

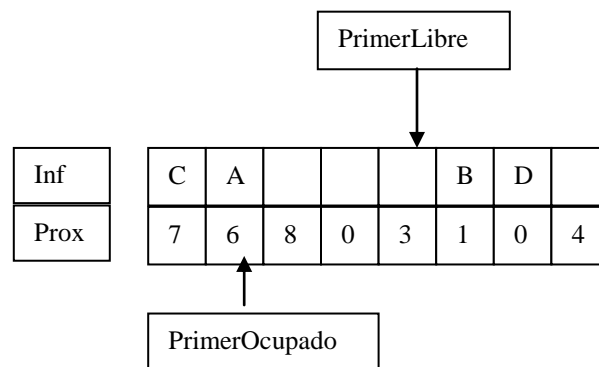
¿Cuántos árboles de altura mínima pueden formarse con n nodos distintos?

Codifique las primitivas de la clase lista en C++, utilizando *template*, e implemente sólo el método destructor. Si es necesario definir otros tipos, defínalos y justifique.

Realice un algoritmo que construya un árbol AVL (utilizando la representación dinámica de árboles AVL) a partir de un arreglo A que contiene n números reales distintos. No asuma ninguna condición inicial.

Considere una matriz $n \times n$ triangular superior ($a_{i,j}=0$ si $i > j$) de tipo base *ELEM*. Proponga una representación estática para almacenar sólo los elementos de la triangular superior, y defina la fórmula de acceso que indica dónde se encuentra el elemento $a_{i,j}$ ($1 \leq i \leq j \leq n$).

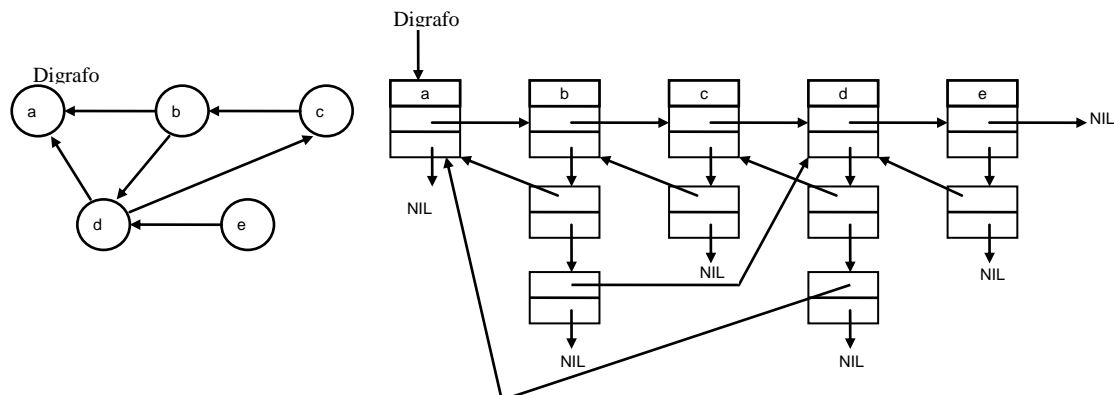
Implemente todas las primitivas de la lista con representación estática que se muestra a continuación. No deje de definir la estructura formalmente.



Realice el algoritmo de eliminación en un árbol de búsqueda.

Dada dos matrices dispersas A y B con la representación dinámica vista en clases, escriba una acción que realice el producto de ambas. No deje de definir la estructura formalmente.

Dada la siguiente ejemplificación gráfica de un digrafo:



1.1 Realice la definición formal de la estructura.

1.2 Realice un algoritmo que retorne todos los caminos existentes entre un par de nodos cualesquiera. Haga todas las validaciones necesarias.

Dada una expresión matemática “con paréntesis”, realice un algoritmo que la evalúe. Ejemplo: $-5/(2+8/2-4)*7$ da como resultado -17.5. Para ello, utilice la siguiente estructura para su algoritmo:

Tipo Elem= Registro
 En caso de T:(Número, Operador)
 Número: (valor: Real)
 Operador: (Op: (DIV, MULT, SUMA, RESTA,
 PAR_ABRE, PAR_CIERRA))
 FinCasos
 ProxElem: ↑Elem
 FinRegistro

Tipo Fórmula = ↑Elem

Nota: Tome en cuenta la prioridad de operadores de los lenguajes de programación, y la eficiencia.

