

PROCESADORES DE LENGUAJES

FASE 2.2:

DESARROLLO DE ANALIZADOR SINTÁCTICO PARA TINY(1).



Grupo 07:

HongXiang Chen y
Andrés Teruel Fernández

1. Especificación sintáctica para Tiny(1).

G = (Vn , Vt, P, S)

VN = {S, SecDec, Declaraciones, Declaracion, DeclaracionVar, DeclaracionTipo, DeclaracionProc, ParametrosFormales, ParametrosFormalesAux, Parametro, Bloque, Tipo, Array, Registro, Campos, Campo, Puntero, SecInstr, Instruccion, InstruccionCall, InstruccionAsignacion, InstruccionIf, SecInstr2, InstruccionIfElse, InstruccionWhile, InstruccionRead, InstruccionWrite, InstruccionNew, InstruccionDelete, ParametrosReales, ParametrosRealesAux, ExpresionBasica, E0, E1, E2, E3, E4, E5, E6, E7, Op1, Op2, Op3, Op4, Indexacion, Acceso_registro}

VT = {NL, OF, TYPE, PROC, SEPARADOR, PUNTOCOMA, VAR, IDENTIFICADOR, PAR_ABIERTO, PAR_CERRADO, COMA, AMPERSAND, LLAVE_ABIERTA, LLAVE_CERRADA, INT, REAL, BOOL, STRING, ARRAY, RECORD, POINTER, CORCHETE_ABIERTO, CORCHETE_CERRADO, ASIGNACION, MAS, MENOS, NOT, NUM_ENT, NUM_REAL, BOOLEAN, AND, OR, MENOR, MAYOR, MENOR_IGUAL, MAYOR_IGUAL, DIFERENTE, IGUAL, MULT, DIV, NULL, IF, ELSE, WHILE, ENDIF, ENDWHILE, DO, THEN, READ, WRITE, NEW, DELETE, CALL, PORCENTAJE, PUNTO, FLECHA , CADENA}

P = {

S → SecDec SecInstr;

SecDec → Declaraciones SEPARADOR | ε;

Declaraciones → Declaraciones PUNTOCOMA Declaracion | Declaracion;

Declaracion → DeclaracionVar | DeclaracionTipo | DeclaracionProc;

DeclaracionVar → VAR Tipo IDENTIFICADOR;

DeclaracionTipo → TYPE Tipo IDENTIFICADOR;

DeclaracionProc → PROC IDENTIFICADOR PAR_ABIERTO ParametrosFormales
PAR_CERRADO Bloque;

ParametrosFormales → ParametrosFormalesAux | ε;

ParametrosFormalesAux → ParametrosFormalesAux COMA Parametro | Parametro;

Parametro → Tipo IDENTIFICADOR | Tipo AMPERSAND IDENTIFICADOR;

Bloque → LLAVE_ABIERTA SecDec SecInstr LLAVE_CERRADA;

Tipo → INT | REAL | BOOL | STRING | IDENTIFICADOR | Array | Registro | Puntero;

Array → ARRAY CORCHETE_ABIERTO NUM_ENT CORCHETE_CERRADO OF Tipo;

Registro → RECORD LLAVE_ABIERTA Campos LLAVE_CERRADA;

Campos → Campos PUNTOCOMA Campo | Campo;

Campo → Tipo IDENTIFICADOR;

Puntero → POINTER Tipo;

SecInstr → SecInstr PUNTOCOMA Instruccion | Instruccion;

Instruccion → InstruccionAsignacion | InstruccionIf | InstruccionIfElse | InstruccionWhile |

InstruccionRead | InstruccionWrite | NL | InstruccionNew | InstruccionDelete |

InstruccionCall | Bloque;

InstruccionAsignacion → E0 ASIGNACION E0;

