

UNIVERSIDAD DE SAN CARLOS DE GUATEMALA  
FACULTAD DE INGENIERIA  
ORGANIZACIÓN DE LENGUAJES Y COMPILADORES 1

ARCHIVO DE GRAMATICAS

ERICK ABDUL CHACON BARILLAS

201807169

SECCION: C

TUTOR: MYNOR OCTAVIO PILO TUY

### Expresiones Regulares:

Espacios en blanco	<code>[ \r\t]+</code>
Salto de línea	<code>\n</code>
Comentario simple	<code>(\s\S)*</code>
Comentario multilinea	<code>[/][^]*[*]+([/][^]*[*]+)*[/]</code>
Identificador	<code>[a-zA-Z][a-zA-Z0-9_]*</code>
Decimal	<code>[0-9]+(".[0-9]+)\b</code>
Entero	<code>[0-9]+\b</code>
Caracter	<code>\'((\\\' )[^\n\'])*\'</code>

### Precedencia de operaciones:

'OR'

'AND'

'FCAST'

'IGUALACION' 'DIFERENCIACION'

'MENOR' 'MAYOR' 'MAYORIGUAL' 'MENORIGUAL'

'MAS' 'MENOS'

'POR' 'DIVISION' 'MODULO'

'POTENCIA'

'UMENOS' 'NOT'

### Terminales:

"==" 'IGUALACION';

"," 'PUNTO\_COMA';

"(" 'PARIZQ';

")" 'PARDER';

"." 'PUNTO';

":" 'DOS\_PUNTOS';

"," 'COMA';

"[" 'CORIZQ';

"]" 'CORDER';

"{"	'LLAVEIZQ';
"}"	'LLAVEDER';
"?"	'INTERROGACION';
"="	'IGUAL';
"print"	'RPRINT';
"true"	'TRUE';
"false"	'FALSE';
"new"	'NEW';
"list"	'LIST';
"add"	'ADD';
"tolower"	'TOLOWER';
"toupper"	'TOUPPER';
"length"	'LENGTH';
"if"	'IF';
"else"	'ELSE';
"switch"	'SWITCH';
"case"	'CASE';
"break"	'BREAK';
"default"	'DEFAULT';
"while"	'WHILE';
"for"	'FOR';
"do"	'DO';
"void"	'VOID';
"continue"	'CONTINUE';
"return"	'RETURN';
"truncate"	'TRUNCATE';
"round"	'ROUND';
"typeof"	'TYPEOF';

"tostring"	'TOSTRING';
"tochararray"	'TOCHARARRAY';
"main"	'MAIN';
"+"	'MAS';
"-"	'MENOS';
"*"	'POR';
"/"	'DIVISION';
"^"	'POTENCIA';
"%"	'MODULO';
"!="	'DIFERENCIACION';
"<"	'MENOR';
"<="	'MENORIGUAL';
">"	'MAYOR';
">="	'MAYORIGUAL';
"&&"	'AND';
"  "	'OR';
"!"	'NOT';
"int"	'RENTERO';
"string"	'RSTRING';
"char"	'RCHAR';
"boolean"	'RBOOLEAN';
"double"	'RDOUBLE';

Producciones:

INICIO : INSTRUCCIONES EOF

;

INSTRUCCIONES : INSTRUCCIONES INSTRUCCION

| INSTRUCCION

;

INSTRUCCION : DEFPRINT

| DECLARACION

| DECLARARARRAY

| DECLARARLISTAS

| ASIGNACION

| ASIGNACIONARRAY

| ASIGNACIONLISTA

| MODIFICARLISTAS

| SENTENCIAIF

| SENTENCIASWITCH

| CICLOWHILE

| SENTENCIAFOR

| SENTENCIADOWHILE

| SENTENCIAFUNCION

| LLAMADAFUNCION

| RETORNAR PUNTO\_COMA

| BREAK PUNTO\_COMA

| CONTINUE PUNTO\_COMA

| MAIN LLAMADAFUNCION  
| error PUNTO\_COMA  
;

