

INSTITUTO TECNOLÓGICO DE NUEVO LEÓN



Ing. Sistemas computacionales

"Diseño de aplicaciones móviles"

Profesor: Juan Pablo Rosas Baldazo

Tarea: "Lenguajes Autómatas 2"

Unidad 1

Presenta.

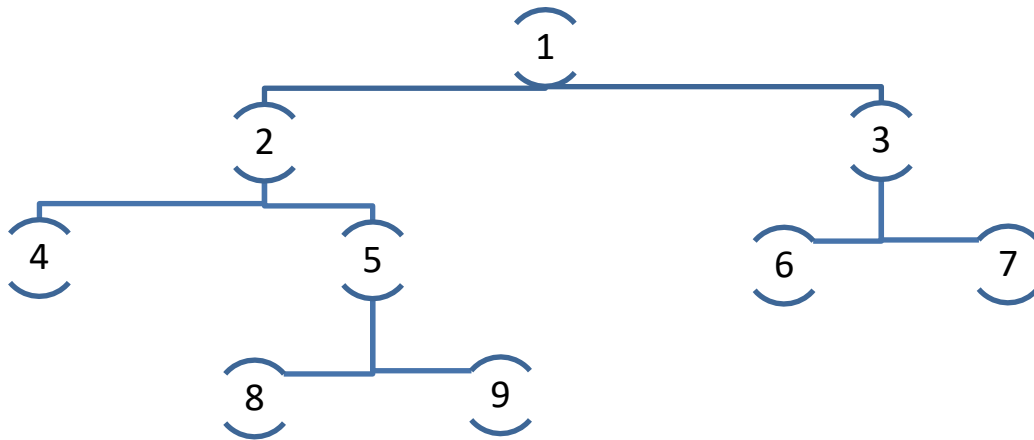
Mario Humberto Uriegas de León

No. De control: 14480514

Cd. Guadalupe; Nuevo León; a 15 de Febrero de 2018

## Introducción

En este documento se mostrara como se elaboró un programa basado en un árbol binario, que hace y que referencias o guías fueron necesarias para llevarlo a cabo.



## Descripción

Se hizo un programa de un árbol binario que imprimiera el postorden, inorden y preorden del mismo, este trabajo se pudo realizar gracias a este video tutorial:

([https://www.youtube.com/watch?v=h\\_8WTKH1OV4](https://www.youtube.com/watch?v=h_8WTKH1OV4)) en él se muestran la realización de un árbol binario y la explicación de sus diferentes métodos y la manera sencillas en que pudieron ser usados.

### Clase nodos:

```
clase nodos {  
    se crea la cadena dato;  
    se crea un nodo izquierdo izq;  
    se crea un nodo derecho der;  
  
    el nodo debe tener un dato  
    public nodos(String dato){  
        this.dato = dato;  
    }  
    Obtener el nodo izquierdo  
    public nodos getNodoIzquierdo(){  
        return izq;  
    }  
}
```

Obtener el nodo derecho

```
public nodos getNodoDerecho(){  
    return der;  
}
```

Asignación de nodo izquierdo

```
public void setNodoIzquierdo(nodos nodo){  
    izq = nodo;  
}
```

Asignación de nodo derecho

```
public void setNodoDerecho(nodos nodo){  
    der = nodo;  
}
```

Obtener el dato

```
public String getDato(){  
    return dato;  
}  
}
```

### **Clase árbol**

Se crea la clase arbol {

Se crean los nodos con sus respectivos valores 1,2,3,4,5,6,7,8,9

El nodo principal es ("1")

El nodo raíz izquierdo es ("2")

El nodo raíz derecho es ("3")

El nodo derecho del 2 es ("5")

El nodo izquierdo de 2 es ("4")

El nodo derecho de 2 es ("5")

El nodo izquierdo de 3 es ("6")

El nodo derecho de 3 es ("7")

El nodo izquierdo de 5 es ("8")

nodo5.setNodoDerecho ("9")

imprime el ("Preorden")

imprime el ("PostOrden")

Imprime el ("InOrden")

}

preOrden(nodos raiz){

si es diferente a nulo

imprime el dato de la raíz

obtén el nodo izquierdo y asigna su valor a la raiz

```

        obtén su nodo derecho y asigna el valor a la raíz
    }
}

postOrden(nodos raiz){
    si es diferente a nulo
        obtén el nodo izquierdo y asigna su valor a la raíz
        obtén su nodo derecho y asigna el valor a la raíz
        imprime el dato de la raíz
    }
}

inOrden(nodos raiz){
    si es diferente a nulo
        obtén el nodo izquierdo y asigna su valor a la raíz
        imprime el dato de la raíz
        obtén su nodo derecho y asigna el valor a la raíz
    }
}
}

```

### Resultado

Los recorridos son:

Preorden: 1-2-4-5-3-6-7

Postorden: 4-5-2-6-7-3-1

Inorden: 4-2-5-1-6-3-7

### Conclusion

Se pueden apreciar los recorridos de Preorden, post orden e in orden pero falta que resuelva las cadenas de operaciones largas para saber la jerarquía en que se deben resolver.

### Referencias

Patricio Perez Pinto. (2013). Arbol binario en Java. 12/02/18, de Youtube Sitio web:  
[https://www.youtube.com/watch?v=h\\_8WTkH1OV4](https://www.youtube.com/watch?v=h_8WTkH1OV4)