada uno de los árboles obtenidos en **a)** obtener la notación *prefijo* y *posfijo* de cada una de las expresiones anteriores. ¿Qué barrido le permite obtener cada una?

**Ejercicio 16:**

Graficar los árboles balanceados en altura (*AVL*) que se obtienen al incorporar las nuplas de las relaciones $R$ y $S$ del **Ejercicio 3**, siguiendo las secuencias dadas en el mismo. Mostrar en cada paso las rotaciones realizadas, cuando corresponda, identificando cada una.

**Ejercicio 17:**

Eliminar los elementos de **uno** de los árboles del ejercicio anterior, utilizando:

**a)** orden FIFO (first in first out)

**b)** orden LIFO (last in first out)

Graficar paso a paso este proceso. Utilizar como política de reemplazo el *mayor de los menores*.

**Ejercicio 18:**

Suponiendo que $R \subseteq X \times Y$ está almacenada en un AVL, se pide:

**a)** Analizar los casos de *ALTA* y *BAJA* que pueden presentarse, dando un ejemplo para cada uno de ellos.

**b)** Desarrollar los operadores *ALTA* y *BAJA*.

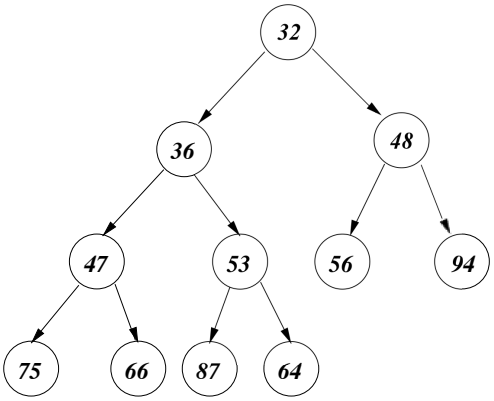
**Ejercicio 19:**

Dados los árboles casi llenos, que aparecen a continuación, se pide, para cada uno de ellos:

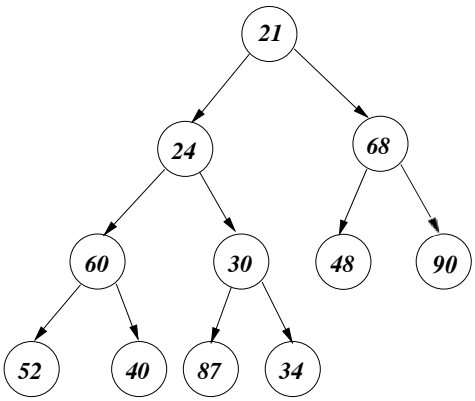
**a)** Verificar si es una parva de mínimos. En caso de no serlo, mostrar en qué posición/es no se cumple la condición de árbol parcialmente ordenado.

**b)** ¿Qué cambios se deberían realizar en esos árboles para que la estructura se transformara en una parva de mínimos?

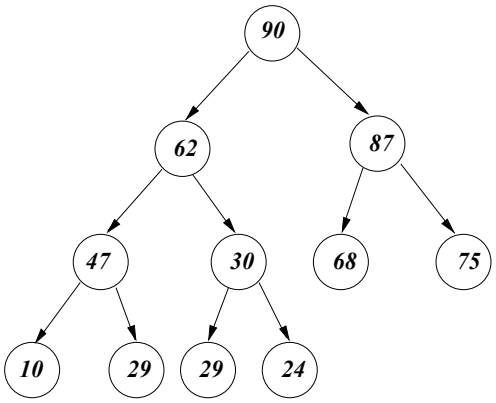
**c)** Realice los cambios sugeridos en el punto b), graficando las parvas resultantes.




**(1)**



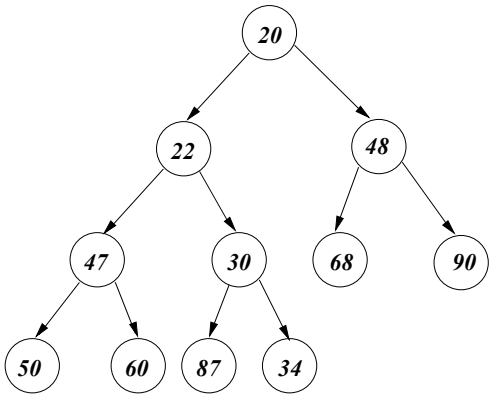

**(2)**




**(3)**

**Ejercicio 20:**

Dada la siguiente parva de mínimos:




1. Dar su representación secuencial
2. Hacer las siguientes operaciones en el orden dado, mostrando como queda la parva (en sus dos representaciones), después de cada operación:
   **a)** Eliminación del mínimo
   **b)** Inserción del elemento 24
   **c)** Inserción del elemento 10
   **d)** Eliminación del mínimo
   **e)** Eliminación del mínimo
   **f)** Cambiar el valor 47 por el 15
   **g)** Cambiar el valor 48 por el 95

**Ejercicio 21:**

Mostrar una parva de mínimos que contenga $N$ elementos distintos de manera tal que, luego de realizar la secuencia de operaciones listadas a continuación, la parva obtenida sea diferente de la inicial. Muestre la estructura inicial y cómo queda luego de cada operación. Puede seleccionar el valor de $N$ que más le convenga.

