**Analizador Sintáctico Descendiente**

**Jaime Sáez de Buruaga Brouns**

**Julia Miguélez Fernández-Villacañas**

1. **Especificación mediante gramática incontextual**

**NOTA:** ¿ID, REAL, ENT quedan dentro o fuera de la definición de la gramática?

ID = LET (LET | DIG | \_)\*

LET = a | A | b | B | … | z | Z

DIG = 0 | 1 | … | 9

REAL = [+ | - | ε] ( ENTDEC | ENTEXP | ENTDECEXP)

ENT = [+ | - | ε] POS (DIG)\*

POS = 1 | 2 | … | 9

DEC = .ENT

EXP = [e | E] ENT

G = (Vt, Vn, P, S)

Vt = [a..z, 0..9, \_, int, real, bool, true, false, E, +, -, ., and, or, >, <, >=, <=, ==, !=, \*, /, not, ]

Vn = [S, VAR, INSTR, DEC, T, ID, L, D, I, E0, E1, E2, E3, E4, E5, OP, ENT, R, DEC, EXP, P]

S = S

P = { S → VAR && INSTR,

VAR → DEC (; DEC)\*,

DEC → T ID,

T → int | real | bool,

*/\* ¿ESTAS SE OMITEN? \*/*

ID → L(L | D | \_)\*,

L → a | b | c | … | d,

D → 0 | 1 | 2 | … | 9,

INSTR → I (; I)\*,

I → ID = E0,

E0 → E0 + E1 | E0 – E1 | E1

E1 → E2 and E1 | E2 or E2 | E2

E2 → E3 OP E3 | E3

E3 → E3 \* E4 | E3 / E4 | E4

E4 → -E4 | not E5 | E5

E5 → (E0) | E0 | ID | R | ENT | true | false

OP → < | > | <= | >= | == | !=

ENT → (+ | - | ε)P(D\*)

D -> 0 | 1 | 2 | … | 9, | - | ε) P(D)\*,

R → ENT (DEC | EXP | DECEXP | ε)

DEC → .ENT

EXP → (e | E)ENT

P -> 1 | 2 | 3 | … | 9

}

**Nota**: como no se sabe poner algo provisional (puede haber + pero puede no haberlo, debería ser algo como (+ | ε), se ha puesto dos veces.

1. **Transformaciones necesarias para LL(1) equivalente.**
   1. **Eliminación de recursión por la izquierda**

G = (Vt, Vn, P, S)

Vt = [a..z, 0..9, \_, int, real, bool, true, false, E, +, -, ., and, or, >, <, >=, <=, ==, !=, \*, /, not]

Vn = [S, Sd, D, T, ID, L, DIG, Si, I, E0, E0’, E1, E2, E3, E3’, E4, E5, OP, ENT, REAL, DEC, EXP, POS]

S = S

P = { S → Sd && Si,

Sd → D | D; Sd,

D → T ID,

T → int | real | bool,

*/\* ¿ESTAS SE OMITEN? \*/*

ID → L(L | D | \_)\*,

L → a | b | c | … | d,

DIG → 0 | 1 | 2 | … | 9,

Si → I | I ; Si

I → ID = E0,

E0 → E1 E0’

E0’ → + E1 E0’ | - E1 E0’ | ε

E1 → E2 and E1 | E2 or E2 | E2

E2 → E3 OP E3 | E3

E3 → E4 E3’

E3’ → \* E4 E3 | / E4 E3 | ε

E4 → -E4 | not E5 | E5

E5 → (E0) | E0 | ID | R | ENT | true | false

OP → < | > | <= | >= | == | !=

ENT → (+ | - | ε)POS(DIG\*)

REAL → ENT (DEC | EXP | DECEXP | ε)

DEC → .ENT

EXP → (e | E)ENT

POS -> 1 | 2 | 3 | … | 9

}

* 1. **Eliminación de factores a la izquierda**

G = (Vt, Vn, P, S)

Vt = [a..z, 0..9, \_, int, real, bool, true, false, E, +, -, ., and, or, >, <, >=, <=, ==, !=, \*, /, not]

Vn = [S, Sd, D, DEC, T, ID, L, DIG, Si, I, INS, E0, E0’, E1, EE1, E2, EE2, E3, E3’, E4, E5, OP, ENT, REAL, DEC, EXP, POS]

