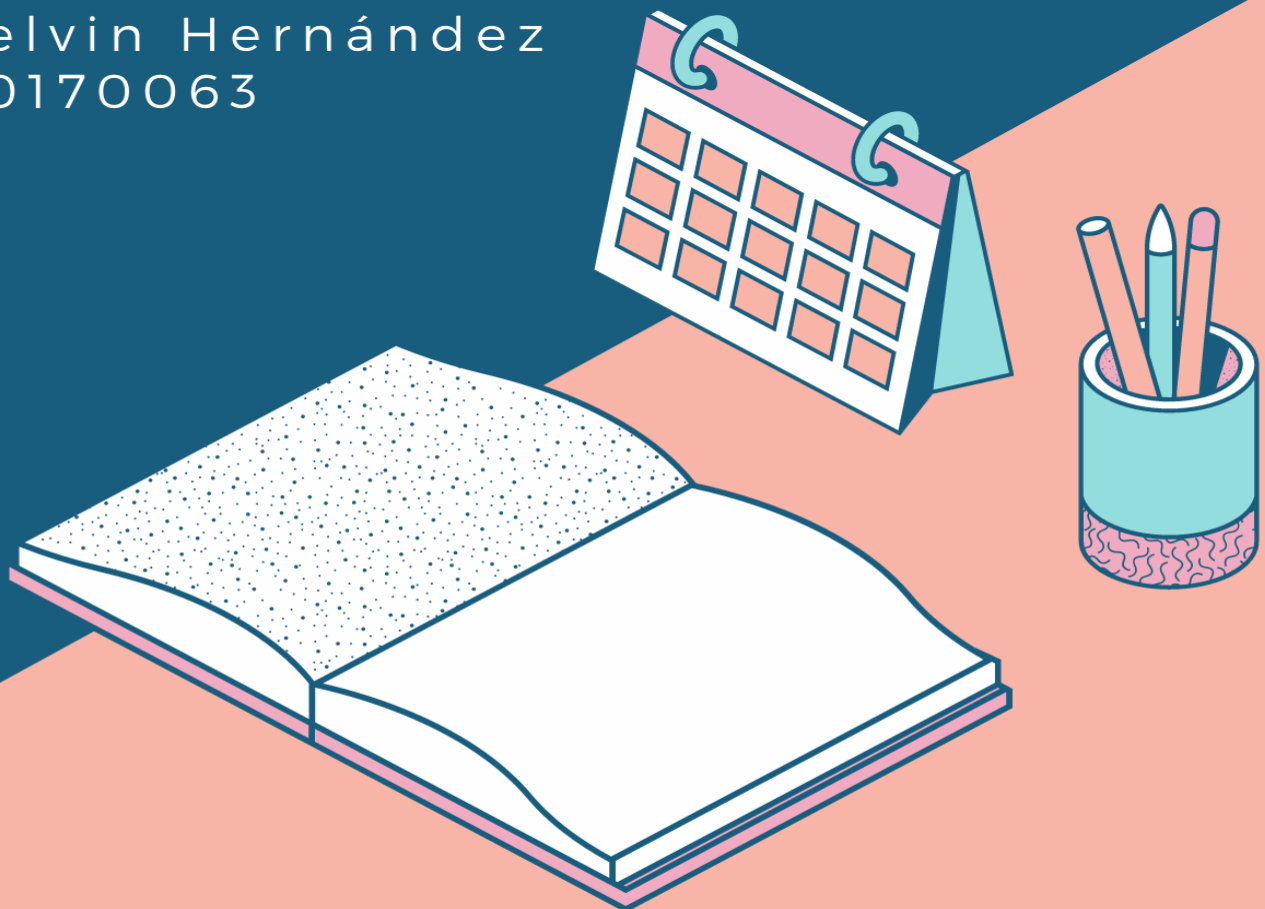


MANUAL DE GRAMATICA SYSCOMPILER

OLC-1C

Selvin Hernández
20170063



Ing. Kevin Lajpop
Aux. Sandra Eunice Jimenez

EXPRESIONES REGULARES	
TOKEN	ER
IDENTIFICADOR	[a-zñA-ZÑ][a-zñA-ZÑ0-9_]*
DECIMAL	[0-9]+(".[0-9]+)\b
ENTERO	0-9]+
escapechar, escapechar2	[\\"\\ntr]
escape	\\{escapechar}
aceptación, aceptacion2	[^"\\]
CADENA	(\"({escape} {aceptacion})*\")
escape2	\\{escapechar2}
caracter	(\"({escape2} {aceptacion2})*\")

Lista de No Terminales	
NO.	No terminal
1	instrucciones
2	instrucción
3	declaración
4	incre
5	decre
6	asignaciones
7	sentenciaif
8	sentenciaswitch
9	sentenciawhile
10	setenciadowhile
11	sentenciafor
13	sentcontinue
14	sentbreak
15	sentreturn
16	funciones
17	métodos
18	llamada
19	startwith
20	writeline
21	declalista
22	lista_append
23	set_lista
24	decla_vector
25	set_vector
26	lista_ids
27	lista_exp

