**Analizador Sintáctico Descendiente**

**Jaime Sáez de Buruaga Brouns**

**Julia Miguélez Fernández-Villacañas**

1. **Especificación mediante gramática incontextual**

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,

ID → L(L | D | \_)\*, */\* ¿ESTAS SE OMITEN? \*/*

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 → (+

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

R → ENT (DEC | EXP | DECEXP | \epsilon)

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 (+ | \epsilon), se ha puesto dos veces.

1. **Transformaciones necesarias para LL(1) equivalente.**

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]

1. **No terminales: *primeros* y *salientes***
2. **Reglas: *directores***