

UNIVERSIDAD SAN CARLOS DE GUATEMALA

FACULTAD DE INGENIERIA

ORGANIZACIÓN DE LENGUAJES Y COMPILADORES

NOMBRE: Damaris Julizza Muralles Véliz

CARNET: 202100953

PROYECTO 2

EXPRESIONES REGULARES:

CADENA = [\"(\\.| [^\"\\])*]
CHARACTER = [\"[^\"]\"]
IDENTIFICADOR = [([a-zA-Z])[a-zA-Z0-9_]*]
BOOLEANO = [true | false]
ENTERO = [[0-9]+\b]
DECIMAL = [[0-9]+\.[0-9]+\b]

TERMINALES:

T_INT,T_DOUBLE,T_BOOLEAN,T_CHAR,T_STRING,T_SWITCH,T_CASE,T_DEFAULT,T_IF,T_ELSE,T_DO,
T_WHILE,T_FOR,T_LIST,T_ADD,T_NEW,T_BREAK,T_CONTINUE,T_RETURN,T_PRINT,T_VOID,T_LENGTH,
T_TOCHARARRAY,T_TOLOWER,T_TOSTRING,T_Toupper,T_TRUNCATE,T_TYPEOF,T_ROUND,T_
MAIN,MENIGQUE,MAYIGQUE,DOBLEIG,NOIG,MENQUE,MAYQUE,DOSPTS,PTCOMA,LLAVIZQ,L
LAYDER,PARIZQ,PARDER,IGUAL,COMA,INTERROGACION,CORCER,CORCIZQ,CORCER,N
OT,AND,OR,AUMENTO,DECREMENTO,MAS,MENOS,POR,DIVIDIDO,POTENCIA,MODULO,CAD
ENA,CHARACTER,IDENTIFICADOR,BOOLEANO,ENTERO,DECIMAL,PTS,

NO TERMINALES:

ini,instrucciones,instruccion,funcion_main,declaracion,asignacion,llamada_funcio
n,expresionllamada,funcion,metodo,parametrosllamada,bloqueinstruccionesmet
odo,bloqueinstrucciones,instruccionesmetodo,instruccionesfuncion,instruccionmet
odo,instruccionfuncion,expresion,manejocarateres,casteo,funcionesnativas,tipo
casteonumber,tipo,casteo,expresion_relacional,expresion_logica,expresionescompu
estas,tipos,sentenciaprint,sentenciacontrolif,sentenciacontrolswitch,casos,case
evaluar,ternarioop,retornos,sentenciabreak,ciclos,incializarfor,actualizar,incremento
Dec, vectores, tipovector, declaracionvector, datosvector, accesovector,
modifivector, declaracionlistas, addlistas, Accesoslistas, modifilistas ,
funcionChararray

INICIO:

ini

PRODUCCIONES:

ini

: instrucciones

;

instrucciones

: instrucciones instruccion

| instruccion

;

instruccion

: declaracion PTCOMA

| asignacion PTCOMA

| funcion_main PTCOMA

| funcion

| metodo

| vectores

| declaracionlistas

| modifilistas

| modifivector

| addlistas

| funcionChararray

;

funcion_main

: T_MAIN llamada_funcion

;

declaracion

: tipos

| T_BOOLEAN IDENTIFICADOR

| tipos IGUAL expresion

| tipos IGUAL ternarioop

| T_BOOLEAN IDENTIFICADOR IGUAL expresion

| T_BOOLEAN IDENTIFICADOR IGUAL expresion_logica

| T_BOOLEAN IDENTIFICADOR IGUAL ternarioop

;

