
Procesadores de Lenguajes

Memoria de proyecto — Hito 2: Analizador Sintáctico

GRUPO 14

RODRIGO SOUTO SANTOS
LEONARDO PRADO DE SOUZA
JUAN ANDRÉS HIBJAN CARDONA
IZAN RODRIGO SANZ

*Grado en Ingeniería informática
Facultad de Informática
Universidad Complutense de Madrid*



Índice general

1. Tiny (0)	2
1.1. Especificación Sintáctica (Gramática)	2
1.1.1. Declaraciones	2
1.1.2. Tipos	2
1.1.3. Instrucciones	2
1.1.4. Expresiones	2
1.1.5. Operadores	3
1.2. Acondicionamiento	3
1.2.1. Declaraciones	3
1.2.2. Tipos	3
1.2.3. Instrucciones	3
1.2.4. Expresiones	4
1.2.5. Operadores	4
1.3. Directores	4
1.3.1. Tabla de Reglas	5
2. Tiny	7
2.1. Especificación Sintáctica (Gramática)	7
2.1.1. Declaraciones	7
2.1.2. Tipos	7
2.1.3. Instrucciones	7
2.1.4. Expresiones	8
2.1.5. Operadores	8
2.2. Acondicionamiento	9
2.2.1. Declaraciones	9
2.2.2. Tipos	9
2.2.3. Instrucciones	10
2.2.4. Expresiones	10
2.2.5. Operadores	11
Índice de cuadros	12

1 | Tiny (0)

1.1. Especificación Sintáctica (Gramática)

Implementamos la gramática que define la especificación sintáctica del lenguaje Tiny0 empleando los patrones explicados en clase (Diseño descendente, Reutilización, Nivel de Abstracción Equilibrado, Opcionalidad, Variantes, Listas y Expresiones).

Para ello definimos primero la estructura básica de todo programa:

```
programa  $\rightarrow$  bloque
bloque  $\rightarrow$  {seccion_declaraciones_opt seccion_intrucciones_opt}
```

1.1.1. Declaraciones

```
seccion_declaraciones_opt  $\rightarrow$  seccion_declaraciones &&
seccion_declaraciones_opt  $\rightarrow$   $\epsilon$ 
seccion_declaraciones  $\rightarrow$  seccion_declaraciones ; declaracion
seccion_declaraciones  $\rightarrow$  declaracion
declaracion  $\rightarrow$  tipo_nombre
```

1.1.2. Tipos

```
tipo_nombre  $\rightarrow$  tipo_base identificador
tipo_base  $\rightarrow$  int
tipo_base  $\rightarrow$  real
tipo_base  $\rightarrow$  bool
```

1.1.3. Instrucciones

```
seccion_intrucciones_opt  $\rightarrow$  seccion_intrucciones
seccion_intrucciones_opt  $\rightarrow$   $\epsilon$ 
seccion_intrucciones  $\rightarrow$  lista_intrucciones
lista_intrucciones  $\rightarrow$  lista_intrucciones ; instruccion
lista_intrucciones  $\rightarrow$  instruccion
instruccion  $\rightarrow$  @ expresion
```

1.1.4. Expresiones

```
expresion  $\rightarrow$  E0
E0  $\rightarrow$  E1 = E0
E0  $\rightarrow$  E1
E1  $\rightarrow$  E1 op_relacional E2
E1  $\rightarrow$  E2
E2  $\rightarrow$  E2 + E3
E2  $\rightarrow$  E3 - E3
E2  $\rightarrow$  E3
E3  $\rightarrow$  E4 and E3
E3  $\rightarrow$  E4 or E4
E3  $\rightarrow$  E4
E4  $\rightarrow$  E4 op_mult E5
E4  $\rightarrow$  E5
E5  $\rightarrow$  - E5
E5  $\rightarrow$  not E5
```

$E5 \rightarrow E6$
 $E6 \rightarrow \text{expresion_basica}$
 $E6 \rightarrow (E0)$
 $\text{expresion_basica} \rightarrow \text{literalEntero}$
 $\text{expresion_basica} \rightarrow \text{literalReal}$
 $\text{expresion_basica} \rightarrow \text{true}$
 $\text{expresion_basica} \rightarrow \text{false}$
 $\text{expresion_basica} \rightarrow \text{identificador}$

