

# UNIVERSIDAD COMPLUTENSE DE MADRID

FACULTAD DE INFORMÁTICA

GRUPO 19



Procesadores de Lenguajes

## **Analizador Léxico**

Realizado por:

**Kevin Oscar Arce Vera, Daixiang Chen, Luna Santos Mirete, Juan  
Diego Martínez Banda**

Profesor:

**Jose Luis Sierra Rodriguez**

Curso: 4º

**2023/2024**

# Tiny 0

## Sin acondicionar la gramática para la implementación

programa\_tiny -> bloque  
bloque -> { lista\_opt\_declaraciones lista\_opt\_instrucciones }

### ## INI Sección declaraciones

lista\_opt\_declaraciones -> lista\_declaraciones &&  
lista\_opt\_declaraciones ->  $\epsilon$

lista\_declaraciones -> lista\_declaraciones ; declaracion  
lista\_declaraciones -> declaracion

declaracion -> tipo **identificador**  
tipo -> **INT**  
tipo -> **REAL**  
tipo -> **BOOL**

### ## INI Sección instrucciones

lista\_opt\_instrucciones -> lista\_instrucciones  
lista\_opt\_instrucciones ->  $\epsilon$

lista\_instrucciones -> lista\_instrucciones ; instruccion  
lista\_instrucciones -> instruccion

instruccion -> @ expresion

### ## INI Sección expresiones

expresion -> E0  
E0 -> E1 = E0  
E0 -> E1  
E1 -> E1 OP1 E2  
E1 -> E2  
E2 -> E2 + E3  
E2 -> E3 - E3  
E2 -> E3

E3 -> E4 **and** E3  
E3 -> E4 **or** E4  
E3 -> E4  
E4 -> E4 OP4 E5  
E4 -> E5  
E5 -> OP5 E5

E5 -> E6  
E6 -> **Entero**  
E6 -> **Real**  
E6 -> **TRUE**  
E6 -> **FALSE**  
E6 -> **Identificador**  
E6 -> ( E0 )

OP1 -> >  
OP1 -> >=  
OP1 -> <  
OP1 -> <=  
OP1 -> ==  
OP1 -> !=

OP4 -> \*  
OP4 -> /

OP5 -> **not**  
OP5 -> -

# Acondicionado para la gramática

programa\_tiny  $\rightarrow$  bloque

bloque  $\rightarrow$  { lista\_opt\_declaraciones lista\_opt\_instrucciones }

## ## INI Sección declaraciones

lista\_opt\_declaraciones  $\rightarrow$  lista\_declaraciones &&

lista\_opt\_declaraciones  $\rightarrow$   $\epsilon$

lista\_declaraciones  $\rightarrow$  declaracion lista\_decs

lista\_decs  $\rightarrow$  ; declaracion lista\_decs

lista\_decs  $\rightarrow$   $\epsilon$

declaracion  $\rightarrow$  tipo **identificador**

tipo  $\rightarrow$  **INT**

tipo  $\rightarrow$  **REAL**

tipo  $\rightarrow$  **BOOL**

## ## INI Sección instrucciones

lista\_opt\_instrucciones  $\rightarrow$  lista\_instrucciones

lista\_opt\_instrucciones  $\rightarrow$   $\epsilon$

lista\_instrucciones  $\rightarrow$  instruccion lista\_ins

lista\_ins  $\rightarrow$  ; instruccion lista\_ins

lista\_ins  $\rightarrow$   $\epsilon$

instruccion  $\rightarrow$  @ expresion

## ## INI Sección expresiones

expresion  $\rightarrow$  E0

E0  $\rightarrow$  E1 FE0

FE0  $\rightarrow$  = E0

FE0  $\rightarrow$   $\epsilon$

E1  $\rightarrow$  E2 RE1

RE1  $\rightarrow$  OP1 E2 RE1

RE1  $\rightarrow$   $\epsilon$

E2  $\rightarrow$  E3 FE2 RE2

RE2  $\rightarrow$  + E3 RE2

RE2  $\rightarrow$   $\epsilon$

FE2  $\rightarrow$  - E3

FE2  $\rightarrow$   $\epsilon$

E3 -> E4 FE3  
FE3 -> **and** E3  
FE3 -> **or** E4  
FE3 ->  $\epsilon$

E4 -> E5 RE4  
RE4 -> OP4 E5 RE4  
RE4 ->  $\epsilon$

E5 -> OP5 E5  
E5 -> E6

E6 -> **Entero**  
E6 -> **Real**  
E6 -> **TRUE**  
E6 -> **FALSE**  
E6 -> **Identificador**  
E6 -> ( E0 )

