

Gramatica Proyecto 1

Bryan Gerardo Paez Morales

201700945

Numero	[0-9]+
Decimal	[0-9]+ "." [0-9]+
Cadena	"simbolos*"
Caracter	('{\$numero}') 'simbolos'
Boolean	true false

$G = \{T, N, S, P\}$

$T = \{+, *, /, \text{potencia}, \text{modulo}, \text{¿}, \text{?}, [,], -, ", " , ; , \text{mayor}, \text{menor}, \text{mayorigual}, \text{menorigual}, \text{esdiferente}, \text{esigual}, \text{or}, \text{and}, \text{not}, \text{)}, (, -, \text{numero}, \text{cadena}, \text{boolean}, \text{caracter}, \text{inicio}, \text{fin}, \text{ingresar}, \text{como}, \text{con_valor}, \text{si}, \text{enton}, \text{de_lo_contrario}, \text{o_si}, \text{fin_si}, \text{segun}, \text{hacer}, \text{fin_segun}, \text{para}, \text{hasta}, \text{con}, \text{incremental}, \text{fin_para}, \text{mientras}, \text{fin_mientras}, \text{hasta_que}, \text{repetir}, \text{retornar}, \text{<metodo>}, \text{fin_metodo}, \text{verdadero}, \text{falso}, \text{<funcion>}, \text{fin_funcion}, \text{con_parametros}, \text{ejecutar}, \text{imprimir}, \text{imprimir_nl}, \text{cadena}, \text{identificador}, \text{var_identificador}\}$

$N = \{\text{INICIO}, \text{COMIENZO}, \text{CODIGO}, \text{CONTENIDO}, \text{DECLARACION}, \text{TIPODATO}, \text{EXPRESIONVAL}, \text{INST_SI}, \text{OSI}, \text{DELOCONTRARIO}, \text{CONDELOSI}, \text{INST_SEGUN}, \text{CASO}, \text{CASOS}, \text{PRINT}, \text{PRINTLN}, \text{CICLOPARA}, \text{CICLOMIENTRAS}, \text{CICLOREPETIR}, \text{METODO}, \text{FUNCION}, \text{LLAMADA}, \text{LPARAMETROS}, \text{PARAMETROS}, \text{LVALORES}\}$

$S = \{\text{INICIO}\}$

$P = \text{Producciones:}$

$\text{<INICIO> ::= inicio <CODIGO> fin}$

$\quad | \text{inicio <CODIGO> <COMIENZO> fin}$

$\text{<COMIENZO> ::= <CONTENIDO>}$

$\quad | \text{<COMIENZO> <CONTENIDO>}$

$\text{<CONTENIDO> ::= <METODO>}$

$\quad | \text{<FUNCION>}$

$\text{<CODIGO> ::= INSTRUCCIONES}$

$\quad | \text{<CODIGO> INSTRUCCIONES}$

$\text{INSTRUCCIONES ::= <DECLARACION> ;}$

| ASIGNACION ;
| <INST_SI> fin_si
| <INST_SEGUN> fin_segun
| PRINT
| <PRINTLN>
| <CICLOPARA>
| <CICLOMIENTRAS>
| <CICLOREPETIR>
| <LLAMADA> ;
| retornar EXPRESION ;

PRINT ::= imprimir EXPRESION ;

<PRINTLN> ::= imprimir_nl EXPRESION ;

<DECLARACION> ::= ingresar NOMBRES como <TIPODATO> con_valor EXPRESION

NOMBRES ::= NOMBRE

| NOMBRES NOMBRE

NOMBRE ::= var_identificador coma

| var_identificador

<TIPODATO> ::= cadena

| numero

| boolean

| caracter

<EXPRESIONVAL> ::= digito

|decimal

|cadena

|caracter

|carAscii

|verdadero

|falso

ASIGNACION ::= NOMBRES -> EXPRESION

EXPRESION ::= EXPRESION + EXPRESION

|EXPRESION - EXPRESION

|EXPRESION *EXPRESION

|EXPRESION / EXPRESION

|EXPRESION potencia [EXPRESION]

|EXPRESION modulo EXPRESION

|EXPRESION mayor EXPRESION

|EXPRESION menor EXPRESION

|EXPRESION mayorigual EXPRESION

|EXPRESION menorigual EXPRESION

|EXPRESION esigual EXPRESION

|EXPRESION esdiferente EXPRESION

|EXPRESION or EXPRESION

|EXPRESION and EXPRESION

| - EXPRESION

|not EXPRESION

| (EXPRESION)

|var_identificador

|<EXPRESIONVAL>

|<LLAMADA>

<INST_SI> ::= si EXPRESION enton <CODIGO>

|si EXPRESION enton <CODIGO> de_lo_contrario <CODIGO>

|si EXPRESION enton <CODIGO> OSI

|si EXPRESION enton <CODIGO> <OSI>de_lo_contrario <CODIGO>

<OSI>::= <OSI><CONDELOSI>

|<CONDELOSI>

CONDEL<OSI>::= o_si EXPRESION enton <CODIGO>

<INST_SEGUN>::= segun var_identificador hacer <CASOS> de_lo_contrario enton <CODIGO>

|segun var_identificador hacer <CASOS>

|segun var_identificador hacer de_lo_contrario enton <CODIGO>

<CASOS>::= <CASOS> <CASO>

|<CASO>

<CASO> ::= ¿ <EXPRESIONVAL> ? enton <CODIGO>

<CICLOPARA>::= para var_identificador -> digito hasta EXPRESION hacer <CODIGO> fin_para

|para var_identificador -> digito hasta EXPRESION con incremental digito hacer
<CODIGO> fin_para

|para var_identificador -> digito hasta EXPRESION hacer fin_para

<CICLOMIENTRAS>::= mientras EXPRESION hacer <CODIGO> fin_mientras

|mientras EXPRESION hacer fin_mientras

<CICLOREPETIR>::= repetir <CODIGO> hasta_que EXPRESION

|repetir hasta_que EXPRESION

<METODO>::= metodo var_identificador <CODIGO> fin_metodo

|metodo var_identificador con_parametros (<LPARAMETROS>) <CODIGO> fin_metodo

<FUNCION>::= funcio var_identificador <TIPODATO> <CODIGO> fin_funcion

|funcion var_identificador <TIPODATO> con_parametros (<LPARAMETROS>) <CODIGO>
fin_funcion

<LLAMADA>::= ejecutar var_identificador ()

|ejecutar var_identificador (<LVALORES>)

<LPARAMETROS>::= <LPARAMETROS> "," PARAMETROS

|PARAMETROS

PARAMETROS::= var_identificador <TIPODATO>

<LVALORES>::= <LVALORES> "," EXPRESION

|EXPRESION