Dado un arreglo con $m=2^n-1$ elementos ordenados ascendentemente, realice un algoritmo que genere un árbol AVL perfectamente balanceado con los m elementos y orden de complejidad $O(m)$.

Dado un bosque, hallar el árbol binario equivalente.

Nota:

- El acceso a las estructuras de datos debe ser totalmente correcta y clara para que el algoritmo tenga alguna puntuación.
- Duración de examen: 2 horas.
- Justifique todas sus respuestas con un análisis, defina las estructuras de datos necesarias, valide los datos, y tome en cuenta todos los casos para cada algoritmo.

Dado un arreglo A con $m=2^n-1$ elementos ordenados ascendentemente, realice un algoritmo que genere un árbol AVL perfectamente balanceado con los m elementos y orden de complejidad $O(m)$.

Escriba un programa completo en C++ que ordene un arreglo de n elementos enteros utilizando ordenamiento por mezclas (merge sort).

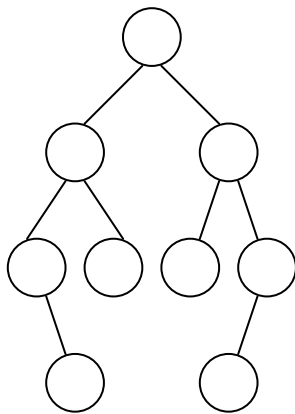
Utilizando únicamente las primitivas de la clase Pila, se quiere que Ud. desarrolle un procedimiento que dada una pila P, la invierta. No debe utilizar estructuras auxiliares.

Dada dos matrices dispersas A y B con la representación dinámica vista en clases, escriba una acción que realice el producto de ambas. No deje de definir la estructura formalmente. Maneje los apuntadores directamente, sin utilizar otros métodos.

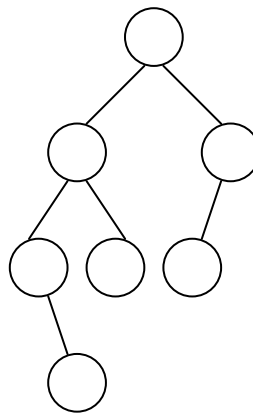
Imprimir todas las posibles combinaciones de números enteros positivos que sumen 100.

Escriba una función que verifique la simetría de un árbol binario cualquiera.

Ejemplos:



Este árbol es simétrico



Este árbol no es simétrico

Diseñe una estructura de datos que permita almacenar, acceder e interpretar fácilmente una expresión matemática, que contiene sólo números reales, paréntesis y operadores $+$, $-$, $*$, $/$. Realice además un procedimiento que tenga como entrada la expresión matemática, y como salida su evaluación.

Nota: Tome en cuenta la prioridad de operadores de los lenguajes de programación, y la eficiencia.

Ejemplo:

La evaluación de $(5+3.7/3.7)*7+1$ da como resultado 43

Considere una gran familia de individuos unidos por la relación $x R y \Leftrightarrow x$ es padre o madre de y .

- Diseñe una estructura de datos de manera tal que permita determinar fácilmente el padre y la madre de cada individuo, sin repetir información. Explique.
- Implemente las operaciones que usted considere para manipular esta estructura, y poder resolver 3.3 a través de estas primitivas
- Usando sólo las primitivas, realice un algoritmo eficiente tal que dado un individuo, imprima todos sus ascendientes (directos o no)

Defina formalmente los siguientes conceptos.