GRAMATICA SYSCOMPILER	
PRODUCCIONES	DESCRIPCION
inicio-> instrucciones	
instrucciones-> instrucciones instruccion instruccion	la producción instrucciones es la encargada de generar una lista de instrucciones que permiten al interprete actuar como tal, por medio de la recursividad de la gramatica
instruccion-> declaracion ID INCRE PYC ID DECRE PYC asignaciones PYC sentenciaif sentenciaswitch sentenciawhile sentenciadowhile sentenciafor sentbreak sentcontinue sentreturn funciones metodos llamadas PYC startwith writeline declalista lista_append set_lista decla_vector set_vector	
declaracion -> tipo lista_ids IGUAL exp PYC tipo lista_ids PYC	esta producción permite comprobar la sintaxis de una declaración de una variable o lista de variables
tipo -> INT DOUBLE STRING CHAR BOOLEAN	
lista_ids -> lista_ids COMA ID ID	
asignaciones -> ID IGUAL exp	Esta expresión permite corroborar la sintaxis de una asignación de una variable
declalista -> RDYNAMICLIST MENORQUE tipo MAYORQUE ID IGUAL RNEW RDYNAMICLIST MENORQUE tipo MAYORQUE PYC	

<p> RDYNAMICLIST MENORQUE tipo MAYORQUE ID IGUAL RTOCHARARRAY PARA exp PARC PYC</p> <p>lista_append-> RAPPEND PARA ID COMA exp PARC PYC</p> <p>get_lista -> RGETVALUE PARA ID COMA exp PARC</p> <p>set_lista -> RSETVALUE PARA ID COMA exp COMA exp PARC PYC</p>	
<p>decla_vector-> tipo ID CORA CORC IGUAL RNEW tipo CORA exp CORC PYC tipo ID CORA CORC IGUAL LLAVEA lista_exp LLAVEC PYC</p> <p>get_vector-> ID CORA exp CORC</p> <p>set_vector-> ID CORA exp CORC IGUAL exp PYC</p>	esta producción se encarga de verificar la sintaxis de la declaración de un vectors
<p>sentenciaif -> RIF PARA exp PARC LLAVEA instrucciones LLAVEC RIF PARA exp PARC LLAVEA instrucciones LLAVEC RELSE LLAVEA instrucciones LLAVEC RIF PARA exp PARC LLAVEA instrucciones LLAVEC RELSE sentenciaif</p>	esta se encarga de verificar la sintaxis de la sentencia de control if
<p>sentenciaswitch -> RSWITCH PARA exp PARC LLAVEA casos LLAVEC</p> <p>casos -> casos caso caso</p> <p>caso -> RCASE exp RDOSPTS instrucciones RDEFAULT RDOSPTS instrucciones</p>	
sentenciawhile -> RWHILE PARA exp PARC LLAVEA instrucciones LLAVEC	
sentbreak-> RBREAK PYC	
sentcontinue-> RCONTINUE PYC	

sentreturn-> RRETURN exp PYC RRETURN PYC	
writeline -> WRITELINE PARA exp PARC PYC sentenciadowhile -> RDO LLAVEA instrucciones LLAVEC RWHILE PARA exp PARC PYC	
sentenciafor -> RFOR PARA declaracion_for PYC exp PYC contador_for PARC LLAVEA instrucciones LLAVEC declaracion_for-> tipo ID IGUAL exp ID IGUAL exp contador_for-> ID INCRE ID DECRE ID IGUAL exp	
funciones -> tipo ID PARA lista_params PARC LLAVEA instrucciones LLAVEC tipo ID PARA PARC LLAVEA instrucciones LLAVEC RVOID ID PARA lista_params PARC LLAVEA instrucciones LLAVEC RVOID ID PARA PARC LLAVEA instrucciones LLAVEC lista_params -> lista_params COMA tipo ID tipo ID lista_params COMA RDYNAMICLIST MENORQUE tipo MAYORQUE ID RDYNAMICLIST MENORQUE tipo MAYORQUE ID llamadas -> ID PARA lista_exp PARC ID PARA PARC lista_exp-> lista_exp COMA exp exp	

exp -> exp MAS exp exp MENOS exp exp MULTI exp exp DIV exp exp POT exp exp MOD exp exp MAYORIGUAL exp exp MAYORQUE exp exp MENORIGUAL exp exp MENORQUE exp exp IGUALIGUAL exp exp DIFERENTE exp exp AND exp exp OR exp NOT exp MENOS exp %prec UMINUS PARA exp PARC DECIMAL ENTERO ID CADENA CHARACTER TRUE FALSE DOBLECOMILLAS PARA tipo PARC exp %prec CAST ID INCRE ID DECRE llamadas RTOLOWER PARA exp PARC RToupper PARA exp PARC RLENGTH PARA exp PARC RTRUNCATE PARA exp PARC RROUND PARA exp PARC RTYPEOF PARA exp PARC RTOSTRING PARA exp PARC RTOCHARARRAY PARA exp PARC get_lista get_vector exp TERNARIO exp RDOSPTS exp	<p>esta producción permite verificar que la sintaxis de expresiones por ejemplo exp+exp, exp-exp, exp*exp, etc, esta por ser un tipo ambigua se debe de definir la precedencia de los operadores para que no haya un erro de desplazamiento – reducción en el análisis al momento de construir la tabla de análisis sintáctico.</p>
startwith-> RSTART RWITH llamadas PYC	