

GRAMATICA

%{

%}

%lex

%options case-insensitive

LETRAS [a-zA-ZñÑ]+

ENTERO [0-9]+

DECIMAL {ENTERO}{"."{ENTERO}}

ID {LETRAS}({LETRAS}|{ENTERO}|"_"*)

ESCAPECHAR [\\\"\\\nrt]

CESPECIALES \\{ESCAPECHAR}

ACEPTACION [^\"\\\n]

CADENA \"({CESPECIALES}|{ACEPTACION})*\"

ACEPTACIONC [^\"\\\n]

CARACTER \"({ACEPTACIONC}|{CESPECIALES})\"

%%

[\r\t\n\s]+ {}

"/".* {}

[/][*](\\n|[^\"*/"]|*|\"/\")*[*]/ {}

"+"	{ return 'inc' }
"--"	{ return 'dec' }
"=="	{ return 'igualigual' }
"!="	{ return 'diferente' }

"("	{ return 'parentesis' }
")"	{ return 'parentesisc' }
"["	{ return 'corchetea' }
"]"	{ return 'corchetec' }
"{"	{ return 'llavea' }
"}"	{ return 'llavec' }
","	{ return 'coma' }
","	{ return 'puntoycoma' }
"="	{ return 'igual' }
"?"	{ return 'interrogacion' }
":"	{ return 'dospuntos' }
"+"	{ return 'mas' }
"_"	{ return 'menos' }
"**"	{ return 'pow' }
"*"	{ return 'multi' }

```
"/"      { return 'div' }
%"       { return 'mod' }
"<="     { return 'menorigual' }
"<"      { return 'menorque' }
">="     { return 'mayorigual' }
">"      { return 'mayorque' }
```

```
"||"     { return 'or' }
"&&"     { return 'and' }
"!"      { return 'not' }
"^"      { return 'xor' }
```

```
"int"     { return 'int' }
"double"  { return 'double' }
"boolean" { return 'boolean' }
"char"    { return 'char' }
"string"  { return 'string' }
//para asignaciones
"const"   { return 'const' }
```

```
//resultado booleanos
"true"    { return 'true' }
"false"   { return 'false' }
```

```
//palabras para ciclos
"if"      { return 'if' }
"else"    { return 'else' }
"switch"  { return 'switch' }
"case"    { return 'case' }
"default" { return 'default' }
"while"   { return 'while' }
"do"      { return 'do' }
"for"     { return 'for' }
```

```
//palabras para funciones
"break"   { return 'break' }
"continue" { return 'continue' }
"return"  { return 'return' }
"println" { return 'println' }
"print"   { return 'print' }
"typeof"  { return 'typeof' }
"void"    { return 'void' }
"length"  { return 'length' }
"call"    { return 'call' }
```

//OTROS METODOS

```
"toLowerCase"    { return 'tolower'}
"toUpperCase"    { return 'toupper'}
"round"          { return 'round'}
"new"            { return 'new'}
"length"         { return 'length'}
"toArray"        { return 'toArray'}
"indexOf"        { return 'indexOf'}
"push"           { return 'push'}
"pop"            { return 'pop'}
"splice"         { return 'splice'}
"graficar_ts()"  { return 'graficar_ts'}
```

```
{ID}             { return 'id'}
{CADENA}          { return 'cadena'}
{DECIMAL}         { return 'decimal'}
{ENTERO}          { return 'entero'}
{CARACTER}        { return 'caracter'}
```

//EOF INDICA EL FIN DEL DOCUMENTO LEIDO

<<EOF>> return 'EOF';

. { //CAPTURA Y RECUPERACION DE ERRORES LEXICOS }

/lex

//Fin del analisis lexico

//INICIO DEL ANALISIS SINTACTICO

//PRECEDENCIA

```
%right 'interrogacion'
%left 'coma','puntoycoma', 'igual'
%left 'or'
%left 'and'
%right 'xor'
%left 'igualigual','diferente','menorque','menorigual','mayorque','mayorigual'
%left 'mas','menos'
%left 'div', 'multi','mod'
%right 'pow'
%left 'inc','dec'
%right 'not'
%right UMINUS
```

