

Gramática proyecto 2

ACCIONES

:

ACCIONES

ACCION

| ACCION

;

ACCION:

CICLO_IF

| DECLARACION

| CASTEOS

| INCREMENTO_DECREMENTO

| VECTOR_TIPO1

| VECTOR_TIPO2

| SWITCH_CASE

| DECLARACION_INCREMENTOS

| CICLO_FOR

| CICLO_WHILE

| SENTENCIA_TRANSFERENCIA

| FUNCIONES

| METODO

| FUNCION_PRINT

| FUNCION_LOWER
| FUNCION_UPPER
| FUNCION_ROUND
| FUNCION_LENGTH
| FUNCION_TYPEOF
| FUNCION_TOSTRING
| TOCHAR_ARRAY
| FUNCION_PUSH
| FUNCION_POP
| FUNCION_RUN
| PARAMETROS_LLAMADA
;

LISTA_IDENTIFICADOR:

LISTA_IDENTIFICADOR coma Identificador

| Identificador

;

DECLARACION:

Int LISTA_IDENTIFICADOR puntoComa

| Int LISTA_IDENTIFICADOR igual ARITMETICA
puntoComa

| LISTA_IDENTIFICADOR igual ARITMETICA
puntoComa

| Boolean LISTA_IDENTIFICADOR igual
ARITMETICA puntoComa

| Boolean LISTA_IDENTIFICADOR puntoComa

| Char LISTA_IDENTIFICADOR igual ARITMETICA
puntoComa

| Char LISTA_IDENTIFICADOR puntoComa

| String LISTA_IDENTIFICADOR igual
ARITMETICA puntoComa

| String LISTA_IDENTIFICADOR puntoComa

| Double LISTA_IDENTIFICADOR igual
ARITMETICA puntoComa

| Double LISTA_IDENTIFICADOR puntoComa

| ACCESO_VECTORES igual ARITMETICA
puntoComa

| LISTA_IDENTIFICADOR parentA ARITMETICA
parentC

| LLAMAR_FUNCIONES

;

ARITMETICA:

ARITMETICA suma ARITMETICA

// PARA LAS OPERACIONES ARITMETICAS

| Numero

| ARITMETICA resta ARITMETICA

| ARITMETICA division ARITMETICA

| ARITMETICA multiplicacion ARITMETICA

//PARA LOS TIPOS DE DATOS

| verdadero

| falso

| Cadena

| Caracter

| LISTA_IDENTIFICADOR

// incrementos

| LISTA_IDENTIFICADOR incremento

| LISTA_IDENTIFICADOR decremento

| Identificador corcheteA ARITMETICA corcheteC

**| Identificador corcheteA ARITMETICA corcheteC
corcheteA ARITMETICA corcheteC**

;

DECLARACION_INCREMENTOS:

LISTA_IDENTIFICADOR incremento puntoComa

| LISTA_IDENTIFICADOR decremento puntoComa

;

CASTEOS :

**Int LISTA_IDENTIFICADOR igual parentA Int
parentC Numero puntoComa**

**| Int LISTA_IDENTIFICADOR igual parentA Int
parentC falso puntoComa**

**| Int LISTA_IDENTIFICADOR igual parentA Int
parentC Caracter puntoComa**

**| Int LISTA_IDENTIFICADOR igual parentA Int
parentC Cadena puntoComa**

**| Double LISTA_IDENTIFICADOR igual parentA
Double parentC Numero puntoComa**

**| Double LISTA_IDENTIFICADOR igual parentA
Double parentC Cadena puntoComa**

**| Boolean LISTA_IDENTIFICADOR igual parentA
Boolean parentC Numero puntoComa**

**| Boolean LISTA_IDENTIFICADOR igual parentA
Boolean parentC Cadena puntoComa**

**| Boolean LISTA_IDENTIFICADOR igual parentA
Boolean parentC Caracter puntoComa**

**| Char LISTA_IDENTIFICADOR igual parentA Char
parentC Numero puntoComa**

**| Char LISTA_IDENTIFICADOR igual parentA Char
parentC falso puntoComa**

**| Char LISTA_IDENTIFICADOR igual parentA Char
parentC verdadero puntoComa**

**| Char LISTA_IDENTIFICADOR igual parentA Char
parentC Cadena puntoComa**

**| String LISTA_IDENTIFICADOR igual parentA
String parentC Numero puntoComa**

**| String LISTA_IDENTIFICADOR igual parentA
String parentC falso puntoComa**

**| String LISTA_IDENTIFICADOR igual parentA
String parentC verdadero puntoComa**

**| String LISTA_IDENTIFICADOR igual parentA
String parentC Caracter puntoComa**

**| String LISTA_IDENTIFICADOR igual parentA
String parentC Cadena puntoComa**

| parentA Int parentC

;

