

PROCESADORES DEL LENGUAJE

ANALIZADOR SINTÁCTICO

IMPLEMENTACIÓN

Ángel Luis Ortiz Folgado 50774922-F

Oscar Eduardo Pérez la Madrid 54268894-A

Esteban Vargas Rastrollo 09207146-Q

Gramática inicial que formaliza la sintaxis del programa:

Programa \rightarrow Declaraciones && Instrucciones
Declaraciones \rightarrow Declaraciones; Declaración | Declaración
Declaración \rightarrow Tipo id
Tipo \rightarrow Int | Bool
Instrucciones \rightarrow Instrucciones; Instrucción | Instrucción
Instrucción \rightarrow id = Expresión
Expresión \rightarrow E1 op0 E1 | E1
E1 \rightarrow E1 op1 E2 | E2
E2 \rightarrow E2 op2 E3 | E3
E3 \rightarrow op3 E3 | E4
E4 \rightarrow num | id | (Expresión) | true | false
op0 \rightarrow < | > | <= | >= | != | ==
op1 \rightarrow + | - | or
op2 \rightarrow * | / | and
op3 \rightarrow - | not

Esta gramática no es LL(1) para que lo sea hay que eliminar recursión a izquierdas y factorizar.

Factorizar:

Expresión \rightarrow E1 op0 E1 | E1 se transforma en: Expresión \rightarrow E1 RE0
RE0 \rightarrow op0 E1 | ϵ

Eliminar recursión a izquierdas:

Declaraciones \rightarrow Declaraciones; Declaración | Declaración se transforma en:
Declaraciones \rightarrow Declaración ReDeclaraciones
ReDeclaraciones \rightarrow ;Declaración ReDeclaraciones | ϵ

Instrucciones \rightarrow Instrucciones; Instrucción | Instrucción se transforma en:
Instrucciones \rightarrow Instrucción ReInstrucciones
ReInstrucciones \rightarrow ; Instrucción ReInstrucciones | ϵ

E1 \rightarrow E1 op1 E2 | E2 se transforma en:
E1 \rightarrow E2 RE1
RE1 \rightarrow op1 E2 RE1 | ϵ

E2 \rightarrow E2 op2 E3 | E3 se transforma en:
E2 \rightarrow E3 RE2
RE2 \rightarrow op2 E3 RE2 | ϵ

Gramática LL (1)

Programa->Declaraciones && Instrucciones

Declaraciones-> Declaración ReDeclaraciones

ReDeclaraciones -> ;Declaración ReDeclaraciones | ϵ

Declaración->Tipo id

Tipo-> Int | Bool

Instrucciones -> Instrucción ReInstrucciones

ReInstrucciones -> ; Instrucción ReInstrucciones | ϵ

Instrucción->id = Expresión

Expresión -> E1 RE0

RE0 -> op0 E1 | ϵ

E1 -> E2 RE1

RE1->op1 E2 RE1 | ϵ

E2 -> E3 RE2

RE2 ->op2 E3 RE2 | ϵ

E3 -> op3 E3 | E4

E4 -> num | id | (Expresión) | true | false

op0 -> < | > | <= | >= | != | ==

op1 -> + | - | or

op2 -> * | / | and

op3 -> - | not