



INSTITUTO TECNOLÓGICO DE NUEVO LEÓN

Ingeniería en sistemas computacionales

Lenguaje y autómatas 2

Proyecto 1

Catedrático

Juan Pablo Rosas Baldazo

Alumno

Rafael Salazar Rodríguez 14480474

16/02/2018

Introducción.

A continuación se presentara un árbol de expresiones en java de forma que imprima los recorridos preOrden, inOrden y postOrden. Básicamente me fue imposible realizar este ejercicio de forma que se capturara mediante un scanner pero aun así a pesar de ellos se realizó de otra forma (video), existía una incompatibilidad entre las clases y el funcionamiento del scanner.

La cadena a resolver será: $3 + (8 * 5) + e / 2$

Pseudocodigo.

Clase Nodo.java

Clase Nodo {

Se crea una cadena llamada "dato"

Se crea un nodo izquierdo y uno derecho

El nodo siempre debe tener un dato en sus parámetros

obtenerNodoIzquierdo(){

return izq;

}

obtenerNodoDerecho(){

return der;

}

asignarNodoIzquierdo(Nodo nodo){

izq = nodo;

}

asignarNodoDerecho(Nodo nodo){

der = nodo;

}

obtenerDato(){

return dato;

}

}

Clase Arbol.java

Clase Arbol {

Método principal{

Creamos los nodos con operadores.

La raíz es +

El nodo 2,3 y 4 serán +, / y *

En la raíz su nodo izquierdo es un "3"

En la raíz su nodo derecho es el nodo 2

En el nodo 2 su nodo derecho es una "e"

En el nodo 2 su nodo izquierdo es el nodo 3

En el nodo 3 su nodo derecho es un "2"

En el nodo 3 su nodo izquierdo es el nodo 4

En el nodo 4 su nodo derecho es una "8"

En el nodo 4 su nodo izquierdo es una "5"

Imprime PreOrden

Imprime PostOrden

Imprime InOrden

}

PreOrden {

Si es diferente de nulo {

Imprime el dato de la raíz

Obtén su nodo izquierdo y asigna el valor a la raíz

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

}

PostOrden {

Si es diferente de nulo {

Obtén su nodo izquierdo y asigna el valor a la raíz

Obtén su nodo derecho y asigna el valor a la raíz

Imprime el dato de la raíz

}

}

InOrden {

Si es diferente de nulo {

Obtén su nodo izquierdo y asigna el valor a la raíz

Imprime el dato de la raíz

Obtén su nodo derecho y asigna el valor a la raíz

}

}

}

Resultados.

Los recorridos del árbol resultaron los siguientes:

- PreOrden: +3+/*852e
- PostOrden: 385*2/e++
- inOrden: 3+8*5/2+e

Conclusión.

Desafortunadamente no logre resolver la situación del scanner y la compatibilidad para el funcionamiento del programa pero respecto al modelo de la jerarquía y los recorridos si funcionaron de acuerdo a lo establecido en el video de referencia.

Referencia.

El siguiente tutorial fue la única información que realmente me sirvió y el cual sirvió para basarme en el código.

https://www.youtube.com/watch?v=h_8WtkH1OV4