asignacion

```

: IDENTIFICADOR IGUAL expresion
| IDENTIFICADOR IGUAL expresion_logica
| IDENTIFICADOR IGUAL ternarioop
;

llamada_funcion
: IDENTIFICADOR PARIZQ PARDER
| IDENTIFICADOR PARIZQ expresionllamada PARDER
;

expresionllamada
: expresionllamada COMA expresion
| expresion

funcion
: tipos PARIZQ PARDER bloqueinstrucciones
| tipos PARIZQ parametrosllamada PARDER bloqueinstrucciones
| T_BOOLEAN IDENTIFICADOR PARIZQ PARDER bloqueinstrucciones
| T_BOOLEAN IDENTIFICADOR IDENTIFICADOR PARIZQ parametrosllamada PARDER
bloqueinstrucciones
;

metodo
:T_VOID IDENTIFICADOR PARIZQ PARDER bloqueinstruccionesmetodo
| T_VOID IDENTIFICADOR PARIZQ parametrosllamada PARDER
bloqueinstruccionesmetodo
;

parametrosllamada
: parametrosllamada COMA tipos
| parametrosllamada COMA T_BOOLEAN IDENTIFICADOR
| tipos
| T_BOOLEAN IDENTIFICADOR
;

bloqueinstruccionesmetodo
: LLAVIZQ instruccionesmetodo LLAVDER
| LLAVIZQ LLAVDER
;

bloqueinstrucciones
: LLAVIZQ instruccionesfuncion LLAVDER
| LLAVIZQ LLAVDER
;

instruccionesmetodo
: instruccionesmetodo instruccionmetodo
| instruccionmetodo
;

instruccionesfuncion
: instruccionesfuncion instruccionfuncion
| instruccionfuncion

```

;

instruccionmetodo

- : declaracion PTCOMA
- | asignacion PTCOMA
- | llamada_funcion PTCOMA
- | sentenciaprint PTCOMA
- | sentenciacontrolF
- | sentenciacontrolswitch
- | sentenciabreak
- | T_CONTINUE PTCOMA
- | incrementoDec PTCOMA
- | ciclos
- | vectores
- | declaracionlistas
- | modifilistas
- | modifivector
- | addlistas
- | funcionChararray

;

instruccionfuncion

- : declaracion PTCOMA
- | asignacion PTCOMA
- | llamada_funcion PTCOMA
- | sentenciaprint PTCOMA
- | sentenciacontrolF
- | sentenciacontrolswitch
- | sentenciabreak
- | retornos
- | T_CONTINUE PTCOMA
- | incrementoDec
- | ciclos
- | vectores
- | declaracionlistas
- | modifilistas
- | modifivector
- | addlistas
- | funcionChararray

;

expresion

- : CADENA
- | ENTERO
- | DECIMAL
- | BOOLEANO
- | CARACTER
- | IDENTIFICADOR
- | MENOS expresion

```

    | expresion POR expresion
    | expresion DIVIDIDO expresion
    | expresion POTENCIA expresion
    | expresion MODULO expresion
    | expresion MAS expresion
    | expresion MENOS expresion
    | PARIZQ expresion PARDER
    | llamada_funcion
    | incrementoDec
    | manejarcarateres
    | casteo
    | funcionesnativas
    | accesolistas
    | accesovector
;

manejarcarateres
: T_TOLOWER PARIZQ expresion PARDER
  | T_Toupper PARIZQ expresion PARDER
;

casteo
: PARIZQ tipocasteo PARDER expresion
  | PARIZQ tipocasteonumber PARDER expresion
;

funcionesnativas
: T_LENGTH PARIZQ expresion PARDER
  | T_TRUNCATE PARIZQ expresion PARDER
  | T_TOCHARARRAY PARIZQ expresion PARDER
  | T_TOSTRING PARIZQ expresion PARDER
  | T_TYPEOF PARIZQ expresion PARDER
  | T_ROUND PARIZQ expresion PARDER
;

funcionChararray
: | T_LIST MENQUE T_CHAR MAYQUE IDENTIFICADOR IGUAL T_TOCHARARRAY PARIZQ
  expresion PARDER
;

tipocasteonumber
: T_INT
  | T_DOUBLE
;

tipocasteo
: T_CHAR
  | T_STRING
;

expresion_relacional
: expresion MAYQUE expresion
  | expresion MENQUE expresion
  | expresion MAYIGQUE expresion