//INICIO DE LA GRAMATICA

%start INICIO

%%

INICIO: INSTRUCCIONES EOF

;

INSTRUCCIONES : INSTRUCCIONES INSTRUCCION

| INSTRUCCION

;

INSTRUCCION: ASIGNACION puntoycoma

| DECLARACION puntoycoma

| IF

| IF_SINLLAVES

| SWITCH

| FOR

| WHILE

| DO_WHILE

| break puntoycoma

| continue puntoycoma

| return puntoycoma

| return EXPRESION puntoycoma

| N_PRINT puntoycoma

| N_PRINTLN puntoycoma

| FUNCIONES

| METODOS

| LLAMADA puntoycoma

| BLOQUE_INST

| id inc puntoycoma

| id dec puntoycoma

| inc id puntoycoma

| dec id puntoycoma

| VECTOR puntoycoma

| MODIVECTOR puntoycoma

| TERNARIO puntoycoma

| PUSH_V puntoycoma

| POP_V puntoycoma

| SPLICE puntoycoma

| graficar_ts puntoycoma

| error puntoycoma

;

DECLARACION: TIPOVAR CONJID igual EXPRESION

| const TIPOVAR CONJID igual EXPRESION

| TIPOVAR CONJID

;

ASIGNACION: CONJID igual EXPRESION

;

CONJID: CONJID coma id
| id
;

TIPOVAR: int
| double
| boolean
| char
| string
;

TIPODATO: cadena
| character
| decimal
| id
| entero
| true
| false
;

IF: if parentesis EXPRESION parentesc BLOQUE_INST
| if parentesis EXPRESION parentesc BLOQUE_INST else BLOQUE_INST
| if parentesis EXPRESION parentesc BLOQUE_INST else IF
;

IF_SINLLAVES: if parentesis EXPRESION parentesc IFS_INSTRUCCION
| if parentesis EXPRESION parentesc IFS_INSTRUCCION else IFS_INSTRUCCION
| if parentesis EXPRESION parentesc IFS_INSTRUCCION else IF_SINLLAVES
;

IFS_INSTRUCCION: ASIGNACION puntoycoma
| DECLARACION puntoycoma
| N_PRINT puntoycoma
| N_PRINTLN puntoycoma
| id inc puntoycoma
| id dec puntoycoma
| inc id puntoycoma
| dec id puntoycoma
;

//SWITCH

SWITCH: switch parentesis EXPRESION parentesc llavea CASES_LIST llavec
| switch parentesis EXPRESION parentesc llavea CASES_LIST DEFAULT llavec
| switch parentesis EXPRESION parentesc llavea DEFAULT llavec
| switch parentesis EXPRESION parentesc llavea DEFAULT CASES_LIST llavec
| switch parentesis EXPRESION parentesc llavea CASES_LIST DEFAULT CASES_LIST llavec
;

CASES_LIST: CASES_LIST CASE

| CASE

;

CASE: case EXPRESION dospuntos INSTRUCCIONES

| case EXPRESION dospuntos

;

DEFAULT: default dospuntos INSTRUCCIONES

| default dospuntos

;

//FOR

FOR: for parentesis DECLARACION puntocom EXPRESION puntocom EXPRESION parentesc

BLOQUE_INST

| for parentesis ASIGNACION puntocom EXPRESION puntocom EXPRESION parentesc

BLOQUE_INST

;

//WHILE

WHILE: while parentesis EXPRESION parentesc BLOQUE_INST

;

//DO WHILE

DO_WHILE: do BLOQUE_INST while parentesis EXPRESION parentesc puntocom

;

//METODOS

METODOS: void id parentesis PARAMETROS parentesc BLOQUE_INST

| void id parentesis parentesc BLOQUE_INST

;

//FUNCIONES

FUNCIONES: TIPOVAR id parentesis PARAMETROS parentesc BLOQUE_INST

| TIPOVAR id parentesis parentesc BLOQUE_INST

;