OP1 -> >  
OP1 -> >=  
OP1 -> <  
OP1 -> <=  
OP1 -> ==  
OP1 -> !=

OP4 -> \*  
OP4 -> /

OP5 -> **not**  
OP5 -> -

<b>REGLA</b>	<b>DIRECTORES</b>	<b>ANULABLE</b>
programa_tiny → bloque	{	NO
bloque → { lista_opt_declaraciones lista_opt_instrucciones }	{	NO
lista_opt_declaraciones → lista_declaraciones &&	INT, REAL, BOOL	NO
lista_opt_declaraciones → ε		SI
lista_declaraciones → declaracion lista_decs	INT, REAL, BOOL	NO
lista_decs → ; declaracion lista_decs	;	NO
lista_decs → ε		SI
declaracion → tipo <b>identificador</b>	INT, REAL, BOOL	NO
tipo → INT	INT	NO
tipo → REAL	REAL	NO
tipo → BOOL	BOOL	NO
lista_opt_instrucciones → lista_instrucciones	@	NO
lista_opt_instrucciones → ε		SI
lista_instrucciones → instruccion lista_ins	@	NO
lista_ins → ; instruccion lista_ins	;	NO
lista_ins → ε		SI
Instruccion → @ expresion	@	NO
expresion → E0	Entero, Real, TRUE, FALSE, Identificador, ( not, -	NO

$E0 \rightarrow E1 FE0$	<b>Entero, Real, TRUE, FALSE, Identificador, (, not, -</b>	NO
$FE0 \rightarrow = E0$	<b>=</b>	NO
$FE0 \rightarrow \epsilon$		SI
$E1 \rightarrow E2 RE1$	<b>Entero, Real, TRUE, FALSE, Identificador, (, not, -</b>	NO
$RE1 \rightarrow OP1 E2 RE1$	<b>&lt;, &lt;=, &gt;, &gt;=, ==, !=</b>	NO
$RE1 \rightarrow \epsilon$		SI
$E2 \rightarrow E3 FE2 RE2$	<b>Entero, Real, TRUE, FALSE, Identificador, (, not, -</b>	NO
$RE2 \rightarrow + E3 RE2$	<b>+</b>	NO
$RE2 \rightarrow \epsilon$		SI
$FE2 \rightarrow - E3$	<b>-</b>	NO
$FE2 \rightarrow \epsilon$		SI
$E3 \rightarrow E4 FE3$	<b>Entero, Real, TRUE, FALSE, Identificador, (, not, -</b>	NO
$FE3 \rightarrow \text{and } E3$	<b>and</b>	NO
$FE3 \rightarrow \text{or } E4$	<b>or</b>	NO
$FE3 \rightarrow \epsilon$		SI
$E4 \rightarrow E5 RE4$	<b>Entero, Real, TRUE, FALSE, Identificador, (, not, -</b>	NO
$RE4 \rightarrow OP4 E5 RE4$	<b>*, /</b>	NO
$RE4 \rightarrow \epsilon$		SI
$E5 \rightarrow OP5 E5$	<b>not, -</b>	NO
$E5 \rightarrow E6$	<b>Entero, Real, TRUE, FALSE, Identificador, (</b>	NO
$E6 \rightarrow \text{Entero}$	<b>Entero</b>	NO
$E6 \rightarrow \text{Real}$	<b>Real</b>	NO

E6 → <b>TRUE</b>	<b>TRUE</b>	NO
E6 → <b>FALSE</b>	<b>FALSE</b>	NO
E6 → <b>Identificador</b>	<b>Identificador</b>	NO
E6 → ( E0 )	(	NO
OP1 → >	>	NO
OP1 → >=	>=	NO
OP1 → <	<	NO
OP1 → <=	<=	NO
OP1 → ==	==	NO
OP1 → !=	!=	NO
OP4 → *	*	NO
OP4 → /	/	NO
OP5 → <b>not</b>	<b>not</b>	NO
OP5 → -	-	NO



# TINY

## Sin acondicionar la gramática para la implementación

programa\_tiny -> bloque  
bloque -> { lista\_opt\_declaraciones lista\_opt\_instrucciones }

### ## INI Sección declaraciones

lista\_opt\_declaraciones -> lista\_declaraciones &&  
lista\_opt\_declaraciones -> ε

lista\_declaraciones -> lista\_declaraciones ; declaracion  
lista\_declaraciones -> declaracion

declaracion -> declaracion\_variable  
declaracion -> declaracion\_tipo  
declaracion -> declaracion\_proc

declaracion\_variable -> tipo **Identificador**  
declaracion\_tipo -> **TYPE** tipo **Identificador**  
declaracion\_proc -> cabecera\_proc bloque  
cabecera\_proc -> **PROC Identificador** ( lista\_opt\_parametros\_formales )

lista\_opt\_parametros\_formales -> lista\_parametros\_formales  
lista\_opt\_parametros\_formales -> ε

lista\_parametros\_formales -> lista\_parametros\_formales , parametro\_formal  
lista\_parametros\_formales -> parametro\_formal

parametro\_formal -> tipo & **Identificador**  
parametro\_formal -> tipo **Identificador**