InstruccionIf → IF E0 THEN SecInstr2 ENDIF;

```

SecInstr2 → SecInstr | ε;
InstruccionIfElse → IF E0 THEN SecInstr2 ELSE SecInstr2 ENDIF;
InstruccionWhile → WHILE E0 DO SecInstr2 ENDWHILE;
InstruccionRead → READ E0;
InstruccionWrite → WRITE E0;
InstruccionNew → NEW E0;
InstruccionDelete → DELETE E0;
InstruccionCall → CALL IDENTIFICADOR PAR_ABIERTO ParametrosReales
PAR_CERRADO;
ParametrosReales → ParametrosRealesAux | ε;
ParametrosRealesAux → ParametrosRealesAux COMA E0 | E0;
ExpresionBasica → NUM_ENT | NUM_REAL | IDENTIFICADOR | BOOLEAN | CADENA
| NULL;
E0 → E1 MAS E0 | E1 MENOS E1 | E1;
E1 → E1 Op1 E2 | E2;
E2 → E2 Op2 E3 | E3;
E3 → E4 Op3 E4 | E4;
E4 → NOT E4 | MENOS E5 | E5;
E5 → E5 Op4 | E6;
E6 → MULT E6 | E7;
E7 → ExpresionBasica | PAR_ABIERTO E0 PAR_CERRADO;
Op1 → AND | OR;
Op2 → MENOR | MAYOR | MENOR_IGUAL | MAYOR_IGUAL | DIFERENTE | IGUAL;
Op3 → MULT | DIV | PORCENTAJE;
Op4 → Indexacion | Acceso_registro;
Indexacion → CORCHETE_ABIERTO E0 CORCHETE_CERRADO;
Acceso_registro → PUNTO IDENTIFICADOR | FLECHA IDENTIFICADOR;
}

```

2. Acondicionamiento de la gramática.

G = (Vn , Vt, P, S)

VN = {S, SecDec, Else, Declaraciones, DeclaracionesAux, Declaracion, DeclaracionVar, DeclaracionTipo, DeclaracionProc, ParametrosFormales, ParametrosFormalesAux, Parametro, ParametroAux, Bloque, Tipo, Array, Registro, Campos, CamposAux, Campo, Puntero, SecInstr, SecInstrAux, Instruccion, InstruccionAsignacion, InstruccionIf, SecInstr2, InstruccionIfElse, InstruccionWhile, InstruccionRead, InstruccionWrite, InstruccionNew, InstruccionDelete, InstruccionCall, ParametrosReales, ParametrosRealesAux, ExpresionBasica, E0, E1, E2, E3, E4, E5, E6, E7, Op1, Op2, Op3, Op4, Indexacion, Acceso_registro, E0_AUX, E1_AUX, E2_AUX, E3_AUX, E5_AUX}

$VT = \{NL, OF, TYPE, PROC, SEPARADOR, PUNTOCOMA, VAR, IDENTIFICADOR, PAR_ABIERTO, PAR_CERRADO, COMA, AMPERSAND, LLAVE_ABIERTA, LLAVE_CERRADA, INT, REAL, BOOL, STRING, ARRAY, RECORD, POINTER, CORCHETE_ABIERTO, CORCHETE_CERRADO, ASIGNACION, MAS, MENOS, NOT, NUM_ENT, NUM_REAL, BOOLEAN, AND, OR, MENOR, MAYOR, MENOR_IGUAL, MAYOR_IGUAL, DIFERENTE, IGUAL, MULT, DIV, NULL, IF, ELSE, WHILE, ENDIF, ENDWHILE, DO, THEN, READ, WRITE, NEW, DELETE, CALL, PORCENTAJE, PUNTO, FLECHA, CADENA\}$

