

# ANALIZADOR SINTÁCTICO TINY(0)

Grupo 29  
Adrian Estevez Gallego  
Alvaro Delgado Gutierrez

# 1. Especificación sintáctica de la gramática

Las palabras subrayadas se refieren a clases léxicas.

Prog -> SeccionDec && SeccionInst

SeccionDec -> Decs

Decs -> Dec

Decs -> Decs ; Dec

Dec -> NombreTipo Variable

NombreTipo -> int

NombreTipo -> bool

NombreTipo -> real

SeccionInst -> Insts

Insts -> Insts ; Inst

Insts -> Inst

Inst -> Variable = Expr0

Expr0 -> Expr1 + Expr0

Expr0 -> Expr1 - Expr1

Expr0 -> Expr1

Expr1 -> Expr1 Op1 Expr2

Expr1 -> Expr2

Expr2 -> Expr2 Op2 Expr3

Expr2 -> Expr3

Expr3 -> Expr4 Op3 Expr4

Expr3 -> Expr4

Expr4 -> not Expr4

Expr4 -> - Expr5

Expr4 -> Expr5

Expr5 -> (Expr0)

Expr5 -> Variable

Expr5 -> true

Expr5 -> false

Expr5 -> EntNum

Expr5 -> RealNum

Op1 -> and

Op1 -> or

Op2 -> <

Op2 -> >

Op2 -> <=

Op2 -> >=

Op2 -> ==

Op2 -> !=

Op3 -> \*

Op3 -> /

## 2. Transformaciones necesarias para obtener una gramática LL(1) equivalente

Las palabras subrayadas se refieren a clases léxicas.

```
Prog -> SeccionDec && SeccionInst
SeccionDec -> Decs
Decs -> Dec ReDecs
ReDecs -> ; Dec ReDecs
ReDecs -> ε
Dec -> NombreTipo Variable
NombreTipo -> int
NombreTipo -> bool
NombreTipo -> real
SeccionInst -> Insts
Insts -> Inst ReInsts
ReInsts -> ; Inst ReInsts
ReInsts -> ε
Inst -> Variable = Expr0
Expr0 -> Expr1 ReExpr0
ReExpr0 -> + Expr0
ReExpr0 -> - Expr1
ReExpr0 -> ε
Expr1 -> Expr2 ReExpr1
ReExpr1 -> Op1 Expr2 ReExpr1
ReExpr1 -> ε
Expr2 -> Expr3 ReExpr2
ReExpr2 -> Op2 Expr3 ReExpr2
ReExpr2 -> ε
Expr3 -> Expr4 ReExpr3
ReExpr3 -> Op3 Expr4
ReExpr3 -> ε
Expr4 -> not Expr4
Expr4 -> - Expr5
Expr4 -> Expr5
Expr5 -> (Expr0)
Expr5 -> Variable
Expr5 -> true
Expr5 -> false
Expr5 -> EntNum
Expr5 -> RealNum
Op1 -> and
Op1 -> or
Op2 -> <
Op2 -> >
Op2 -> <=
Op2 -> >=
Op2 -> ==
Op2 -> !=
Op3 -> *
Op3 -> /
```

### 3. Directores de la gramática

```
DIR(Prog -> SeccionDec && SeccionInst) -> int bool real
DIR(SeccionDec -> Decs) -> int bool real
DIR(Decs -> Dec ReDecs) -> int bool real
DIR(ReDecs -> ; Dec ReDecs) -> ;
DIR(ReDecs -> ε) -> &&
DIR(Dec -> NombreTipo Variable) -> int bool real
DIR(NombreTipo -> int) -> int
DIR(NombreTipo -> bool) -> bool
DIR(NombreTipo -> real) -> real
DIR(SeccionInst -> Insts) -> Variable
DIR(Insts -> Inst ReInsts) -> Variable
DIR(ReInsts -> ; Inst ReInsts) -> ;
DIR(ReInsts -> ε) -> EOF
DIR(Inst -> Variable = Expr0) -> Variable
DIR(Expr0 -> Expr1 ReExpr0) -> not - ( Variable true false EntNum RealNum
DIR(ReExpr0 -> + Expr0) -> +
DIR(ReExpr0 -> - Expr1) -> -
DIR(ReExpr0 -> ε) -> ) ; EOF
DIR(Expr1 -> Expr2 ReExpr1) -> not - ( Variable true false EntNum RealNum
DIR(ReExpr1 -> Op1 Expr2 ReExpr1) -> and or
DIR(ReExpr1 -> ε) -> ) + - ; EOF
DIR(Expr2 -> Expr3 ReExpr2) -> not - ( Variable true false EntNum RealNum
DIR(ReExpr2 -> Op2 Expr3 ReExpr2) -> < > <= >= == !=
DIR(ReExpr2 -> ε) -> ) and or + - ; EOF
DIR(Expr3 -> Expr4 ReExpr3) -> not - ( Variable true false EntNum RealNum
DIR(ReExpr3 -> Op3 Expr4) -> * /
DIR(ReExpr3 -> ε) -> ) < > <= >= == != and or + - ; EOF
DIR(Expr4 -> not Expr4) -> not
DIR(Expr4 -> - Expr5) -> -
DIR(Expr4 -> Expr5) -> ( Variable true false EntNum RealNum
DIR(Expr5 -> ( Expr0 )) -> (
DIR(Expr5 -> Variable) -> Variable
DIR(Expr5 -> true) -> true
DIR(Expr5 -> false) -> false
DIR(Expr5 -> EntNum) -> EntNum
DIR(Expr5 -> RealNum) -> RealNum
DIR(Op1 -> and) -> and
DIR(Op1 -> or) -> or
DIR(Op2 -> <) -> <
DIR(Op2 -> >) -> >
DIR(Op2 -> <=) -> <=
DIR(Op2 -> >=) -> >=
DIR(Op2 -> ==) -> ==
DIR(Op2 -> !=) -> !=
DIR(Op3 -> *) -> *
DIR(Op3 -> /) -> /
```

## 4. Símbolos de diagnóstico de la gramática

```
DIAG(Prog) -> int bool real
DIAG(SeccionDec) -> int bool real
DIAG(Decs) -> int bool real
DIAG(ReDecs)-> ; &&
DIAG(Dec) -> int bool real
DIAG(NombreTipo) -> int bool real
DIAG(SeccionInst) -> Variable
DIAG(Insts) -> Variable
DIAG(ReInsts) -> ;
DIAG(Inst) -> Variable
DIAG(Expr0) -> not - ( Variable true false EntNum RealNum
DIAG(ReExpr0) -> + -
DIAG(Expr1) -> not - ( Variable true false EntNum RealNum
DIAG(ReExpr1) -> and or + -
DIAG(Expr2) -> not - ( Variable true false EntNum RealNum
DIAG(ReExpr2) -> < > <= >= == != and or + -
DIAG(Expr3) -> not - ( Variable true false EntNum RealNum
DIAG(ReExpr3) -> * / < > <= >= == != and or + -
DIAG(Expr4) -> not - ( Variable true false EntNum RealNum
DIAG(Expr5) -> ( Variable true false EntNum RealNum
DIAG(Op1) -> and or
DIAG(Op2) -> < > <= >= == !=
DIAG(Op3 -> *) -> * /
```