1.1.5. Operadores

$op_relacional \rightarrow <$
 $op_relacional \rightarrow <=$
 $op_relacional \rightarrow >$
 $op_relacional \rightarrow >=$
 $op_relacional \rightarrow ==$
 $op_relacional \rightarrow !=$
 $op_mult \rightarrow *$
 $op_mult \rightarrow /$
 $op_mult \rightarrow \%$

1.2. Acondicionamiento

Acondicionamos la gramática definida en la sección anterior. Ésto, con el fin de implementar un analizador sintáctico descendente predictivo recursivo.

$\text{programa} \rightarrow \text{bloque}$
 $\text{bloque} \rightarrow \{\text{seccion_declaraciones_opt seccion_instrucciones_opt}\}$

1.2.1. Declaraciones

$\text{seccion_declaraciones_opt} \rightarrow \text{seccion_declaraciones} \&\&$
 $\text{seccion_declaraciones_opt} \rightarrow \epsilon$
 $\text{seccion_declaraciones} \rightarrow \text{declaracion resto_sd}$
 $\text{resto_sd} \rightarrow ; \text{declaracion resto_sd}$
 $\text{resto_sd} \rightarrow \epsilon$
 $\text{declaracion} \rightarrow \text{tipo_nombre}$

1.2.2. Tipos

$\text{tipo_nombre} \rightarrow \text{tipo_base identificador}$
 $\text{tipo_base} \rightarrow \text{int}$
 $\text{tipo_base} \rightarrow \text{real}$
 $\text{tipo_base} \rightarrow \text{bool}$

1.2.3. Instrucciones

$\text{seccion_instrucciones_opt} \rightarrow \text{seccion_instrucciones}$
 $\text{seccion_instrucciones_opt} \rightarrow \epsilon$
 $\text{seccion_instrucciones} \rightarrow \text{lista_instrucciones}$
 $\text{lista_instrucciones} \rightarrow \text{instruccion resto_li}$
 $\text{resto_li} \rightarrow ; \text{instruccion resto_li}$
 $\text{resto_li} \rightarrow \epsilon$
 $\text{instruccion} \rightarrow @ \text{expresion}$

1.2.4. Expresiones

$expresion \rightarrow E0$
 $E0 \rightarrow E1 \text{ resto_} E0$
 $resto_E0 \rightarrow = E0$
 $resto_E0 \rightarrow \epsilon$
 $E1 \rightarrow E2 \text{ resto_} E1$
 $resto_E1 \rightarrow op_relacional E2 \text{ resto_} E1$
 $resto_E1 \rightarrow \epsilon$
 $E2 \rightarrow E3 \text{ resto_} E2_F \text{ resto_} E2_R$
 $resto_E2_R \rightarrow + E3 \text{ resto_} E2_R$
 $resto_E2_R \rightarrow \epsilon$
 $resto_E2_F \rightarrow - E3$
 $resto_E2_F \rightarrow \epsilon$
 $E3 \rightarrow E4 \text{ resto_} E3$
 $resto_E3 \rightarrow \text{and } E3$
 $resto_E3 \rightarrow \text{or } E4$
 $resto_E3 \rightarrow \epsilon$
 $E4 \rightarrow E5 \text{ resto_} E4$
 $resto_E4 \rightarrow op_mult E5 \text{ resto_} E4$
 $resto_E4 \rightarrow \epsilon$
 $E5 \rightarrow - E5$
 $E5 \rightarrow \text{not } E5$
 $E5 \rightarrow E6$
 $E6 \rightarrow expresion_basica$
 $E6 \rightarrow (E0)$
 $expresion_basica \rightarrow \text{literalEntero}$
 $expresion_basica \rightarrow \text{literalReal}$
 $expresion_basica \rightarrow \text{true}$
 $expresion_basica \rightarrow \text{false}$
 $expresion_basica \rightarrow \text{identificador}$

1.2.5. Operadores

$op_relacional \rightarrow <$
 $op_relacional \rightarrow <=$
 $op_relacional \rightarrow >$
 $op_relacional \rightarrow >=$
 $op_relacional \rightarrow ==$
 $op_relacional \rightarrow !=$
 $op_mult \rightarrow *$
 $op_mult \rightarrow /$
 $op_mult \rightarrow \%$