Gramática para función print:

DEFPRINT : RPRINT PARIZQ LISTAEXPRESION PARDER  
PUNTO\_COMA  
;

LISTAEXPRESION : LISTAEXPRESION COMA EXPRESION  
| EXPRESION  
;

Gramática para declarar variables:

DECLARACION : TIPOS DECLARAVARIOS PUNTO\_COMA  
| TIPOS DECLARAVARIOS IGUAL EXPRESION  
PUNTO\_COMA  
| TIPOS DECLARAVARIOS IGUAL TERNARIO PUNTO\_COMA  
| TIPOS DECLARAVARIOS IGUAL CASTEO PUNTO\_COMA  
| TIPOS DECLARAVARIOS IGUAL LLAMADAFUNCION  
;

DECLARARARRAY : TIPOS CORIZQ CORDER DECLARAVARIOS  
IGUAL NEW TIPOS CORIZQ EXPRESION CORDER PUNTO\_COMA  
| TIPOS CORIZQ CORDER DECLARAVARIOS IGUAL  
LLAVEIZQ LISTAVALORES LLAVEDER PUNTO\_COMA  
;

DECLARARLISTAS : LIST MENOR TIPOS MAYOR DECLARAVARIOS  
IGUAL NEW LIST MENOR TIPOS MAYOR PUNTO\_COMA

| LIST MENOR TIPOS MAYOR DECLARAVARIOS IGUAL  
TOCHARARRAY PARIZQ EXPRESION PARDER PUNTO\_COMA

;

DECLARAVARIOS : DECLARAVARIOS COMA ID

| ID

;

LISTAVALORES : LISTAVALORES COMA EXPRESION

| EXPRESION

;

Gramática para asignar variables:

ASIGNACION : ID IGUAL EXPRESION PUNTO\_COMA

| ID IGUAL TERNARIO PUNTO\_COMA

| ID IGUAL CASTEO PUNTO\_COMA

| ID IGUAL LLAMADAFUNCION

| EXPRESION PUNTO\_COMA

;

ASIGNACION2 : ID IGUAL EXPRESION

| ID IGUAL TERNARIO

| ID IGUAL CASTEO

;

ASIGNACIONARRAY : ID CORIZQ EXPRESION CORDER IGUAL  
EXPRESION PUNTO\_COMA

;

ASIGNACIONLISTA : ID PUNTO ADD PARIZQ EXPRESION PARDER  
PUNTO\_COMA

| ID IGUAL TOCHARARRAY PARIZQ EXPRESION PARDER  
PUNTO\_COMA

;

MODIFICARLISTAS : ID CORIZQ CORIZQ EXPRESION CORDER  
CORDER IGUAL EXPRESION PUNTO\_COMA

;

Gramática para casteo de valores:

CASTEO : PARIZQ TIPOCASTEO PARDER EXPRESION

;

TIPOCASTEO : RENTERO

| RDOUBLE

| RCHAR

;

TIPOS : RENTERO

| RDOUBLE

| RSTRING

| RCHAR

| RBOOLEAN



;

Gramática para operador ternario:

TERNARIO : EXPRESION INTERROGACION EXPRESION  
DOS\_PUNTOS EXPRESION

;

Gramática para sentencia if:

SENTENCIAIF : IF PARIZQ EXPRESION PARDER STATEMENT  
SENTENCIAELSE

;

SENTENCIAELSE : ELSE STATEMENT

| ELSE SENTENCIAIF

|

;

STATEMENT : LLAVEIZQ INSTRUCCIONES LLAVEDER

| LLAVEIZQ LLAVEDER

| error LLAVEDER

;

Gramática para la sentencia switch:

SENTENCIASWITCH : SWITCH PARIZQ EXPRESION PARDER  
LLAVEIZQ LISTACASOS LLAVEDER

| error LLAVEDER

;

