Tarea Nº3

**Programación en C: Árboles Binarios Ordenados.**

**I. Objetivos.**

El presente enunciado tiene como objetivo: Diseñar e implementar operaciones sobre árboles binarios ordenados de números enteros para el manejo interno de información.

**II. Interfaz.**

La interfaz solicitada para esta tarea se ejemplifica en la figura 1.a y el archivo de entrada usado se muestra en la figura 1.b. Como puedes observar se trata de un archivo con valores enteros. En esta tarea se espera que aparezca la información del árbol binario ordenado y que luego de **pedirle al usuario** un elemento cualquiera del árbol, aparezca la información del nodo que contiene a tal elemento.

En las figuras 2 se muestra la interfaz solicitada al momento de analizar el elemento que está en la raíz del árbol (usando el mismo archivo 1.txt). En la figura 3 puedes observar la interfaz que se espera si el usuario pide analizar un elemento que corresponde a una hoja.

Si el árbol generado posee una altura superior a 10, no se imprimirá (figura 4.a). En la figura 4.b se presenta el archivo de entrada usado en este ejemplo.

|  |  |
| --- | --- |
|  |  |
| **(a)** | **(b)** |

**Figura 1.** **(a)** Ejemplo de la interfaz de esta tarea. **(b)** ejemplo del archivo de entrada.

|  |
| --- |
|  |

**Figura 2.** Ejemplo de la interfaz de esta tarea si el usuario solicita analizar la raíz del árbol.

|  |
| --- |
|  |

**Figura 3.** Ejemplo de la interfaz de esta tarea si el usuario solicita analizar una hoja del árbol.

|  |
| --- |
|  |
| **(a)** |
|  |
| **(b)** |

**Figura 4.** Ejemplo de la interfaz de esta tarea para un árbol con más de 10 niveles. **(b)** ejemplo del archivo de entrada.

**III. Actividades de los talleres y autónomo.**

Para el desarrollo de esta tarea se te entregará el **ejecutable** (**Tarea3 - SOLUCIÓN.exe**) para que puedas verificar los resultados que obtengas en la generación de las diferentes funciones solicitadas, tanto en tu trabajo dentro del taller, como en forma autónoma.

Además, el trabajo en taller se realizará a partir de un código fuente llamado **Tarea3.c**. que se te entregará. Podrás observar en ese código fuente que **cada función que deberás construir está comentada tanto en las entradas, salida y el proceso que se desea que realice**.

En la **primera clase de taller** asociado a esta tarea desarrollarás junto a tu profesor las siguientes funciones: insertaABO, estaEnABO, peso y altura.

En la **segunda clase de taller** asociado a esta tarea desarrollarás junto a tu profesor las siguientes funciones: menorABO, nivelDeUnNodoABO, antecesoresNodoABO y DescendientesNodoABO.

**Autoestudio:**

* Para tu autoestudio deberás desarrollar el resto de las funciones que no se realicen en taller.
* Recuerda que en tus clases de cátedra se te han entregado diferentes algoritmos que te pueden servir para ejercitar tanto para la **evaluación de esta tarea como para la Solemne N°3.**
* **Recuerda contactar a tu profesor de taller por cualquier duda que tengas en este proceso.**

**IV. Sobre Buenas Prácticas de Programación.**

1. Debes usar identificadores representativos para tus constantes, variables, parámetros de entrada y funciones.
2. Las variables locales a cada función **debes** definirlas al **principio** de la función. Puedes darle valores iniciales al momento de definirlas (esto se llama “inicializarlas”).
3. Tu código debe estar correctamente *indentado* (uso de sangrías para cada sub-bloque de instrucciones), esto incluye el correcto alineamiento de las llaves (“**{**” y “**}**”) que delimitan tales bloques.
4. Tu código no puede presentar más de 1 línea en blanco.
5. Tu código no puede poseer instrucciones “basura”.
6. Debes comentar cada una de las funciones que definas como se indicó en la tarea 0.

**IV. Sobre la Evaluación.**

La semana del **lunes 27-noviembre** será la evaluación de esta tarea, en el día y horario que te corresponda el taller del ramo. Al igual que las tareas anteriores, la evaluación se realizará manualmente.