VECTOR_TIPO1:

**Int corcheteA corcheteC ARITMETICA igual New
Int corcheteA ARITMETICA corcheteC
puntoComa**

**| Double corcheteA corcheteC ARITMETICA igual
New Double corcheteA ARITMETICA corcheteC
puntoComa**

**| Boolean corcheteA corcheteC ARITMETICA
igual New Boolean corcheteA ARITMETICA
corcheteC puntoComa**

**| Char corcheteA corcheteC ARITMETICA igual
New Char corcheteA ARITMETICA corcheteC
puntoComa**

**| String corcheteA corcheteC ARITMETICA igual
New String corcheteA ARITMETICA corcheteC
puntoComa**

// vectores de 2x2

**|Int corcheteA corcheteC corcheteA corcheteC
ARITMETICA igual**

**New Int corcheteA Numero corcheteC corcheteA
Numero corcheteC puntoComa**

**|Char corcheteA corcheteC corcheteA corcheteC
ARITMETICA igual**

**New Char corcheteA CASTEOS Cadena
corcheteC corcheteA Numero corcheteC
puntoComa**

**|Double corcheteA corcheteC corcheteA
corcheteC ARITMETICA igual**

**New Double corcheteA Numero corcheteC
corcheteA Numero corcheteC puntoComa**

**|String corcheteA corcheteC corcheteA
corcheteC ARITMETICA igual**

**New String corcheteA Numero corcheteC
corcheteA Numero corcheteC puntoComa**

**|Boolean corcheteA corcheteC corcheteA
corcheteC ARITMETICA igual**

**New Boolean corcheteA Numero corcheteC
corcheteA Numero corcheteC puntoComa**

;

VECTOR_TIPO2:

// vector tipo 2 de 1 dimension

**Int corcheteA corcheteC ARITMETICA igual
llaveA DATOS_VECTOR llaveC puntoComa**

| String corcheteA corcheteC ARITMETICA igual
llaveA DATOS_VECTOR llaveC puntoComa

| Double corcheteA corcheteC ARITMETICA igual
llaveA DATOS_VECTOR llaveC puntoComa

| Boolean corcheteA corcheteC ARITMETICA
igual DATOS_VECTOR

| Char corcheteA corcheteC ARITMETICA igual
llaveA DATOS_VECTOR llaveC puntoComa

// vector tipo 2 de 2 dimensiones

// ver si colocar los boolean, char y string

| Int corcheteA corcheteC corcheteA corcheteC
ARITMETICA igual

llaveA llaveA DATOS_VECTOR llaveC coma
llaveA DATOS_VECTOR llaveC llaveC puntoComa

| String corcheteA corcheteC corcheteA
corcheteC ARITMETICA igual

llaveA llaveA DATOS_VECTOR llaveC coma
llaveA DATOS_VECTOR llaveC llaveC puntoComa

;

DATOS_VECTOR:

DATOS_VECTOR coma Numero

| Numero

| coma Cadena

| Cadena

| coma Caracter

| Caracter

| llaveA verdadero llaveC puntoComa

| llaveA verdadero coma verdadero llaveC
puntoComa

| llaveA falso llaveC puntoComa

| llaveA falso coma falso llaveC puntoComa

;

ACCESO Vectores:

LISTA_IDENTIFICADOR corcheteA ARITMETICA
corcheteC

| LISTA_IDENTIFICADOR corcheteA ARITMETICA
corcheteC corcheteA ARITMETICA corcheteC

;

CONDICIONAL:

ARITMETICA mayor ARITMETICA

| ARITMETICA menor ARITMETICA

| ARITMETICA igual ARITMETICA

| ARITMETICA menorIgual ARITMETICA
| ARITMETICA mayorIgual ARITMETICA
| ARITMETICA igualIgual ARITMETICA
| ARITMETICA diferente ARITMETICA
| ARITMETICA operadorAnd ARITMETICA
| ARITMETICA operadorNot ARITMETICA
| ARITMETICA operadorOr ARITMETICA

;

CICLO_IF:

IF

| **IF ELSE**

| **IF VARIOS_ELIF ELSE**

| **IF VARIOS_ELIF**

;

**IF: if parentA CONDICIONAL parentC llaveA
ACCIONES llaveC**

;

ELSE: else llaveA ACCIONES llaveC

;

ELIF:

**elif parentA CONDICIONAL parentC llaveA
ACCIONES llaveC**

;

VARIOS_ELIF:

VARIOS_ELIF ELIF

| ELIF

;

SWITCH_CASE:

**switch parentA LISTA_IDENTIFICADOR parentC
llaveA CASES llaveC**

;

CASES:

CASES case ARITMETICA dosPuntos ACCIONES

| case ARITMETICA dosPuntos ACCIONES

| CASES default dosPuntos ACCIONES

;

CICLO_WHILE:

**While parentA CONDICIONAL parentC llaveA
ACCIONES llaveC**

;

CICLO_FOR:

**for parentA DECLARACION CONDICIONAL
puntoComa ARITMETICA parentC llaveA
ACCIONES llaveC**

// for con incremento en la asignacion

**| for parentA DECLARACION CONDICIONAL
puntoComa**

**ARITMETICA igual ARITMETICA suma
ARITMETICA**

parentC llaveA ACCIONES llaveC

// fot con decremento en la asignacion

**| for parentA DECLARACION CONDICIONAL
puntoComa**

**ARITMETICA igual ARITMETICA resta
ARITMETICA**

parentC llaveA ACCIONES llaveC

// for con variables en la asignacion

;

CICLO_WHILE:

**do llaveA ACCIONES llaveC While parentA
CONDICIONAL parentC puntoComa**

// Do until

**do llaveA ACCIONES llaveC until parentA
CONDICIONAL parentC puntoComa**

;

SENTENCIA_TRANSFERENCIA:

break puntoComa

| continue puntoComa

| return puntoComa

| return ARITMETICA puntoComa

;

FUNCIONES:

**LISTA_IDENTIFICADOR parentA PARAMETROS
parentC dosPuntos**

TIPO_DATO llaveA ACCIONES llaveC

// FUNCION SIN PARAMETROS

**| LISTA_IDENTIFICADOR parentA parentC
dosPuntos**

TIPO_DATO llaveA ACCIONES llaveC

;

LLAMAR_FUNCIONES:

**LISTA_IDENTIFICADOR parentA parentC
puntoComa**

;

LLAMAR_PARAMETROS:

PARAMETROS coma ARITMETICA

| ARITMETICA

;

PARAMETROS:

PARAMETROS coma TIPO_DATO Identificador
| TIPO_DATO Identificador

;

TIPO_DATO:

Int | String | Boolean | Char | Double
;

METODO:

LISTA_IDENTIFICADOR parentA parentC
dosPuntos Void llaveA ACCIONES llaveC

// metodo sin tipo definido

| LISTA_IDENTIFICADOR parentA parentC
dosPuntos llaveA ACCIONES llaveC

;

FUNCION_PRINT:

print parentA ARITMETICA parentC puntoComa
| println parentA ARITMETICA parentC
puntoComa

;

FUNCION_LOWER:

String LISTA_IDENTIFICADOR igual toLower
parentA ARITMETICA parentC puntoComa

;

FUNCION_UPPER:

String LISTA_IDENTIFICADOR igual toUpper
parentA ARITMETICA parentC puntoComa

;

FUNCION_ROUND:

Double LISTA_IDENTIFICADOR igual round
parentA Numero parentC puntoComa

| Int LISTA_IDENTIFICADOR igual round parentA
Numero parentC puntoComa

;

FUNCION_LENGTH:

String LISTA_IDENTIFICADOR igual length
parentA Identificador parentC puntoComa

| Int LISTA_IDENTIFICADOR igual length
parentA Identificador corcheteA Numero
corcheteC parentC puntoComa

;

FUNCION_TYPEOF:

**String LISTA_IDENTIFICADOR igual typeof
parentA ARITMETICA parentC puntoComa**

**| Int LISTA_IDENTIFICADOR igual typeof parentA
ARITMETICA parentC puntoComa**

**| Boolean LISTA_IDENTIFICADOR igual typeof
parentA ARITMETICA parentC puntoComa**

**| Char LISTA_IDENTIFICADOR igual typeof
parentA ARITMETICA parentC puntoComa**

**| Double LISTA_IDENTIFICADOR igual typeof
parentA ARITMETICA parentC puntoComa**

;

FUNCION_TOSTRING:

**Int LISTA_IDENTIFICADOR igual ToString
parentA ARITMETICA parentC puntoComa**

**| String LISTA_IDENTIFICADOR igual ToString
parentA ARITMETICA parentC puntoComa**

**| Boolean LISTA_IDENTIFICADOR igual ToString
parentA ARITMETICA parentC puntoComa**

;

TOCHAR_ARRAY:

**Char corcheteA corcheteC ARITMETICA igual
toCharArray**

parentA Cadena parentC puntoComa

;

FUNCION_PUSH:

**LISTA_IDENTIFICADOR punto Push parentA
ARITMETICA parentC puntoComa**

;

FUNCION_POP:

**LISTA_IDENTIFICADOR punto Pop parentA
parentC puntoComa**

;

FUNCION_RUN:

Run LLAMAR_FUNCIONES

;