## Aplicamos el patrón expresiones para los tipos

## Puede haber array de cualquier cosa

## No puede haber punteros a array

tipo -> tipo\_array

tipo\_array -> tipo\_array [ **Entero** ]    **##** Prioridad 0

tipo\_array -> tipo\_puntero

tipo\_puntero -> ^ tipo\_puntero        **##** Prioridad 1

tipo\_puntero -> T2

**##** Prioridad 2

T2 -> tipo\_basico

T2 -> tipo\_registro

T2 -> tipo\_definido

tipo\_basico -> **INT**

tipo\_basico -> **REAL**

tipo\_basico -> **BOOL**

tipo\_basico -> **STRING**

tipo\_registro -> **STRUCT** { lista\_opt\_parametros\_registro }

tipo\_definido -> **Identificador**

lista\_opt\_parametros\_registro -> lista\_parametros\_registro

lista\_opt\_parametros\_registro -> ε

lista\_parametros\_registro -> lista\_parametros\_registro , parametro\_registro

lista\_parametros\_registro -> parametro\_registro

parametro\_registro -> tipo **Identificador**

### **## INI Sección instrucciones**

lista\_opt\_instrucciones -> lista\_instrucciones

lista\_opt\_instrucciones -> ε

lista\_instrucciones -> lista\_instrucciones ; instruccion

lista\_instrucciones -> instruccion

instruccion -> instruccion-eval

instruccion -> instruccion-if

instruccion -> instruccion-if-else

instruccion -> instruccion-while

instruccion -> instruccion-read

instruccion -> instruccion-write

instruccion -> instruccion-nl

instruccion -> instruccion-new

instruccion -> instruccion-delete

instruccion -> instruccion-call

instruccion -> instruccion-compuesta

### **## INI Sección expresiones**

instruccion-eval -> @ expresion

expresion -> E0

E0 -> E1 = E0

E0 -> E1

E1 -> E1 OP1 E2

E1 -> E2

E2 -> E2 + E3

E2 -> E3 - E3

E2 -> E3

E3 -> E4 **and** E3

E3 -> E4 **or** E4

E3 -> E4

E4 -> E4 OP4 E5

E4 -> E5

E5 -> OP5 E5

E5 -> E6

E6 -> E6 OP6

E6 -> E7

E7 -> **Entero**

E7 -> **Real**

E7 -> **TRUE**

E7 -> **FALSE**

E7 -> **NULL**

E7 -> **Cadena**

E7 -> **Identificador**

E7 -> ( E0 )

OP1 -> >

OP1 -> >=

OP1 -> <

OP1 -> <=

OP1 -> ==

OP1 -> !=

OP4 -> \*

OP4 -> /

OP4 -> %

OP5 -> **not**

OP5 -> -

OP6 -> [ E0 ]

OP6 -> . **Identificador**

OP6 -> ^

**## FIN Sección Expresiones**

instruccion-if -> **IF** expresion bloque  
instruccion-if-else -> instruccion-if **ELSE** bloque  
instruccion-while -> **WHILE** expresion bloque  
instrucción-read -> **READ** expresion  
instruccion-write -> **WRITE** expresion  
instruccion-nl -> **NL**  
instruccion-new -> **NEW** expresion  
instruccion-delete -> **DELETE** expresion  
instruccion-call -> **CALL** **Identificador** parametros

parametros -> ( lista\_opt\_parametros )

lista\_opt\_parametros -> lista\_parametros  
lista\_opt\_parametros ->  $\epsilon$

lista\_parametros -> lista\_parametros , expresion  
lista\_parametros -> expresion

instruccion-compuesta -> bloque

## Acondicionada la gramática

programa\_tiny -> bloque  
bloque -> { lista\_opt\_declaraciones lista\_opt\_instrucciones }