**a)** Evocación del mínimo, almacenándolo en una variable $m$.

**b)** Inserción del elemento guardado en $m$.

**Ejercicio 22:**

Suponga que tiene un arreglo con $N$ elementos, que representan las prioridades de un proceso. Cada elemento del arreglo es de la forma *(proceso, prioridad)*. Se pide, desarrollar en pseudo-código:

**a)** Una rutina que permita la creación de una parva de máximos, a partir de los elementos almacenados en el arreglo, utilizando el algoritmo de orden lineal.

**b)** Una rutina que permita realizar la evocación del máximo.

**c)** Una rutina que permita realizar el alta de un nuevo elemento.

**d)** ¿Para qué tipo de evocaciones, de las que usted conoce, es eficiente la parva?

**e)** Suponga que necesita modificar la prioridad de uno de los procesos almacenados en la misma:

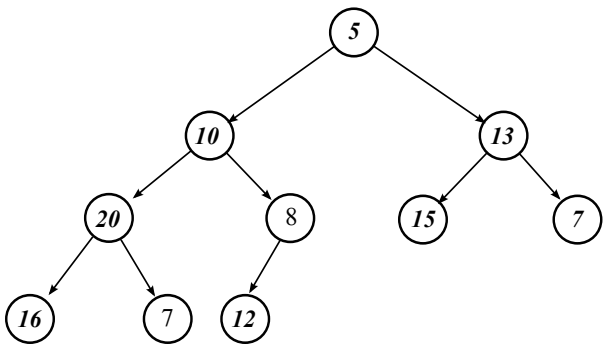
- **i)** Describa los pasos que debería seguir para hacerlo. ¿Tendría algún sentido utilizar alguna estructura adicional? Justifique.
- **ii)** ¿Cuál es la cota superior del esfuerzo de modificar la prioridad de un elemento de la parva, medido en cantidad de intercambios? ¿Este costo cambiaría si lo que se mide son las *celdas consultadas*?
- **iii)** Desarrollar una rutina que permita realizar la modificación de la prioridad de un proceso, reemplazándola por un valor *mayor* al almacenado.
- **iv)** Desarrollar una rutina que permita realizar la modificación de la prioridad de un proceso, reemplazándola por un valor *menor* al almacenado.
- **v)** Desarrollar una rutina que permita modificar la prioridad de un proceso en la parva, utilizando las rutinas desarrolladas en los ítems **iii)** y **iv)**.

**Nota:** Para los puntos **iii)**, **iv)** y **v)** suponga ya programada la rutina de localización.

**Ejercicio 23:**

a) Diseñar un algoritmo que, teniendo como entrada un árbol binario ordenado permita obtener una parva de mínimos. Aclarar las hipótesis que considere necesarias.

b) Ejecutar el algoritmo diseñado por usted sobre el siguiente árbol:



c) Si en lugar de tener como entrada un árbol binario ordenado, se tiene un $ABB$ ¿Se podría diseñar un algoritmo más eficiente que el anterior? Si su respuesta es si, explique qué haría este nuevo algoritmo; si su respuesta es no, justifique el por qué.

**Ejercicio 24:**

Desarrollar en pseudo-código una rutina que realice el ordenamiento de los elementos de un conjunto almacenado en un vector (devolviéndolos ordenados de menor a mayor) utilizando como estructura auxiliar una parva (*Heapsort*), Suponga que usa:

**a)** Una parva de mínimos.

**b)** Una parva de máximos.

y que cuenta con todas las rutinas que permiten el manejo de las estructuras ya programadas.
¿Puede hacerlo utilizando solamente el vector que recibe como entrada? Justifique su respuesta.

## Ejercicios Adicionales

**Ejercicio 1:**

Demostrar que todo árbol ternario tiene un número *impar* de hojas.

**Ejercicio 2:**

Sea $A$ un árbol ternario, se pide:

**a)** Definir recursivamente $i(A)$ y $e(A)$.

**b)** Definir recursivamente $I(A)$ y $E(A)$.

**c)** Encontrar y demostrar la relación que existe entre $i(A)$ y $e(A)$.

**d)** Encontrar y demostrar la relación que existe entre $I(A)$ y $E(A)$.

**Ejercicio 3:**

**a)** Encontrar las fórmulas que permitan obtener la posición de los hijos y la del padre (si los tiene) de un nodo, si éste está almacenado en la celda $i$ de la representación secuencial de una parva. Suponer que los elementos de la misma se encuentran a partir de la posición $0$.

**b)** Proponer una definición de una *parva ternaria*, análoga a una parva común, tal que en cada nodo el árbol puede tener a lo más 3 hijos. Encontrar las fórmulas para referenciar a los hijos y al padre del nodo *i* (si los tiene) sabiendo que:

1. la representación secuencial comienza en la posición 1.
2. la representación secuencial comienza en la posición 0.