

PRIMERA FASE: ANALIZADOR LÉXICO TINY(1)

*Javier Gómez Moraleda
Mario Quiñones Pérez
Grupo 18*

1. ENUMERACIÓN DE LAS CLASES LÉXICAS DE TINY(1)

Clase léxica	Descripción
ID	Identificador para variables.
LETRA	Letra mayúscula o minúscula.
DIGITO	Dígito entre 0 y 9.
DIGITO_POS	Dígito entre 1 y 9.
SIGNO	+ o -.
LIT_ENT	Comienza por SIGNO opcionalmente. Seguido aparece una secuencia de 1 o más dígitos (no se admiten ceros no significativos a la izquierda).
PDEC	Comienza con un punto, seguido de una secuencia de 1 o más dígitos (no se admiten ceros no significativos a la derecha).
PEXP	Comienza con e o E, seguida de un exponente que es igual que los números enteros.
LIT_REAL	Parte entera seguida de bien una parte decimal, bien una parte exponencial, o bien una parte decimal seguida de una parte exponencial.
LIT_CAD	Cadena de caracteres que empiezan y terminan por comillas dobles “ y que entre ambas hay cualquier carácter distinto de “, retroceso (\b), retorno de carro (\r), y salto de línea (\n).
AMP	Símbolo &
DAMP	Separador entre la sección de declaraciones y la de instrucciones.
MAS	Operador aritmético binario +.
MENOS	Operador aritmético binario -.
POR	Operador aritmético binario *.
DIV	Operador aritmético binario /.
PORCENT	Operador aritmético binario %.
MENOR	Operador relacional <.

MAYOR	Operador relacional >.
MENOR_IGUAL	Operador relacional <=.
MAYOR_IGUAL	Operador relacional >=.
IGUAL	Operador de asignación =.
DIGUAL	Operador relacional ==.
DIF	Operador relacional !=.
PAP	Paréntesis de apertura.
PCIERRE	Paréntesis de cierre.
CAP	Corchete de apertura.
CCIERRE	Corchete de cierre.
LLAP	Llave de apertura.
LLCIERRE	Llave de cierre.
PUNTO	Separador ‘.’.
COMA	Separador ,.
PCOMA	Separador ; entre cada instrucción o cada declaración.
FLECHA	Símbolo de flecha ->
EOF	Símbolo de final de fichero -.
INT	Palabra reservada para los números enteros (int).
REAL	Palabra reservada para los números reales (real).
BOOL	Palabra reservada para los booleanos (bool).
STRING	Palabra reservada para las cadenas de caracteres (string).
AND	Operador lógico binario and.
OR	Operador lógico binario or.
NOT	Operador lógico binario not.
NULL	Palabra reservada para la creación de una
FALSE	Expresión booleana false.

TRUE	Expresión booleana true.
PROC	Palabra reservada para la creación de un proceso.
IF	Palabra reservada if.
THEN	Palabra reservada then.
ELSE	Palabra reservada else.
ENDIF	Palabra reservada endif.
WHILE	Palabra reservada while.
DO	Palabra reservada do.
ENDWHILE	Palabra reservada endwhile.
CALL	Palabra reservada call.
RECORD	Palabra reservada record.
ARRAY	Palabra reservada array.
OF	Palabra reservada of.
POINTER	Palabra reservada pointer.
NEW	Palabra reservada new.
DELETE	Palabra reservada delete.
READ	Palabra reservada read.
WRITE	Palabra reservada write.
NL	Palabra reservada nl.
VAR	Palabra reservada var.
TYPE	Palabra reservada type.
SEPARADOR	Cadenas ignorables como espacios, tabuladores o saltos de línea.
COMENTARIO	Cadena de caracteres distintos de salto de línea que empiezan por #

2. ESPECIFICACIÓN FORMAL DEL LÉXICO DEL LENGUAJE MEDIANTE DEFINICIONES REGULARES

Clase léxica	=	Definición regular
ID	=	LETRA (LETRA + DIGITO + _)*
LETRA	=	[a-z, A-Z]
DIGITO	=	[0-9]
DIGITO_POS	=	[1-9]
SIGNO	=	(+ -)
LIT_ENT	=	SIGNO? ((DIGITO_POS (DIGITO)*) + 0)
PDEC	=	. (DIGITO* DIGITO_POS + 0)
PEXP	=	(e E) LIT_ENT
LIT_REAL	=	LIT_ENT (PDEC PEXP PDEC PEXP)
LIT_CAD	=	\ " (CARACTER)* \ "
AMP	=	\ &
DAMP	=	AMP AMP
MAS	=	\ +
MENOS	=	\ -
POR	=	\ *
DIV	=	/
PORCENT	=	\ %
MENOR	=	\ <
MAYOR	=	\ >
MENOR_IGUAL	=	MENOR IGUAL
MAYOR_IGUAL	=	MAYOR IGUAL
IGUAL	=	\ =

DIGUAL	=	IGUAL IGUAL
DIF	=	\! IGUAL
PAP	=	\(
PCIERRE	=	\)
CAP	=	\[
CCIERRE	=	\]
LLAP	=	\{
LLCIERRE	=	\}
PUNTO	=	\.
COMA	=	\,
PCOMA	=	\;
FLECHA	=	\->
EOF	=	-
INT	=	int
REAL	=	real
BOOL	=	bool
STRING	=	string
AND	=	and
OR	=	or
NOT	=	not
NULL	=	null
FALSE	=	false
TRUE	=	true
PROC	=	proc
IF	=	if

THEN	=	then
ELSE	=	else
ENDIF	=	endif
WHILE	=	while
DO	=	do
ENDWHILE	=	endwhile
CALL	=	call
RECORD	=	record
ARRAY	=	array
OF	=	of
POINTER	=	pointer
NEW	=	new
DELETE	=	delete
READ	=	read
WRITE	=	write
NL	=	nl
VAR	=	var
TYPE	=	type
SEPARADOR	=	“ ” + \t + \n + \r + \b
COMENTARIO	=	#[^\n]*