```

```

    | expresion MENIGQUE expresion
    | expresion DOBLEIG expresion
    | expresion NOIG expresion
    | PARIZQ expresion_relacional PARDER
;

expresion_logica
: expresionescompuestas AND expresionescompuestas
  | expresionescompuestas OR expresionescompuestas
  | NOT expresionescompuestas
  | expresion_relacional
;

expresionescompuestas
:expresion_logica
  | expresion
;

tipos
: T_INT IDENTIFICADOR
  | T_DOUBLE IDENTIFICADOR
  | T_CHAR IDENTIFICADOR
  | T_STRING IDENTIFICADOR
;

sentenciaprint
:T_PRINT PARIZQ expresionescompuestas PARDER
;

sentenciacontrolIF
: T_IF PARIZQ expresion_logica PARDER bloqueinstrucciones
  | T_IF PARIZQ expresion_logica PARDER bloqueinstrucciones T_ELSE
bloqueinstrucciones
  | T_IF PARIZQ expresion_logica PARDER bloqueinstrucciones T_ELSE
sentenciacontrolIF
;

sentenciacontrolswitch
:T_SWITCH PARIZQ expresion PARDER LLAVIZQ casos LLAVDER
;

casos
: casos caso_evaluar
  | caso_evaluar
;

caso_evaluar
: T_CASE expresion DOSPTS instruccionesfuncion
  | T_DEFAULT DOSPTS instruccionesfuncion
;

```

```

ternarioop
    :expresion_logica INTERROGACION expresion DOSPTS expresion
;

retornos
    :T_RETURN expresionescompuestas PTCOMA
    | T_RETURN ternarioop PTCOMA
;

sentenciabreak
    : T_BREAK PTCOMA
;

ciclos
    :T_WHILE PARIZQ expresion_logica PARDER bloqueinstrucciones
    | T_DO bloqueinstrucciones T_WHILE PARIZQ expresion_logica PARDER PTCOMA
    | T_FOR PARIZQ inicializarfor PTCOMA expresion_relacional PTCOMA actualizar
    PARDER bloqueinstrucciones
;

inicializarfor
    :IDENTIFICADOR IGUAL expresion
    | tipos IGUAL expresion
;

actualizar
    :IDENTIFICADOR IGUAL expresion
    | incrementoDec
;

incrementoDec
    : IDENTIFICADOR AUMENTO
    | IDENTIFICADOR DECREMENTO
;

vectores
    : tipovector CORCIZQ CORCDER IDENTIFICADOR IGUAL declaracionvector
    PTCOMA
;

tipovector
    : T_INT
    | T_DOUBLE
    | T_CHAR
    | T_STRING
    | T_BOOLEAN
;

declaracionvector
    : T_NEW tipovector CORCIZQ ENTERO CORCDER
    | LLAVIZQ datosvector LLAVDER
;

datosvector

```

```

        : datosvector COMA expresion
        | expresion
;
accesovector
    : IDENTIFICADOR CORCIZQ expresionescompuestas CORCDER
;
modifivector
    : IDENTIFICADOR CORCIZQ expresionescompuestasCORCDER IGUAL expresion
PTCOMA
;
declaracionlistas
    :T_LIST MENQUE tipovector MAYQUE IDENTIFICADOR IGUAL T_NEW T_LISTMENQUE
tipovector MAYQUE PTCOMA
;
addlistas
    : IDENTIFICADOR PTS T_ADD PARIZQ expresion PARDER PTCOMA
;
accesolistas
    : IDENTIFICADOR CORCIZQ CORCIZQ expresionescompuestasCORCDER CORCDER
;
modifilistas
    : IDENTIFICADOR CORCIZQ CORCIZQ CORCIZQ expresionescompuestas CORCDER
CORCDER CORCDER IGUAL expresion PTCOMA
;

```