### **## INI Sección declaraciones**

lista\_opt\_declaraciones -> lista\_declaraciones &&  
lista\_opt\_declaraciones ->  $\epsilon$

lista\_declaraciones  $\rightarrow$  declaracion lista\_decs  
lista\_decs  $\rightarrow$  ; declaracion lista\_decs  
lista\_decs  $\rightarrow$   $\epsilon$

declaracion -> declaracion\_variable  
declaracion -> declaracion\_tipo  
declaracion -> declaracion\_proc

declaracion\_variable -> tipo **Identificador**  
declaracion\_tipo -> **TYPE** tipo **Identificador**  
declaracion\_proc -> **PROC** **Identificador** ( lista\_opt\_parametros\_formales ) bloque

lista\_opt\_parametros\_formales -> lista\_parametros\_formales  
lista\_opt\_parametros\_formales ->  $\epsilon$

lista\_parametros\_formales -> parametro\_formal RECP  
RECP -> , parametro\_formal RECP  
RECP ->  $\epsilon$

parametro\_formal -> tipo RECpf  
RECpf -> **& Identificador**  
RECpf -> **Identificador**

tipo -> tipo\_array

tipo\_array -> tipo\_puntero RETA      **## Prioridad 0**  
RETA -> [ **Entero** ] RETA  
RETA ->  $\epsilon$

tipo\_puntero -> ^ tipo\_puntero      **## Prioridad 1**  
tipo\_puntero -> T2

T2 -> tipo\_basico      **##Prioridad 2**  
T2 -> tipo\_registro  
T2 -> tipo\_definido

tipo\_basico -> **INT | REAL | BOOL | STRING**  
tipo\_registro -> **STRUCT** { lista\_opt\_parametros\_registro }  
tipo\_definido -> **Identificador**

lista\_opt\_parametros\_registro -> lista\_parametros\_registro  
lista\_opt\_parametros\_registro ->  $\epsilon$

lista\_parametros\_registro -> parametro\_registro REClpr  
REClpr -> , parametro\_registro REClpr  
REClpr ->  $\epsilon$

parametro\_registro -> tipo **Identificador**

### **## INI Sección instrucciones**

lista\_opt\_instrucciones -> lista\_instrucciones  
lista\_opt\_instrucciones ->  $\epsilon$

lista\_instrucciones -> instruccion RECI  
RECI -> ; instruccion RECI  
RECI ->  $\epsilon$

instruccion -> instruccion-eval  
instruccion -> instruccion-if  
instruccion -> instruccion-while

instruccion -> instruccion-read  
instruccion -> instruccion-write  
instruccion -> instruccion-nl  
instruccion -> instruccion-new  
instruccion -> instruccion-delete  
instruccion -> instruccion-call

## **## INI Sección expresiones**

instruccion-eval -> @ expresion

expresion -> E0

E0 -> E1 FE0

FE0 -> = E0

FE0 ->  $\epsilon$

E1 -> E2 RE1

RE1 -> OP1 E2 RE1

RE1 ->  $\epsilon$

E2 -> E3 FE2 RE2

RE2 -> + E3 RE2

RE2 ->  $\epsilon$

FE2 -> - E3

FE2 ->  $\epsilon$

E3 -> E4 FE3

FE3 -> **and** E3

FE3 -> **or** E4

FE3 ->  $\epsilon$

E4 -> E5 RE4

RE4 -> OP4 E5 RE4

RE4 ->  $\epsilon$

E5 -> OP5 E5

E5 -> E6

E6 -> E7 RE6

RE6 -> OP6 RE6

RE6 ->  $\epsilon$

E7 -> **Entero**

E7 -> **Real**

E7 -> **TRUE**

E7 -> **FALSE**

E7 -> **NULL**

E7 -> **Cadena**  
E7 -> **Identificador**  
E7 -> ( E0 )

OP1 -> >  
OP1 -> >=  
OP1 -> <  
OP1 -> <=  
OP1 -> ==  
OP1 -> !=

OP4 -> \*  
OP4 -> /  
OP4 -> %

OP5 -> **not**  
OP5 -> -

OP6 -> [ E0 ]  
OP6 -> . **Identificador**  
OP6 -> ^

## **## FIN Sección Expresiones**

instruccion-if -> **IF** expresion bloque instruccion-else  
instruccion-else -> **ELSE** bloque  
instruccion-else ->  $\epsilon$   
instruccion-while -> **WHILE** expresion bloque  
instrucción-read -> **READ** expresion  
instruccion-write -> **WRITE** expresion  
instruccion-nl -> **NL**  
instruccion-new -> **NEW** expresion  
instruccion-delete -> **DELETE** expresion  
instruccion-call -> **CALL Identificador** parametros

parametros -> ( lista\_opt\_parametros )

lista\_opt\_parametros -> lista\_parametros  
lista\_opt\_parametros ->  $\epsilon$

lista\_parametros -> expresion REClp  
REClp -> , expresion REClp  
REClp ->  $\epsilon$

instruccion -> instruccion-compuesta  
instruccion-compuesta -> bloque