

Emiliano José Alexander Velásquez Najera

Carne: 201903848

Gramática

JPR Editor

Palabras Reservadas

Palabra : 'Referencia dentro del documento'

'int' : 'RINT',

'double' : 'RDOUBLE',

'string': 'RSTRING',

'char' : 'RCHAR',

'boolean' : 'RBOOL',

'print' : 'RPRINT',

'var' : 'RVAR',

'null' : 'RNULL',

'if' : 'RIF',

'else' : 'RELSE',

'true' : 'RTRUE',

'false' : 'RFALSE',

'switch': 'RSWITCH',

'case' : 'RCASE',

'break' : 'RBREAK',

'default' : 'RDEFAULT',

'while' : 'RWHILE',

'for' : 'RFOR',

'main' : 'RMAIN',

'func' : 'RFUNC',

'return': 'RRETURN',
'continue' : 'RCONTINUE',
'read' : 'RREAD',
'new' : 'RNEW',

Tokens:

Token = 'Lexema'

PUNTOCOMA = ';' ;

PARA = '\('

PARC = '\)'

LLAVEA = '{'

LLAVEC = '}'

COMA = ','

MAS = '\+'

MENOS = '-'

POR = '*'

DIV = '/'

POT = '**'

MOD = '%'

MENORQUE = '<'

MAYORQUE = '>'

DIFERENTE = '!='

IGUALIGUAL == '=='

MENORIGUAL = '<='

MAYORIGUAL = '>='

IGUAL = '='

AND = '&&'

OR = '\\|\\|'

NOT = '!'

INC = '\+\'

DEC = '--'

DOSPUNTOS = ':'

CORA = '\['

CORC = $r' \setminus]'$

Producciones Terminales:

Token -> 'Lexema'

Decimal -> '\d+\.\d+'

Enterro -> '\d+'

ID-> '[a-zA-Z][a-zA-Z_0-9]*'

[illegible]

Carácter -> `"'(\[nN][\\\/*][tT]'\\"/>`"

Comentario Multilínea -> `'\#*(.|\\n)*?*\'#'`

Comentario Simple -> `'\#.*\'`

Nueva Línea -> `'\\n+'`

Producciones No Terminales:

Producción -> Patron

Inicio -> instrucciones

Instrucciones -> instrucción

Instrucción -> imprimir_instr finins

| declaracion_instr finins

| asignacion_instr finins

| if_instr

| while_instr

| break_instr finins

| for_instr

| switch_instr

| main_instr

- | funcion_instr
- | llamada_instr finins
- | return_instr finins
- | continue_instr finins
- | declArr_instr finins
- | modArr_instr finins

finis -> PUNTOCOMA

- |

imprimir_instr -> RPRINT PARA expresion PARC

declaracion_instr -> RVAR ID IGUAL expresion

- | RVAR ID

declArr_instr -> tipo1

tipo1 -> tipo lista_Dim ID IGUAL RNEW tipo lista_expresiones

lista_Dim -> lista_Dim CORA CORC

lista_expresiones -> lista_expresiones CORA expresion COR

lista_expresiones -> CORA expresion CORC

modArr_instr -> ID lista_expresiones IGUAL expresión

asignacion_instr -> ID IGUAL expresion

| ID INC

| ID DEC

expresion -> ID lista_expresiones

tipo -> RINT

| RDOUBLE

| RSTRING

| RCHAR

| RBOOL

if_instr -> RIF PARA expresion PARC LLAVEA instrucciones LLAVEC

| RIF PARA expresion PARC LLAVEA instrucciones LLAVEC RELSE LLAVEA instrucciones LLAVEC

| RIF PARA expresion PARC LLAVEA instrucciones LLAVEC RELSE if_instr

while_instr -> RWHILE PARA expresion PARC LLAVEA instrucciones LLAVEC

for_instr -> RFOR PARA declaracion_instr PUNTOCOMA expresion PUNTOCOMA asignacion_instr PARC LLAVEA instrucciones LLAVEC
| RFOR PARA asignacion_instr PUNTOCOMA expresion PUNTOCOMA asignacion_instr PARC LLAVEA instrucciones LLAVEC

reak_instr -> RBREAK

switch_instr -> RSWITCH PARA expresion PARC LLAVEA cases LLAVEC

cases -> cases case
| case
| RCASE expresion DOSPUNTOS instrucciones
| RDEFAULT DOSPUNTOS instrucciones

expresion -> expresion MAS expresion
| expresion MENOS expresion
| expresion POR expresion
| expresion DIV expresion
| expresion POT expresion
| expresion MOD expresion
| expresion MENORQUE expresion
| expresion MAYORQUE expresion
| expresion IGUALIGUAL expresion
| expresion DIFERENTE expresion
| expresion MENORIGUAL expresion

| expresion MAYORIGUAL expresion

| expresion AND expresion

| expresion OR expresion

| MENOS expresion

| NOT expresion

| PARA expresion PARC

| ID

| llamada_inst

| ENTERO

| DECIMAL

| CADENA

| CARÁCTER

| RNULL

| RTRUE

| RFALSE

| RREAD PARA PARC

| PARA tipo PARC expresión

main_instr -> RMAIN PARA PARC LLAVEA instrucciones LLAVEC

funcion_instr -> RFUNC ID PARA parametros PARC LLAVEA instrucciones LLAVEC
| RFUNC ID PARA PARC LLAVEA instrucciones LLAVEC

parametros -> parametros COMA parámetro
| parámetro

parametro -> tipo ID

llamada_instr -> ID PARA PARC
| ID PARA parametros_llamada PARC

parametros_llamada -> parametros_llamada COMA parametro_llamada
| parametro_llamada

parametros_llamada -> expresión

continue_instr -> RCONTINUE