

Nombre y Apellido: \_\_\_\_\_ Nro. C.I: \_\_\_\_\_ Firma:\_\_\_\_\_

- Indicaciones Generales:**
- Como parte de la evaluación se incluye el uso correcto del idioma, ortografía, orden y pulcritud.
  - No se permite el uso de ningún material de ayuda.
  - Puede utilizar calculadora científica.
  - Verifique que su celular se encuentre apagado. No puede utilizar su celular como calculadora.
  - Puede utilizar lápiz o bolígrafo con tinta de color azul o negro para completar los temas.
  - El examen tiene una duración de 180 minutos. Verifique que tenga 2 (dos) hojas numeradas del 1 al 2.
  - Coloque su nombre, nro. de cédula y firma en cada hoja del examen, los que no pueden estar escritos a lápiz.

**Tema 1 – 13 Puntos**

**1.1) (2p c/u)** Dados los siguientes fragmentos de código, analice asintóticamente cada fragmento en términos de O (o grande)

<pre>sum1 = 0 for (k = 1; k &lt;= n; k=k*2)   for (j=1; j&lt;=n; j++)     sum1++;</pre> <p><math>O(n \log n)</math>_____</p>	<pre>sum2 = 0 for (k = 1; k &lt;= n; k++)   for (j=1; j&lt;=k; j++)     sum2++;</pre> <p><math>O(n^2)</math>_____</p>	<pre>sum3 = 0 for (k = 1; k &lt;= n; k=k*2)   for (j=1; j&lt;=k; j++)     sum3++;</pre> <p><math>O(n)</math>_____</p>
--	---	---

**1.2) (3p)** Demuestre porqué  $6n^3 \neq O(n^2)$

Aplicar la definición

**1.3) (3p)** Indique el costo asintótico en términos de O (o grande) para un algoritmo cuyo análisis arrojó que la T(n) encontrada es:

$$T(n) = \log_2(n!) + (100 * n) - \log_2^2(n)$$

El resultado final es

$O(n \log n)$

**1.4) (1p)** Ordene las siguientes funciones de acuerdo a su tasa de crecimiento, de mayor a menor

$n^2, n \log_2 n, n, n \log_2 \log_2 n, 44, n \log_2^2 n, n \log(n^2), 2^n, n^3, \frac{7}{n}, n^5, n^2 \log(n^2), 1$

La respuesta es

$2^n, n^5, n^3, n^2 \log(n^2), n^2, n \log_2^2 n, n \log_2(n^2), n \log_2 n, n \log_2 \log_2 n, n, 44, 1, \frac{7}{n}$

**Tema 2 – 6 Puntos**

**2.1 (4p)** Muestre paso a paso el resultado de insertar los elementos 2, 1, 4, 5, 9, 3, 6, 7 y 8 en un AVL inicialmente vacío. Cada vez que ocurra una rotación, indique el tipo de rotación que ocurre.

**2.2 (2p)** Imprima el resultado obtenido al aplicar los recorridos: pre-orden, post-orden en el árbol AVL generado en el item previo.

**Tema 3 – 10 Puntos**

**3.1 (4p)** Utilizando una tabla hash cerrada con 13 slots(o cubetas), con doble hashing para resolución de conflictos, inserte las siguientes claves: 2, 8, 31, 20, 19, 18, 53, 27. Las funciones hash son:

$H1(k) = k \bmod 13$   
 $H2(k) = (Rev(k+1) \bmod 11)$

Donde Rev(k) invierte los dígitos decimales de k. Por ejemplo, Rev(37) = 73; Rev(7) = 7

**3.2 (6p)** Dada la siguiente entrada 371, 323, 173, 199, 344, 679, 989, 661, una tabla hash de tamaño 11 y la función hash  $H(X) = X \bmod 11$ , muestre:

- a) La tabla hash resultante con exploración lineal.
- b) La tabla hash resultante con exploración cuadrática sin rehashing.
- c) La tabla hash resultante con exploración cuadrática y con rehashing considerando un factor de carga de 0.5.

**Tema 4 – 8 Puntos**

**4.1 (4p)** Un promotor de la empresa Dell ofrece su último producto de la línea de servidores y menciona que el mismo ejecuta instrucciones 100 veces más rápido que su principal competidor, la empresa HP. Si en el servidor HP puede ejecutar un programa de tamaño  $n$  en 1 hora, cuál es el tamaño de entrada que puede procesar el servidor Dell, en el mismo tiempo, para cada algoritmo cuyas tasas de crecimiento de su tiempo de ejecución son:  $n, n^2, n^3, 2^n$

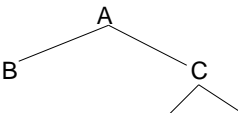
**4.2 (4p)** Considerando a  $n$  como una potencia de 2, y utilizando inducción matemática, muestre que la siguiente recurrencia:

$$T(n) = \begin{cases} 2 & \text{si } n=2 \\ 2T(n/2)+n & \text{si } n=2^k, \text{ para } k > 1 \end{cases}$$

Es  $T(n)=n \log_2(n)$

**Tema 5 – 5 Puntos**

Considerando el siguiente árbol, marque con V o F las afirmaciones siguientes para indicar si las mismas son verdaderas o falsas. En cada caso, fundamente su respuesta.



Nombre y Apellido: \_\_\_\_\_

Nro. C.I: \_\_\_\_\_

Firma: \_\_\_\_\_

D

E

- 5.1 (1p) El recorrido en orden de los nodos generó la salida A B C D E.

Respuesta: (\_\_\_\_)

Fundamento: \_\_\_\_\_
- 5.2 (1p) La altura del árbol es 2.

Respuesta: (\_\_\_\_)

Fundamento: \_\_\_\_\_
- 5.3 (1p) El árbol es AVL.