P = {

$SAux \rightarrow S \mid -;$
 $S \rightarrow SecDec\ SecInstr;$
 $SecDec \rightarrow Declaraciones\ SEPARADOR \mid \epsilon;$
 $Declaraciones \rightarrow Declaracion\ DeclaracionesAux;$
 $DeclaracionesAux \rightarrow PUNTOCOMA\ Declaracion\ DeclaracionesAux \mid \epsilon;$
 $Declaracion \rightarrow DeclaracionVar \mid DeclaracionTipo \mid DeclaracionProc;$
 $DeclaracionVar \rightarrow VAR\ Tipo\ IDENTIFICADOR;$
 $DeclaracionTipo \rightarrow TYPE\ Tipo\ IDENTIFICADOR;$
 $DeclaracionProc \rightarrow PROC\ IDENTIFICADOR\ PAR_ABIERTO\ ParametrosFormales\ PAR_CERRADO\ Bloque;$
 $ParametrosFormales \rightarrow Parametro\ ParametrosFormalesAux \mid \epsilon;$
 $ParametrosFormalesAux \rightarrow COMA\ Parametro\ ParametrosFormalesAux \mid \epsilon;$
 $Parametro \rightarrow Tipo\ ParametroAux;$
 $ParametroAux \rightarrow IDENTIFICADOR \mid AMPERSAND\ IDENTIFICADOR;$
 $Bloque \rightarrow LLAVE_ABIERTA\ SecDec\ SecInstr\ LLAVE_CERRADA;$
 $Tipo \rightarrow INT \mid REAL \mid BOOL \mid STRING \mid IDENTIFICADOR \mid Array \mid Registro \mid Puntero;$
 $Array \rightarrow ARRAY\ CORCHETE_ABIERTO\ NUM_ENT\ CORCHETE_CERRADO\ OF\ Tipo;$
 $Registro \rightarrow RECORD\ LLAVE_ABIERTA\ Campos\ LLAVE_CERRADA;$
 $Campos \rightarrow Campo\ CamposAux;$
 $CamposAux \rightarrow PUNTOCOMA\ Campo\ CamposAux \mid \epsilon;$
 $Campo \rightarrow Tipo\ IDENTIFICADOR;$
 $Puntero \rightarrow POINTER\ Tipo;$
 $SecInstr \rightarrow Instruccion\ SecInstrAux;$
 $SecInstrAux \rightarrow PUNTOCOMA\ Instruccion\ SecInstrAux \mid \epsilon;$
 $Instruccion \rightarrow InstruccionAsignacion \mid InstruccionIf \mid InstruccionWhile \mid InstruccionRead \mid InstruccionWrite \mid NL \mid InstruccionNew \mid InstruccionDelete \mid InstruccionCall \mid Bloque;$
 $InstruccionAsignacion \rightarrow E0\ ASIGNACION\ E0;$
 $InstruccionIf \rightarrow IF\ E0\ THEN\ SecInstr2\ Else\ ENDIF;$
 $SecInstr2 \rightarrow SecInstr \mid \epsilon;$
 $Else \rightarrow ELSE\ SecInstr2 \mid \epsilon;$
 $InstruccionWhile \rightarrow WHILE\ E0\ DO\ SecInstr2\ ENDWHILE;$
 $InstruccionRead \rightarrow READ\ E0;$
 $InstruccionWrite \rightarrow WRITE\ E0;$
 $InstruccionNew \rightarrow NEW\ E0;$
 $InstruccionDelete \rightarrow DELETE\ E0 ;$

InstruccionCall \rightarrow CALL IDENTIFICADOR PAR_ABIERTO ParametrosReales
PAR_CERRADO;
ParametrosReales \rightarrow E0 ParametrosRealesAux | ϵ ;
ParametrosRealesAux \rightarrow COMA E0 ParametrosRealesAux | ϵ ;
ExpresionBasica \rightarrow NUM_ENT | NUM_REAL | IDENTIFICADOR | BOOLEAN |
CADENA | NULL;
E0 \rightarrow E1 E0_AUX ;
E0_AUX \rightarrow MAS E0 | MENOS E1 | ϵ ;
E1 \rightarrow E2 E1_AUX;
E1_AUX \rightarrow Op1 E2 E1_AUX | ϵ ;
E2 \rightarrow E3 E2_AUX;
E2_AUX \rightarrow Op2 E3 E2_AUX | ϵ ;
E3 \rightarrow E4 E3_AUX;
E3_AUX \rightarrow Op3 E4 | ϵ ;
E4 \rightarrow NOT E4 | MENOS E5 | E5;
E5 \rightarrow E6 E5_AUX ;
E5_AUX \rightarrow Op4 E5_AUX | ϵ ;
E6 \rightarrow MULT E6 | E7;
E7 \rightarrow ExpresionBasica | PAR_ABIERTO E0 PAR_CERRADO
Op1 \rightarrow AND | OR
Op2 \rightarrow MENOR | MAYOR | MENOR_IGUAL | MAYOR_IGUAL | DIFERENTE | IGUAL;
Op3 \rightarrow MULT | DIV | PORCENTAJE;
Op4 \rightarrow Indexacion | Acceso_registro
Indexacion \rightarrow CORCHETE_ABIERTO E0 CORCHETE_CERRADO
Acceso_registro \rightarrow PUNTO IDENTIFICADOR | FLECHA IDENTIFICADOR;

}