Universidad Católica San Pablo

Compiladores

Práctica 7 – Transformar análisis sintáctico a árbol sintáctico

Objetivos:

- Transformar el análisis sintáctico arrojado por nuestro parser LL(1) a la forma de un árbol sintáctico.

1. Crear nuestro árbol sintáctico

Lo recomendable será crear un árbol multicamino con referencia al padre y a su hermano a la derecha.

```
class Nodo:
    etiqueta = ''
    hijos = list()  # lista de referencias a los nodos hijos
    padre = None
    siguiente = None
```

2. Crear nuestro árbol sintáctico estrategia con tabla

Crear una tabla de la siguiente forma:

Pila	Entrada	Operación	Generado

- Los valores de pila y entrada están definidos en la práctica 5.
- Los valores de operación pueden ser (1) bajar al primer hijo, (2) subir e ir al siguiente, (3) Crear un nodo hijo lambda e ir al siguiente nodo.
- Ejemplo, para el análisis de *id + id:*

Pila	Entrada	Operación	Adicionar
$ E	*id + id $*	*1*	*T Ep*
$ Ep T	*id + id $*	*1*	*F Tp*
$ Ep Tp F	*id + id $*	*1*	*id*
$ Ep Tp id	*id + id $*	*2*	

$ E' T'	*+ id $*	*3*	*‘’*
$ E’	*+ id $*	*1*	*+ T Ep*
$ Ep T +	*+ id $*	*2*	
$ Ep T	*id $*	*1*	*F Tp*
$ Ep Tp F	*id $*	*1*	*id*
$ Ep Tp id	*id $*	*2*	
$ Ep Tp	*$*	*3*	*‘’*
$ Ep	*$*	*3*	*‘’*
$	*$*		

3. **Crear las operaciones 1, 2 y 3:**

```
def opera1(pivote, literales):
  #Para cada literal, adicionarlo al nodo pivote.
  #retornar referencia al primer hijo (por la izquierda)
```

```
def opera2(pivote):
   # retornar referencia al siguiente hermano
   # si no hubiera, retornar el siguiente hermano del padre
   # si no hubiera continuar sucesivamente hasta encontrar
   # un hermano o el nodo raiz (en ese caso retornar vacio)
```

```
def opera3(pivote):
   # adicionar el operador lambda en el nodo en donde se encuentre
   # retornar opera2(pivote)
```

4. **Colocar las llamadas a las operaciones en el algoritmo presentado en la parte 3 de la práctica 5.**

Universidad Católica
San Pablo

```
def analizar(gramatica, sentencia)

    raiz = Nodo(gramatica->estadoInicial)
    pivote = raiz

    #… //cambiar
    # pivote = opera1 u opera2 u opera3 según corresponda
```