1.3. Directores

Directores de cada regla de la gramática acondicionada

1.3.1. Tabla de Reglas

Cuadro 1.3.1: Directores de las reglas de la gramática

Regla	Directores	Anulable
$\text{programa} \rightarrow \text{bloque}$	{	No
$\text{bloque} \rightarrow \{ \text{declaraciones instrucciones} \}$	{	No
$\text{declaraciones} \rightarrow \text{lista-declaraciones} \&\&$	int real bool	No
$\text{declaraciones} \rightarrow \epsilon$		Sí
$\text{lista-declaraciones} \rightarrow \text{declaracion } r\text{-lista-declaraciones}$	int real bool	No
$r\text{-lista-declaraciones} \rightarrow ; \text{declaracion } r\text{-lista-declaraciones}$;	No
$r\text{-lista-declaraciones} \rightarrow \epsilon$		Sí
$\text{declaracion} \rightarrow \text{tipo } \mathbf{identificador}$	bool int real	No
$\text{tipo} \rightarrow \mathbf{bool}$	bool	No
$\text{tipo} \rightarrow \mathbf{int}$	int	No
$\text{tipo} \rightarrow \mathbf{real}$	real	No
$\text{instrucciones} \rightarrow \text{lista-instrucciones}$	@	No
$\text{instrucciones} \rightarrow \epsilon$		Sí
$\text{lista-instrucciones} \rightarrow \text{instruccion } r\text{-lista-instrucciones}$	@	No
$r\text{-lista-instrucciones} \rightarrow ; \text{instruccion } r\text{-lista-instrucciones}$;	No
$r\text{-lista-instrucciones} \rightarrow \epsilon$		Sí
$\text{instruccion} \rightarrow \text{eval}$	@	No
$\text{eval} \rightarrow @ E0$	@	No
$E0 \rightarrow E1 RE0$	+ - true false literalEntero literalReal identificador (No
$RE0 \rightarrow = E0$	=	No
$RE0 \rightarrow \epsilon$		Sí
$E1 \rightarrow E2 RE1$	+ - true false literalEntero literalReal identificador (No
$RE1 \rightarrow OP1 E2 RE1$	< <= > >= == !=	No
$RE1 \rightarrow \epsilon$		Sí
$E2 \rightarrow E3 RE2 RE2'$	+ - true false literalEntero literalReal identificador (No
$RE2 \rightarrow - E3$	-	No
$RE2 \rightarrow \epsilon$		Sí
$RE2' \rightarrow + E3 RE2'$	+	No
$RE2' \rightarrow \epsilon$		Sí
$E3 \rightarrow E4 RE3$	+ - true false literalEntero literalReal identificador (No

Continúa en la siguiente página

Cuadro 1.3.1: Directores de las reglas de la gramática (Continuación)

Regla	Directores	Anulable
$RE3 \rightarrow \text{and } E3$	and	No
$RE3 \rightarrow \text{or } E4$	or	No
$RE3 \rightarrow \epsilon$		Sí
$E4 \rightarrow E5 \text{ } RE4$	+ - true false literalEntero literalReal identificador (No
$RE4 \rightarrow OP4 \text{ } E5 \text{ } RE4$	* /	No
$RE4 \rightarrow \epsilon$		Sí
$E5 \rightarrow OP5 \text{ } E5$	+ -	No
$E5 \rightarrow E6$	true false literalEntero literalReal identificador (No
$E6 \rightarrow \text{true}$	true	No
$E6 \rightarrow \text{false}$	false	No
$E6 \rightarrow \text{literalEntero}$	literalEntero	No
$E6 \rightarrow \text{literalReal}$	literalReal	No
$E6 \rightarrow \text{identificador}$	identificador	No
$E6 \rightarrow (E0)$	(No
$OP1 \rightarrow <$	<	No
$OP1 \rightarrow <=$	<=	No
$OP1 \rightarrow >$	>	No
$OP1 \rightarrow >=$	>=	No
$OP1 \rightarrow ==$	==	No
$OP1 \rightarrow !=$!=	No
$OP4 \rightarrow *$	*	No
$OP4 \rightarrow /$	/	No
$OP5 \rightarrow +$	+	No
$OP5 \rightarrow -$	-	No

2 | Tiny

2.1. Especificación Sintáctica (Gramática)

Implementamos la gramática que define la especificación sintáctica del lenguaje Tiny empleando los patrones explicados en clase (Diseño descendente, Reutilización, Nivel de Abstracción Equilibrado, Opcionalidad, Variantes, Listas y Expresiones).