//Parametros

PARAMETROS:PARAMETROS coma TIPOVAR id

| TIPOVAR id

;

//LLAMADA

LLAMADA: call id parentesis PARAMETROSLAMADA parentesc

| call id parentesis parentesc

```

    | id parentesis PARAMETROSLAMADA parentesc
    | id parentesis parentesc
    ;

PARAMETROSLAMADA: PARAMETROSLAMADA coma EXPRESION
    | EXPRESION
    ;

N_PRINTLN: println parentesis EXPRESION parentesc
    | println parentesis parentesc
    ;
N_PRINT: print parentesis EXPRESION parentesc
    | print parentesis parentesc
    ;
N_TYPEOF: typeof parentesis EXPRESION parentesc
    ;

//Bloque de Instrucciones
BLOQUE_INST: llavea INSTRUCCIONES llavec
    | llavea llavec
    ;
//TERNARIOS
TERNARIO: parentesis EXPRESION parentesc interrogacion INST_1LINE dospuntos INST_1LINE
    ;

INST_1LINE: ASIGNACION
    | DECLARACION
    | N_PRINT
    | N_PRINTLN
    | LLAMADA
    | id inc
    | id dec
    | inc id
    | dec id
    ;

//VECTORES
VECTOR: TIPOVAR id corchetea corchetec igual new TIPOVAR corchetea EXPRESION corchetec
    | TIPOVAR id corchetea corchetec corchetea corchetec igual new TIPOVAR corchetea EXPRESIO
N corchetec corchetea EXPRESION corchetec
    | TIPOVAR id corchetea corchetec igual CONJVECTOR
    | TIPOVAR id corchetea corchetec corchetea corchetec igual CONJVECTOR
    | TIPOVAR id corchetea corchetec igual TO_CHAR_ARRAY
    ;

CONJVECTOR: corchetea CONJVECTOR corchetec
    | CONJVECTOR coma corchetea CONJEXP corchetec
    | corchetea CONJEXP corchetec {$$= [$2];}

```

;

CONJEXP: CONJEXP coma EXPRESION

| EXPRESION

;

MODIVECTOR: id corchetea EXPRESION corchetec igual EXPRESION

| id corchetea EXPRESION corchetec corchetea EXPRESION corchetec igual EXPRESION

;

TO_CHAR_ARRAY: toCharArray parentesis EXPRESION parentesc

;

PUSH_V: id punto push parentesis EXPRESION parentesc

;

POP_V: id punto pop parentesis parentesc

;

SPLICE: id punto splice parentesis EXPRESION coma EXPRESION parentesc

;

EXPRESION:

 menos EXPRESION %prec UMINUS

 | EXPRESION mas EXPRESION

 | EXPRESION menos EXPRESION

 | EXPRESION div EXPRESION

 | EXPRESION multi EXPRESION

 | EXPRESION mod EXPRESION

 | EXPRESION pow EXPRESION

 | EXPRESION igualigual EXPRESION

 | EXPRESION diferente EXPRESION

 | EXPRESION menorque EXPRESION

 | EXPRESION menorigual EXPRESION

 | EXPRESION mayorque EXPRESION

 | EXPRESION mayorigual EXPRESION

 | EXPRESION or EXPRESION

 | EXPRESION and EXPRESION

 | EXPRESION xor EXPRESION

 | not EXPRESION

 | id inc

 | inc id

 | id dec

 | dec id

 | parentesis EXPRESION parentesc

 | TIPODATO

 | N_TYPEOF

 | id igual EXPRESION

| LLAMADA
| parentesis EXPRESSION parentesc interrogacion EXPRESSION dospuntos EXPRESSION
| tolower parentesis EXPRESSION parentesc
| toupper parentesis EXPRESSION parentesc
| round parentesis EXPRESSION parentesc
| id corchetea EXPRESSION corchetec
| id corchetea EXPRESSION corchetec corchetea EXPRESSION corchetec
| length parentesis EXPRESSION parentesc
| id punto indexof parentesis EXPRESSION parentesc
| id punto push parentesis EXPRESSION parentesc
;