LISTACASOS : CASE EXPRESION DOS\_PUNTOS  
CASESTATEMENT

```
| LISTACASOS CASE EXPRESION DOS_PUNTOS
CASESTATEMENT
| DEFAULT DOS_PUNTOS CASESTATEMENT
| LISTACASOS DEFAULT DOS_PUNTOS CASESTATEMENT
;
```

CASESTATEMENT : INSTRUCCIONES

```
|
;
```

Gramática para la sentencia while:

SENTENCIADOWHILE : DO STATEMENT WHILE PARIZQ  
EXPRESION PARDER PUNTO\_COMA

;

Gramática para la sentencia for:

SENTENCIAFOR : FOR PARIZQ DECLARACION EXPRESION  
PUNTO\_COMA EXPRESION PARDER STATEMENT

```
| FOR PARIZQ ASIGNACION EXPRESION PUNTO_COMA
EXPRESION PARDER STATEMENT
```

```
| FOR PARIZQ DECLARACION EXPRESION PUNTO_COMA
ASIGNACION2 PARDER STATEMENT
```

```
| FOR PARIZQ ASIGNACION EXPRESION PUNTO_COMA
ASIGNACION2 PARDER STATEMENT
```

;

Gramática para la sentencia do while:

SENTENCIADOWHILE : DO STATEMENT WHILE PARIZQ  
EXPRESION PARDER PUNTO\_COMA

;

Gramática para la sentencia función:

SENTENCIAFUNCION : TIPOS ID PARIZQ PARAMETROS PARDER  
STATEMENT

| TIPOS ID PARIZQ PARDER STATEMENT

| VOID ID PARIZQ PARAMETROS PARDER STATEMENT

| VOID ID PARIZQ PARDER STATEMENT

;

PARAMETROS : PARAMETROS COMA TIPOS ID

| TIPOS ID

;

Gramática para la sentencia llamada de funciones:

LLAMADAFUNCION : ID PARIZQ PARDER PUNTO\_COMA

| ID PARIZQ LISTAPARAMETROS PARDER PUNTO\_COMA

;

LISTAPARAMETROS : LISTAPARAMETROS COMA EXPRESION

| EXPRESION

;

Gramática para la sentencia retornar:

RETORNAR : RETURN

| RETURN EXPRESION

;

Gramática para expresión:

EXPRESION : EXPRESION MAS EXPRESION

| EXPRESION MENOS EXPRESION

| EXPRESION POR EXPRESION

| EXPRESION DIVISION EXPRESION  
| EXPRESION POTENCIA EXPRESION  
| EXPRESION MODULO EXPRESION  
| MENOS EXPRESION %prec U MENOS

| EXPRESION IGUALACION EXPRESION  
| EXPRESION DIFERENCIACION EXPRESION  
| EXPRESION MENOR EXPRESION  
| EXPRESION MAYOR EXPRESION  
| EXPRESION MENOR IGUAL EXPRESION  
| EXPRESION MAYOR IGUAL EXPRESION

| EXPRESION AND EXPRESION  
| EXPRESION OR EXPRESION  
| NOT EXPRESION %prec NOT

| EXPRESION MAS MAS  
| EXPRESION MENOS MENOS

| PARIZQ EXPRESION PARDER  
| ENTERO  
| DECIMAL  
| CADENA  
| CARACTER  
| TRUE

| FALSE

| ACCEDERVAR

| ACCEDERARRAY

| ACCEDERLISTA

| TOLOWER PARIZQ EXPRESION PARDER

| TOUPPER PARIZQ EXPRESION PARDER

| LENGTH PARIZQ EXPRESION PARDER

| TRUNCATE PARIZQ EXPRESION PARDER

| ROUND PARIZQ EXPRESION PARDER

| TYPEOF PARIZQ EXPRESION PARDER

| TOSTRING PARIZQ EXPRESION PARDER

;

Gramática para accesos:

ACCEDERVAR : ID

;

ACCEDERARRAY : ID CORIZQ EXPRESION CORDER

;

ACCEDERLISTA : ID CORIZQ CORIZQ EXPRESION CORDER  
CORDER

;