S = S

P = { S → Sd && Si,

Sd → D | DEC

D → T ID,

DEC → ε | ; D DEC

T → int | real | bool,

*/\* ¿ESTAS SE OMITEN? \*/*

ID → L(L | D | \_)\*,

L → a | b | c | … | d,

DIG → 0 | 1 | 2 | … | 9,

Si → I | I INS

I → ID = E0,

INS → ε | ; I SINS

E0 → E1 E0’

E0’ → + E1 E0’ | - E1 E0’ | ε

E1 → E2 EE1

EE1 → and E1 | or E2 | ε

E2 → E3 EE2

EE2 → OP E3 | E3

E3 → E4 E3’

E3’ → \* E4 E3 | / E4 E3 | ε

E4 → -E4 | not E5 | E5

E5 → (E0) | E0 | ID | R | ENT | true | false

OP → < | > | <= | >= | == | !=

ENT → (+ | - | ε)POS(DIG\*)

REAL → ENT (DEC | EXP | DECEXP | ε)

DEC → .ENT

EXP → (e | E)ENT

POS -> 1 | 2 | 3 | … | 9

}

1. **No terminales: *primeros* y *salientes***

|  |  |  |
| --- | --- | --- |
| Productor | Primeros | Siguientes |
| S | int, real, bool | Ø |
| Sd | int, real, bool | && |
| D | int, real, bool | ;, && |
| DEC | ; | && |
| T | int, real, bool | ID |
| Si | ID | Ø |
| I | ID | ; |
| INS | ; | Ø |
| E0 | -, not, (, ID, REAL, ENT, true, false | ), ; |
| E0’ | +, - | ), ; |
| E1 | -, not, (, ID, REAL, ENT, true, false | ), +, -, ; |
| EE1 | and, or | ), +, -, ; |
| E2 | -, not, (, ID, REAL, ENT, true, false | and, or, ), +, -, ; |
| EE2 | <, >, <=, >=, !=, == | and, or, ), +, -, ; |
| E3 | -, not, (, ID, REAL, ENT, true, false | ), >, <, >=, <=, !=, ==, and, or, +, -, ; |
| E3’ | \*, / | ), >, <, >=, <=, !=, ==, and, or, +, -, ; |
| E4 | -, not, (, ID, REAL, ENT, true, false | ), \*, /, <, >, <=, >=, !=, ==, and, or, +, -, ; |
| E5 | (, ID, REAL, ENT, true, false | ), \*, /, <, >, <=, >=, !=, ==, and, or, +, -, ; |
| OP | <, >, <=, >=, ==, != | -, not, (, ID, REAL, ENT, true, false |

1. **Reglas: *directores***

|  |  |
| --- | --- |
| Productor | Directores |
| S → Sd && Si | int, real, bool |
| Sd → D DEC | int, real, bool |
| D → T ID | int, real, bool |
| DEC → ε | && |
| DEC → ; D DEC | int, real, bool |
| T → int | ID |
| T → real | ID |
| T → bool | ID |
| I → ID = E0 | ; |
| INS → ε | - |
| INS → ; I INS | ; |
| E0 → E1 E0’ | -, not, (, ID, INT, REAL, true, false |
| E0’ → + E1 E0’ | + |
| E0’ → - E1 E0’ | - |
| E0’ → ε | ), -, ; |
| E1 → E2 EE1 | -, not, (, ID, INT, REAL, true, false |
| EE1 → and E1 | and |
| EE1 → or E2 | or |
| EE1 → ε | +, -, ), -, ; |
| E2 → E3 EE2 | -, not, (, ID, INT, REAL, true, false |
| EE2 → OP E3 | <, >, <=, >=, ==, != |
| EE2 → ε | and, or, +, -, ), -, ; |
| E3 → E4 E3’ | -, not, (, INT, REAL, true, false |
| E3’ → \* E4 E3’ | \* |
| E3’ → / E4 E3’ | / |
| E3’ → ε | <, >, <=, >=, ==, !=, and, or, +, -, ), -, ;, not, (, ID, INT, REAL, true, false |
| E4 → - E4 | - |
| E4 → not E5 | not |
| E4 → E5 | (, ID, INT, REAL, true, false |
| E5 → (E0) | ( |
| E5 → ID | ID |
| E5 → REAL | REAL |
| E5 → ENT | ENT |
| E5 → true | True |
| E5 → false | False |
| OP → < | < |
| OP → > | > |
| OP → <= | <= |
| OP → >= | >= |
| OP → == | == |
| OP → != | != |