

Práctica 1

Arboles Binarios

Ejercicio 1

a. Calcula las siguientes sumatorias:

$$\bullet \sum_{i=1}^5 7 =$$

$$\bullet \sum_{i=10}^{150} i =$$

$$\bullet \sum_{i=1}^n \sum_{j=1}^n n =$$

$$\bullet \sum_{i=1}^7 i =$$

$$\bullet \sum_{i=0}^{10} 2^{-i} =$$

$$\bullet \sum_{i=1}^n \left[i + \sum_{j=i}^n n - 1 \right] =$$

$$\bullet \sum_{j=1}^6 \frac{1}{j} =$$

$$\bullet \sum_{j=0}^n 2^j =$$

$$\bullet \sum_{i=1}^n \left[n + 2 \sum_{j=0}^n j \cdot \log_2(2^i) \right] =$$

b. Dadas las siguiente relaciones de recurrencia hallar la fórmula explícita:

$$\bullet a_1 = 1, \quad a_n = a_{n-1} + 2 \quad \text{para } n \geq 1$$

$$\bullet a_1 = 2, \quad a_n = 2 a_{n-1} \quad \text{para } n \geq 1$$

$$\bullet a_1 = 1, \quad a_n = a_{n-1} / 2 \quad \text{para } n \geq 1$$

$$\bullet a_1 = 1, \quad a_n = n + a_{n-1} \quad \text{para } n \geq 1$$

Ejercicio 2

- c. Considerando que un algoritmo requiere $f(n)$ operaciones para resolver un problema y la computadora procesa 1.024 operaciones por segundo y $f(n) = 2^n$, determine el tiempo en segundos requerido por el algoritmo para resolver un problema de tamaño $n=20$.
- d. Considerando que un algoritmo requiere $f(n)$ operaciones para resolver un problema y la computadora procesa 1.000 operaciones por segundo y $f(n) = n \log_{10} n$, determine el tiempo en segundos requerido por el algoritmo para resolver un problema de tamaño $n=100.000$.

Ejercicio 3

Suponga que Ud. tiene un algoritmo con $f(n)$ operaciones igual a $10n^2$. ¿En cuánto se hace más lento cuando el parámetro n aumenta:.....?

- i. El doble
- ii. El triple

Ejercicio 4

En base a la implementación de árboles binarios proporcionada por la cátedra, complete la misma agregando los siguientes métodos:

<code>incluye(elemento)</code>	# Retorna True si elemento está en el árbol, False en caso contrario
<code>inorden()</code>	# imprime un recorrido inorden del árbol binario
<code>preorden()</code>	# imprime un recorrido preorden del árbol binario
<code>postorden()</code>	# imprime un recorrido postorden del árbol binario

Pruebe la implementación de árbol binario, armando un árbol para luego imprimirlos con los todos recorridos implementados.

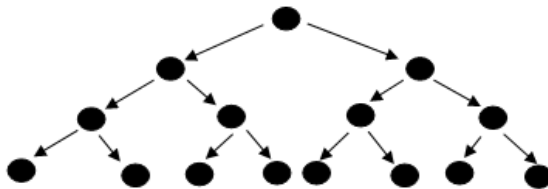
Ejercicio 5

Agregue a la clase `ArbolBinario` los siguientes métodos:

<code>contarHojas()</code>	#devuelve la cantidad de subárboles hojas del árbol receptor.
<code>entreNiveles(int n, m)</code>	# imprime el recorrido por niveles de los elementos del árbol receptor entre los niveles n y m (ambos inclusive). ($0 \leq n < m \leq \text{altura del árbol}$)

Ejercicio 6

Una red binaria es una red que posee una topología de árbol binario lleno. Por ejemplo:



Los nodos que conforman una red binaria llena tiene la particularidad de que todos ellos conocen cuál es su retardo de reenvío. El retardo de reenvío se define como el período comprendido entre que un nodo recibe un mensaje y lo reenvía a sus dos hijos.

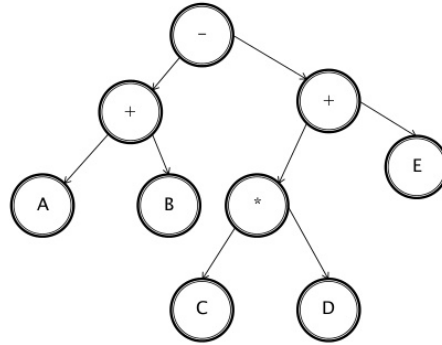
Su tarea es calcular el mayor retardo posible, en el camino que realiza un mensaje desde la raíz hasta llegar a las hojas en una red binaria llena. Cree una clase llamada **RedBinariaLlena** donde implementará lo solicitado en el método **retardoReenvio():int**.

Ejercicio 7

Implemente una clase llamada **ProfundidadDeArbolBinario** que tiene como variable de instancia un árbol binario de números enteros y un método de instancia **sumaElementosProfundidad(int p):int** el cuál devuelve la suma de todos los nodos del árbol que se encuentren a la profundidad pasada como argumento.

Ejercicio 8

Dado el siguiente árbol de expresión:



Indique las expresiones aritméticas resultantes de realizar una impresión en pre-orden, in-orden y post-orden.