

Programación III

Teoría y Práctica de Compilación

ISC-314

Descripción Léxica del Lenguaje Mini-P

Objetivos Generales

Este documento provee una descripción general de los aspectos léxicos del lenguaje Mini-P. Se espera que, en base a este documento, el estudiante pueda producir la especificación léxica de Mini-P en Flex/Lex.

Generalidades de Mini-P

1. Mini-P no diferencia entre minúsculas y mayúsculas, es decir, **no** es *case-sensitive*
2. Los comentarios en Mini-P están delimitados por “{“ y “}”. Un comentario no cerrado **debe** producir un error léxico
3. Las constantes de cadena en Mini-P están delimitadas por comillas dobles. Una constante de cadena sin comillas al final **debe** producir un error léxico
4. Los identificadores en Mini-P siguen las mismas reglas que los identificadores en C/C++ y JAVA con la excepción de que sólo se admiten caracteres alfanuméricos y de suscrito ()
5. Las constantes para números reales (de coma flotante) no necesariamente requieren una parte entera o una parte decimal, pero debe aparecer el punto decimal siempre y cuando el número tenga una mantisa.
6. Mini-P soporta notación científica o notación E.

Tabla de Unidades Léxicas (Tokens)

Esta tabla muestra la relación entre los lexemas representativos (*token string*), las constantes nombradas (*token name*) y los valores de dichas constantes (*token code*) correspondientes a cada unidad léxica/token del lenguaje Mini-P. Para las cuatro primeras unidades léxicas (ID, ICONST, RCONST, ECONST, LITERAL) se espera que el estudiante especifique las expresiones regulares que describen dichos tokens.

A partir de la tercera columna, se enumeran las unidades léxicas que corresponden con las palabras claves del lenguaje Mini-P.

token string	token name	token code	token string	token name	token code
<identifier>	_ID	1	program	_PROGRAM	51
<integer const>	_ICONST	2	endvars	_ENDVARS	52
<real const>	_RCONST	3	endprocs	_ENDPROCS	53
<Enot const>	_ECONST	4	variables	_VARS	54
<literal>	_LITERAL	5	novariables	_NOVARS	55
[_LBRACK	10	integer	_INTEGER	56
]	_RBRACK	11	real	_REAL	57
(_LPAREN	12	string	_STRING	58
)	_RPAREN	13	procedures	_PROCS	59
;	_SEMI	14	noprocedures	_NOPROCS	60
:	_COLON	15	procedure	_PROC	61
,	_COMMA	16	begin	_BEGIN	62
:=	_ASSIGN	17	end	_END	63
+	_PLUS	20	for	_FOR	64
-	_MINUS	21	to	_TO	65
*	_MULT	22	do	_DO	66
/	_DIVIDE	23	if	_IF	67
			then	_THEN	68
=	_EQL	40	else	_ELSE	69
<	_LESS	41	read	_READ	70
>	_GTR	42	write	_WRITE	71
<=	_LEQ	43			
>=	_GEQ	44			
<>	_NEQ	45			