



Universidad Autónoma de Zacatecas

Unidad Académica de Ingeniería Eléctrica

Programa Académico de Ingeniería de Software

Práctica 24

Datos generales:

Nombre de la Práctica	Árboles
Nombre de la carrera	Ingeniería de Software
Nombre de la materia	Laboratorio de Estructuras de Datos
Número y nombre de Unidad(es) temática(s)	IV. Estructuras no lineales.
Docente que imparte la materia	Aldonso Becerra Sánchez
Fecha de entrega para los alumnos	18-octubre-2021
Fecha de entrega con extensión y penalización	19-octubre-2021
Fecha de elaboración:	18-octubre-2021

Objetivo de la Práctica	Practicar con las operaciones sobre árboles binarios
Tiempo aproximado de realización	2.5 horas
Introducción	Los árboles binarios son un tipo especial de árbol que permite realizar de forma simple y eficiente las operaciones sobre él, a diferencia de un árbol general.

Referencias que debe consultar el alumno (si se requieren):

Referencia 1:

1. Cairo, Osvaldo; Guardati, Silvia. Estructura de Datos, Tercera Edición. McGraw-Hill, México, Tercera Edición, 2006.



Universidad Autónoma de Zacatecas

Unidad Académica de Ingeniería Eléctrica

Programa Académico de Ingeniería de Software

Referencia 2:

2. Mark Allen Weiss. Estructura de datos en Java. Ed. Addison Wesley.

Referencia 3:

3. Joyanes Aguilar, Luis. Fundamentos de Programación. Algoritmos y Estructuras de Datos. Tercera Edición, 2003. McGraw – Hill.

Actividades que debe realizar el alumno:

Actividad inicial:

Generar el reporte en formato IDC.

Actividad 1:

Primero genere la **Introducción**.

Actividad 2:

Genere un programa que cree un árbol binario con expresiones como:

$x + (\text{resta} / 4) * y ^ 2$

NOTA: Use el método visto en clase para crear un árbol manualmente e introduzca esta expresión manualmente en el árbol con el método recursivo usado en clase (crearArbol()). Recuerde que el programa del árbol le va preguntando de uno por uno los padres e hijos, usted debe valores como considere, el ejemplo de arriba solo es un ejemplo particular. El programa no va a leer una cadena aritmética y la va a separar, no se confunda.

1. Posteriormente extraiga en una lista enlazada todos los operadores (recuerde los operadores son las raíces de cada subárbol); así como en una lista enlazada hash todos los operandos (utilice su agrupador de operandos de práctica pasadas), los cuales tendrán valores substituidos por el usuario para cada variable.

Lista 1: x (5) -> resta (7) -> 4 (4)-> y (7)-> 2 (2) -> null

Lista 2: + -> / -> * -> ^ -> null

Después imprima las dos listas enlazadas.

2. Genere automáticamente otro árbol a partir del primero, de tal manera que permita suplantarse las variables como “x”, “resta” y cualquier otra por un valor introducido por el usuario. Muestre los dos árboles con los tres recorridos vistos



Universidad Autónoma de Zacatecas

Unidad Académica de Ingeniería Eléctrica

Programa Académico de Ingeniería de Software

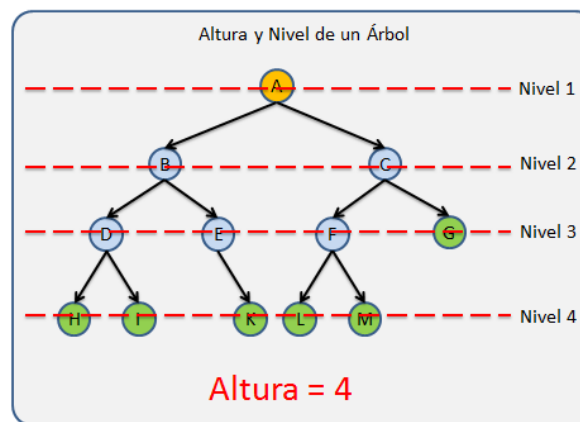
anteriormente (preorden, postorden, inorden). Nota: en este método el usuario no intervendrá en la generación del árbol, solo en la sustitución de valores.

Haga el programa (actividad 2, la cual es el **Desarrollo** del programa, junto con la captura de pantalla del programa funcionando).

Actividad 3:

Haga un método recursivo en la clase ArbolBinario que permita obtener:

- La altura del árbol.
- El nivel en el que se encuentra algún nodo proporcionado por el usuario. Por ejemplo, si se da el nodo B, la altura de B debe ser 2.
- Cuántos elementos existen en cada nivel presente en el árbol: 1 en nivel 1, 2 en nivel 2, 4 en nivel 3 y 5 en nivel 4.
- Que indique, a petición del usuario, si un nodo en particular es raíz, interior u hoja, además de indicar si tiene padre y cuál es. Por ejemplo, el usuario introduce B, la respuesta del programa debe ser “nodo intermedio” y su padre es A.



Recuerde que los métodos deben ser naturalmente recursivos.

Haga el programa (actividad 3, la cual es el **Desarrollo** del programa, junto con la captura de pantalla del programa funcionando).

Actividad 4:

Pruebe el funcionamiento del programa de las actividades con todo y sus capturas de pantalla.



Universidad Autónoma de Zacatecas

Unidad Académica de Ingeniería Eléctrica

Programa Académico de Ingeniería de Software

Actividad 5:

Realice la sección de **Código agregado** (diagrama de clases UML).

Actividad 6:

Realice la sección de **Pre-evaluación** (use los lineamientos establecidos).

Actividad 7:

Finalmente haga las **Conclusiones**.

Actividad 8:

Enviar en <http://ingsoftware.reduaz.mx/moodle>

Archivo anexo que se requiere para esta tarea (opcional):

Dudas o comentarios: a7donso@gmail.com

Archivo anexo que se requiere para esta tarea (opcional):

Dudas o comentarios: a7donso@hotmail.com