Para ello definimos primero la estructura básica de todo programa:

```
programa  $\rightarrow$  bloque
bloque  $\rightarrow$  {seccion_declaraciones_opt seccion_intrucciones_opt}
```

2.1.1. Declaraciones

```
seccion_declaraciones_opt  $\rightarrow$  seccion_declaraciones &&
seccion_declaraciones_opt  $\rightarrow$   $\epsilon$ 
seccion_declaraciones  $\rightarrow$  seccion_declaraciones ; declaracion
seccion_declaraciones  $\rightarrow$  declaracion
declaracion  $\rightarrow$  tipo_nombre
declaracion  $\rightarrow$  type tipo_nombre
declaracion  $\rightarrow$  proc identificador parametros_formales bloque
parametros_formales  $\rightarrow$  (lista_parametros_opt)
lista_parametros_opt  $\rightarrow$  lista_parametros
lista_parametros_opt  $\rightarrow$   $\epsilon$ 
lista_parametros  $\rightarrow$  lista_parametros , parametro
lista_parametros  $\rightarrow$  parametro
parametro  $\rightarrow$  tipo ref_opt identificador
ref_opt  $\rightarrow$  &
ref_opt  $\rightarrow$   $\epsilon$ 
```

2.1.2. Tipos

```
tipo_nombre  $\rightarrow$  tipo identificador
tipo  $\rightarrow$  tipo0
tipo0  $\rightarrow$  tipo0 tamano_opt
tipo0  $\rightarrow$  tipo1
tipo1  $\rightarrow$  ^tipo1
tipo1  $\rightarrow$  tipo_base
tipo_base  $\rightarrow$  struct lista_campos
tipo_base  $\rightarrow$  int
tipo_base  $\rightarrow$  real
tipo_base  $\rightarrow$  bool
tipo_base  $\rightarrow$  string
tipo_base  $\rightarrow$  identificador
tamano_opt  $\rightarrow$  [literalEntero]
tamano_opt  $\rightarrow$   $\epsilon$ 
lista_campos  $\rightarrow$  lista_campos , tipo_nombre
lista_campos  $\rightarrow$  tipo_nombre
```

2.1.3. Instrucciones

```
seccion_intrucciones_opt  $\rightarrow$  seccion_intrucciones
seccion_intrucciones_opt  $\rightarrow$   $\epsilon$ 
```


seccion_instrucciones \rightarrow *lista_instrucciones*
lista_instrucciones \rightarrow *lista_instrucciones ; instruccion*
lista_instrucciones \rightarrow *instruccion*
instruccion \rightarrow @ *expresion*
instruccion \rightarrow *if_ins*
instruccion \rightarrow *if_ins else_ins*
instruccion \rightarrow **while** *exp_bloque*
instruccion \rightarrow **read** *expresion*
instruccion \rightarrow **write** *expresion*
instruccion \rightarrow **nl**
instruccion \rightarrow **new** *expresion*
instruccion \rightarrow **delete** *expresion*
instruccion \rightarrow **call** **identificador** *parametros_reales*
instruccion \rightarrow *bloque*
if_ins \rightarrow **if** *exp_bloq*
else_ins \rightarrow **else** *bloque*
exp_bloq \rightarrow *expresion* *bloque*
parametros_reales \rightarrow (*lista_expresiones_opt*)
lista_expresiones_opt \rightarrow *lista_expresiones*
lista_expresiones_opt \rightarrow ϵ
lista_expresiones \rightarrow *lista_expresiones , expresion*
lista_expresiones \rightarrow *expresion*