Respuesta: (\_\_\_\_)

Fundamento: \_\_\_\_\_
- 5.4 (1p) El árbol es completo.

Respuesta: (\_\_\_\_)

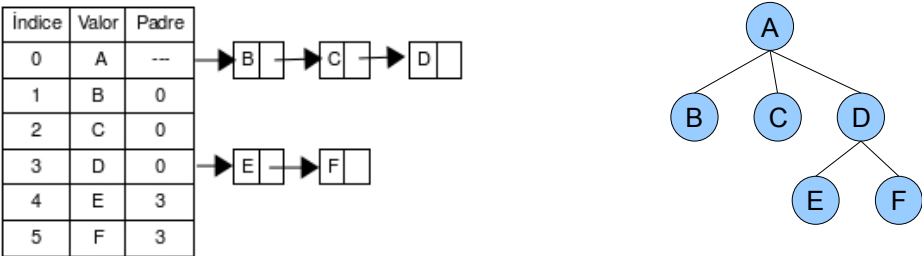
Fundamento: \_\_\_\_\_
- 5.5 (1p) Los descendientes de A son C, D y E.

Respuesta: (\_\_\_\_)

Fundamento: \_\_\_\_\_

Tema 6 – 10 Puntos

6.1 (4p) Una posible implementación de un árbol general consiste en el esquema conocido como Lista de Hijos. En este enfoque, los nodos se almacenan en un arreglo y en cada posición un nodo posee además de su valor, una referencia al padre y una lista enlazada de todos sus nodos hijos, en orden de izquierda a derecha. Considerando esta implementación, escriba una función que reciba 2 árboles generales y retorne TRUE o FALSE si es que los árboles son o no iguales. Se define que un árbol es igual al otro si tiene la misma estructura de nodos y los valores de cada uno de ellos son iguales. Abajo se muestra un ejemplo.



Representación Lista de hijos del árbol de la derecha

- (1p) Calcule el tiempo de ejecución de su algoritmo en términos de O (o grande)
- 6.2 (4p) Un problema común para los compiladores consiste en determinar si los símbolos de agrupamiento usados en una expresión (paréntesis, corchetes y llaves) están correctamente balanceados y anidados de manera apropiada. Por ejemplo: { (2+3-[4\*2]) / 2 } es una expresión cuyos símbolos están balanceados y anidados, mientras que una expresión incorrecta es: (2+3]-4)). Proponga un algoritmo capaz de resolver este problema, que retorne TRUE si es que la expresión es correcta y FALSE si es incorrecta. Indique claramente las estructuras de datos que utilizará para resolver el problema.

- (1p) Calcule el tiempo de ejecución de su algoritmo en términos de O (o grande)

Tema 7 – 8 Puntos

Marque la respuesta correcta, existe una sola. Observaciones: +1p por cada ítem marcado y correcto; -1p por cada ítem marcado e incorrecto; -2p por cada ítem no marcado.

- 7.1) Al recorrer un árbol, en cuál de los siguientes recorridos se procesa el nodo luego de recorrer recursivamente todos sus hijos?  
a) En orden      b) Por Niveles      c) Post orden      d) Pre orden      e) ninguna de las anteriores
- 7.2) ¿Cuál es la estructura de datos cuyas características la hacen ideal para implementar la tabla de símbolos de un compilador?  
a) BST      b) AVL      c) Tabla Hash      d) Cola      e) ninguna de las anteriores.
- 7.3) Un algoritmo toma 5 segundos para un tamaño N de 100. Si el algoritmo es cuadrático, aproximadamente cuanto tiempo tomará para resolver un problema de tamaño 200  
a) 7 segundos      b) 10 segundos      c) 15 segundos      d) 25 segundos      e) ninguna de las anteriores
- 7.4) La habilidad de usar un tipo referencia para referenciar a objetos de diferentes tipos durante la ejecución es:  
a) Composición      b) Encapsulamiento      c) Ocultamiento de información  
d) Polimorfismo      e) Enlace estático
- 7.5)Cuál de las siguientes afirmaciones no es verdadera acerca de los Strings en Java?  
a) Son tipos referencia      b) Puede accederse a caracteres individuales      c) Deben crearse con el operador new  
d) Su valor es inmutable      e) Siempre puede determinarse la longitud
- 7.6) Suponga  $T1(N) = O(F(N))$  y  $T2(N) = O(F(N))$ .Cuál de las siguientes afirmaciones es verdadera?  
a)  $T1(N) + T2(N) = O(F(N))$       b)  $T1(N) \times T2(N) = O(F(N))$       c)  $T1(N) / T2(N) = O(1)$   
d)  $T1(N) = O(T2(N))$       e) Ninguna de las anteriores
- 7.7) El algoritmo de Strassen para multiplicación de dos matrices de tamaño NxN, realiza siete(7) llamadas recursivas para multiplicar dos

Nombre y Apellido: \_\_\_\_\_ Nro. C.I.: \_\_\_\_\_ Firma: \_\_\_\_\_

matrices de  $N/2 \times N/2$ . El tiempo adicional que se suma a las llamadas recursivas es cuadrático. Entonces, la solución se encuentra en:

- a)  $O(N^2)$       b)  $O(N^3)$       c)  $O(N^{2.81})$       d)  $O(N^{2.5})$       e) ninguna de las anteriores.

**7.8)** Dado un nodo X en un árbol representado mediante la técnica del primer hijo, siguiente hermano. ¿Cuál será el tiempo de ejecución en el peor caso, de un algoritmo que permita encontrar el padre de X, donde H es la altura del árbol con N nodos?

- a)  $O(1)$       b)  $O(H)$       c)  $O(N)$       d)  $O(\log H)$       e) Ninguna