- Clase y objeto
- Estructura de datos
- Tipo de dato abstracto
- Complejidad en tiempo
- Complejidad en espacio
- Orden de complejidad

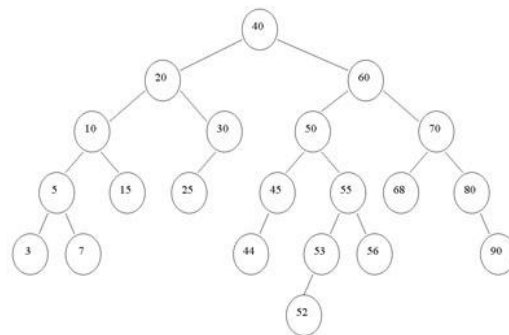
Realice una función que reciba como parámetro un árbol binario cualquiera y devuelva verdadero si el árbol está perfectamente balanceado, y falso en caso contrario.

Para cualquier árbol balanceado, ¿la diferencia de nivel entre cualquier par de hojas es a lo sumo uno?. Si es cierto, demuéstrela; sino, busque un contraejemplo.

Considere un árbol genealógico representado gráficamente mediante un árbol general, y cuya representación física corresponde a su árbol binario equivalente. Realice una operación que dado dos nombres N1 y N2, inserte N2 como descendiente de N1 en el árbol binario equivalente.

Muestre gráficamente cómo queda el árbol AVL dado, y que tipo de rotación se efectúa al realizar cada una de las siguientes líneas de operaciones en forma independiente (aplicándolas sobre el árbol original):

```
Insertar(A, 89);  
Insertar(A, 8);  
Eliminar(A, 7) y seguido a esto Eliminar(A, 20);  
Insertar(A, 0);
```



¿Cuáles de las siguientes secuencias no pudieron haber sido las secuencias de números revisados para insertar el número 363?

- a) 2,252,401,398,330,344,397,363.
- b) 924,220,911,244,898,258,362,363
- c) 925,202,911,240,912,245,363
- d) 2,399,387,219,266,382,381,278,363
- e) 935,278,347,621,299,392,358,363
- f) 0,1,2,3,4,5,6,7,8,9,10,11,5,12,13,14,6,15,16,17,18,363

Diseñe una estructura de datos que permita almacenar, acceder e interpretar fácilmente una expresión matemática, que contiene sólo números enteros, reales, paréntesis y operadores +, -, *, /. Realice además una función que tenga como entrada la expresión matemática, y como salida su evaluación.

Nota: Tome en cuenta la prioridad de operadores de los lenguajes de programación, y la eficiencia. Asuma que la expresión viene en un arreglo de caracteres de n casillas. Esta expresión debe ser ubicada en una estructura de datos adecuada para su posterior evaluación.

Ejemplo:

La evaluación de

(5	+	3	.	7	/	3	.	7)	+	1
---	---	---	---	---	---	---	---	---	---	---	---	---

Da como resultado 42.0

Dibuje un árbol AVL de manera tal que al insertar el elemento 24 se produzcan dos rotaciones dobles para su balanceo.

Demuestre que:

$$\begin{aligned}
 &4.1.- a \cdot \log_{10} n = O(\log_2 n) \\
 &4.2.- T(n) = \begin{cases} 1 & \text{si } n=1 \\ a \cdot T(n-c) + n & \text{si } n > 1 \end{cases} \quad \Rightarrow \quad T(n) = \begin{cases} O(n) & \text{si } a < 1 \\ O(n^2) & \text{si } a = 1 \\ O(a^{n/c}) & \text{si } a > 1 \end{cases}
 \end{aligned}$$

Utilizando únicamente las primitivas de la clase Pila, se quiere que Ud. desarrolle un procedimiento que dada una pila P, la invierta. No debe utilizar estructuras auxiliares

Utilizando únicamente las primitivas de la clase Pila, se quiere que Ud. desarrolle un procedimiento que dada una pila P, la invierta. No debe utilizar estructuras auxiliares.

Dada dos matrices dispersas A y B con la representación dinámica vista en clases, escriba una acción que realice el producto de ambas. No deje de definir la estructura formalmente. Maneje los apuntadores directamente, sin utilizar otros métodos.

Imprimir todas las posibles combinaciones de números enteros positivos que sumen 100.