2.1.4. Expresiones

expresion \rightarrow *E0*
E0 \rightarrow *E1 = E0*
E0 \rightarrow *E1*
E1 \rightarrow *E1 op_relacional E2*
E1 \rightarrow *E2*
E2 \rightarrow *E2 + E3*
E2 \rightarrow *E3 - E3*
E2 \rightarrow *E3*
E3 \rightarrow *E4 and E3*
E3 \rightarrow *E4 or E4*
E3 \rightarrow *E4*
E4 \rightarrow *E4 op_mult E5*
E4 \rightarrow *E5*
E5 \rightarrow - *E5*
E5 \rightarrow **not** *E5*
E5 \rightarrow *E6*
E6 \rightarrow *E6 op_dirs*
E6 \rightarrow *E7*
E7 \rightarrow *expresion_basica*
E7 \rightarrow (*E0*)
expresion_basica \rightarrow **literalEntero**
expresion_basica \rightarrow **literalReal**
expresion_basica \rightarrow **true**
expresion_basica \rightarrow **false**
expresion_basica \rightarrow **literalCadena**
expresion_basica \rightarrow **identificador**
expresion_basica \rightarrow **null**

2.1.5. Operadores

op_relacional \rightarrow <
op_relacional \rightarrow <=
op_relacional \rightarrow >
op_relacional \rightarrow >=

```

op_relacional  $\rightarrow$  ==
op_relacional  $\rightarrow$  !=
op_mult  $\rightarrow$  *
op_mult  $\rightarrow$  /
op_mult  $\rightarrow$  %
op_dirs  $\rightarrow$  [expresion]
op_dirs  $\rightarrow$  . identificador
op_dirs  $\rightarrow$  ^

```

2.2. Acondicionamiento

Acondicionamos la gramática definida en la sección anterior. Ésto, con el fin de implementar un analizador sintáctico descendente predictivo recursivo.

```

programa  $\rightarrow$  bloque
bloque  $\rightarrow$  {seccion_declaraciones_opt seccion_intrucciones_opt}

```

2.2.1. Declaraciones

```

seccion_declaraciones_opt  $\rightarrow$  seccion_declaraciones &&
seccion_declaraciones_opt  $\rightarrow$   $\epsilon$ 
seccion_declaraciones  $\rightarrow$  declaracion resto_sd
resto_sd  $\rightarrow$  ; declaracion resto_sd
resto_sd  $\rightarrow$   $\epsilon$ 
declaracion  $\rightarrow$  tipo_nombre
declaracion  $\rightarrow$  type tipo_nombre
declaracion  $\rightarrow$  proc identificador parametros_formales bloque
parametros_formales  $\rightarrow$  (lista_parametros_opt)
lista_parametros_opt  $\rightarrow$  lista_parametros
lista_parametros_opt  $\rightarrow$   $\epsilon$ 
lista_parametros  $\rightarrow$  parametro resto_lp
resto_lp  $\rightarrow$  , parametro resto_lp
resto_lp  $\rightarrow$   $\epsilon$ 
parametro  $\rightarrow$  tipo ref_opt identificador
ref_opt  $\rightarrow$  &
ref_opt  $\rightarrow$   $\epsilon$ 

```

2.2.2. Tipos

```

tipo_nombre  $\rightarrow$  tipo identificador
tipo  $\rightarrow$  tipo0
tipo0  $\rightarrow$  tipo0 tamano_opt
tipo0  $\rightarrow$  tipo1
tipo1  $\rightarrow$  ^tipo1
tipo1  $\rightarrow$  tipo_base
tipo_base  $\rightarrow$  struct lista_campos
tipo_base  $\rightarrow$  int
tipo_base  $\rightarrow$  real
tipo_base  $\rightarrow$  bool
tipo_base  $\rightarrow$  string
tipo_base  $\rightarrow$  identificador
tamano_opt  $\rightarrow$  [literalEntero]
tamano_opt  $\rightarrow$   $\epsilon$ 
lista_campos  $\rightarrow$  tipo_nombre resto_lc
resto_lc  $\rightarrow$  , tipo_nombre resto_lc
resto_lc  $\rightarrow$   $\epsilon$ 

```

2.2.3. Instrucciones

$seccion_instrucciones_opt \rightarrow seccion_instrucciones$
 $seccion_instrucciones_opt \rightarrow \epsilon$
 $seccion_instrucciones \rightarrow lista_instrucciones$
 $lista_instrucciones \rightarrow instruccion\ resto_li$
 $resto_li \rightarrow ;\ instruccion\ resto_li$
 $resto_li \rightarrow \epsilon$
 $instruccion \rightarrow @\ expresion$
 $instruccion \rightarrow if_ins\ resto_ii$
 $resto_ii \rightarrow else_ins$
 $resto_ii \rightarrow \epsilon$
 $instruccion \rightarrow \textbf{while}\ exp_bloque$
 $instruccion \rightarrow \textbf{read}\ expresion$
 $instruccion \rightarrow \textbf{write}\ expresion$
 $instruccion \rightarrow \textbf{nl}$
 $instruccion \rightarrow \textbf{new}\ expresion$
 $instruccion \rightarrow \textbf{delete}\ expresion$
 $instruccion \rightarrow \textbf{call}\ identificador\ parametros_reales$
 $instruccion \rightarrow bloque$
 $if_ins \rightarrow \textbf{if}\ exp_bloq$
 $else_ins \rightarrow \textbf{else}\ bloque$
 $exp_bloq \rightarrow expresion\ bloque$
 $parametros_reales \rightarrow (lista_expresiones_opt)$
 $lista_expresiones_opt \rightarrow lista_expresiones$
 $lista_expresiones_opt \rightarrow \epsilon$
 $lista_expresiones \rightarrow expresion\ resto_le$
 $resto_le \rightarrow ,\ expresion\ resto_le$
 $resto_le \rightarrow \epsilon$

2.2.4. Expresiones

$expresion \rightarrow E0$
 $E0 \rightarrow E1\ resto_E0$
 $resto_E0 \rightarrow =\ E0$
 $resto_E0 \rightarrow \epsilon$
 $E1 \rightarrow E2\ resto_E1$
 $resto_E1 \rightarrow op_relacional\ E2\ resto_E1$
 $resto_E1 \rightarrow \epsilon$
 $E2 \rightarrow E3\ resto_E2_F\ resto_E2_R$
 $resto_E2_R \rightarrow +\ E3\ resto_E2_R$
 $resto_E2_R \rightarrow \epsilon$
 $resto_E2_F \rightarrow -\ E3$
 $resto_E2_F \rightarrow \epsilon$
 $E3 \rightarrow E4\ resto_E3$
 $resto_E3 \rightarrow \textbf{and}\ E3$
 $resto_E3 \rightarrow \textbf{or}\ E4$
 $resto_E3 \rightarrow \epsilon$
 $E4 \rightarrow E5\ resto_E4$
 $resto_E4 \rightarrow op_mult\ E5\ resto_E4$
 $resto_E4 \rightarrow \epsilon$
 $E5 \rightarrow -\ E5$
 $E5 \rightarrow \textbf{not}\ E5$
 $E5 \rightarrow E6$
 $E6 \rightarrow E7\ resto_E6$
 $resto_E6 \rightarrow op_dirs\ resto_E6$
 $resto_E6 \rightarrow \epsilon$
 $E7 \rightarrow expresion_basica$
 $E7 \rightarrow (E0)$
 $expresion_basica \rightarrow \textbf{literalEntero}$

$expresion_basica \rightarrow \text{literalReal}$
 $expresion_basica \rightarrow \text{true}$
 $expresion_basica \rightarrow \text{false}$
 $expresion_basica \rightarrow \text{literalCadena}$
 $expresion_basica \rightarrow \text{identificador}$
 $expresion_basica \rightarrow \text{null}$

2.2.5. Operadores

$op_relacional \rightarrow <$
 $op_relacional \rightarrow <=$
 $op_relacional \rightarrow >$
 $op_relacional \rightarrow >=$
 $op_relacional \rightarrow ==$
 $op_relacional \rightarrow !=$
 $op_mult \rightarrow *$
 $op_mult \rightarrow /$
 $op_mult \rightarrow \%$
 $op_dirs \rightarrow [expresion]$
 $op_dirs \rightarrow \text{.identificador}$
 $op_dirs \rightarrow ^$

Índice de cuadros

1.3.1.Directores de las reglas de la gramática	5
--	---