**UNIVERSIDAD DE SAN CARLOS DE GUATEMALA**

**FACULTAD DE INGENIERÍA**

**ESCUELA DE CIENCIAS Y SISTEMAS**

**ORGANIZACIÓN DE LENGUAJES Y COMPILADORES 1**

**SEGUNDO SEMESTRE 2021**

**CATEDRATICO: ING. KEVIN LAJPOP**

**TUTOR ACADEMICO: SANDRA JIMÉNEZ**



**PROYECTO No.2**

**[GRAMATICA]**

Lester Efrain Ajucum Santos 201504510

# GRAMATICA PARA PROYECTO 2 (SYSCOMPILER)

INICIO -> INSTRUCCIONES

Al finalizar el análisis sintáctico con el analizador generado con la herramienta Jison, la producción INICIO es la ultima producción en ser reducida. Esta producción retorna un Ast que guarda una lista de instrucciones.

INSTRUCCIONES 🡪 INSTRUCCIONES INSTRUCTION

| INSTRUCTION

La producción INSTRUCCIONES produce una lista donde cada elemento reconocido en el programa es una instrucción. Esta lista se devuelve para ser guardada en el Ast.

INSTRUCTION 🡪 WRITELINE

| VARIABLE\_DECLARATION ‘;’

| VARIABLE\_ASSIGNMENT ‘;’

| IF\_STATEMENT

| FOR\_STATEMENT

| WHILE\_STATEMENT

| SWITCH\_STATEMENT

| POST\_INCREMENT ‘;’

| POST\_DECREMENT ‘;’

| FUNCTIONS

| FUNC\_CALL ‘;’

| BREAK ‘;’

| CONTINUE ‘;’

| RETURN ‘;’

| RETURN E ‘;’

La producción INSTRUCTION contiene cada una de las producciones que representan una sentencia reconocida por el lenguaje. Cuando alguna de estas producciones se reduce devuelve un elemento instrucción que será guardado en la lista de instrucciones.

## Declaración de variables

Esta producción reconoce una lista de identificadores, a estos identificadores es posible que se les asocie un valor inicial o no.

VARIABLE\_DECLARATION 🡪 DECL\_TYPE ID\_LIST ‘=’ E

| DECL\_TYPE ID\_LIST

ID\_LIST 🡪 ID\_LIST ‘,’ id

| id

DECL\_TYPE 🡪 "int"

| "double"

| "boolean"

| "char"

| "string"

## Instrucción WriteLine

Esta producción reconoce una sentencia de impresión en consola. El valor que se imprime es devuelto por la producción E (Corto para expresión).

WRITELINE 🡪 "WriteLine" ‘(‘ E ‘)’ ‘;’

## Asignación de variable

VARIABLE\_ASSIGNMENT -> id "=" E

## Sentencia de control IF

En esta producción se reconoce la sentencia de control if e if anidado. Esta instrucción devuelve un elemento IF con una condición, una lista de instrucciones o devuelve un elemento If con una lista de instrucciones que contiene sentencias if, generando de esa forma la estructura de un if anidado.

IF\_STATEMENT 🡪 "if" ‘(‘E ‘)’ ‘{‘INSTRUCCIONES ‘}’

| “if" ‘(‘E ‘)’ ‘{‘INSTRUCCIONES ‘}’ “else” ‘{‘INSTRUCCIONES ‘}’

| “if" ‘(‘E ‘)’ ‘{‘INSTRUCCIONES ‘}’ “else” IF\_STATEMENT

## Ciclo For

FOR\_STATEMENT 🡪 "for" ‘(‘FOR\_INIT\_OPT ‘;’ E ‘;’ FOR\_UPDATE ‘)’ ‘{‘ INSTRUCCIONES ‘}’

FOR\_INIT\_OPT 🡪 VARIABLE\_ASSIGNMENT

| VARIABLE\_DECLARATION

FOR\_UPDATE 🡪 POST\_INCREMENT

|POST\_DECREMENT

|VARIABLE\_ASSIGNMENT

## Ciclo While

WHILE\_STATEMENT -> "while" ‘(‘ E ‘)’ ‘{‘ INSTRUCCIONES ‘}’

## Sentencia de control Switch

SWITCH 🡪 "switch" ‘(‘ E ‘)’ ‘{‘ CASE\_LIST ‘}’

| "switch" ‘(‘ E ‘)’ ‘{‘ CASE\_LIST DEFAULT ‘}’

| "switch" ‘(‘ E ‘)’ ‘{‘ DEFAULT ‘}’

CASE\_LIST 🡪 CASE\_LIST CASE

| CASE

CASE 🡪 "case" E ‘:’ INSTRUCCIONES

DEFAULT 🡪 “default” ‘:’ INSTRUCCIONES

## Declaración de Funciones

FUNCTIONS 🡪 DECL\_TYPE id ‘(‘ PARAMS\_LIST ‘)’ ‘{‘ INSTRUCCIONES ‘}’

| DECL\_TYPE id ‘(‘ ‘)’ ‘{‘ INSTRUCCIONES ‘}’

| VOID id ‘(‘ PARAMS\_LIST ‘)’ ‘{‘ INSTRUCCIONES ‘}’

| VOID id ‘(‘ ‘)’ ‘{‘ INSTRUCCIONES ‘}’

PARAMS\_LIST 🡪 PARAMS\_LIST "," DECL\_TYPE id

| DECL\_TYPE id

## Llamada a función

FUNC\_CALL 🡪 id ‘(‘ VALUE\_LIST ‘)’

| id ‘(‘ ‘)’

VALUE\_LIST 🡪 VALUE\_LIST ‘,’ E

|E

## Expresiones de incremento/decremento

POST\_INCREMENT 🡪 id “++”

POST\_DECREMENT 🡪 id “--”

## Expresión ternaria

Esta expresión utiliza tres operandos para simplificar una instrucción IF.

TERNARY 🡪 E ‘?’ E ‘:’ E

## Ciclo Do While

DO\_WHILE\_STATEMENT 🡪 "do" ‘{' INSTRUCCIONES ‘}’ "while" ‘(‘ E ‘)’

## Producción de Expresiones

E 🡪 E "+" E

| E "-" E

| E "\*" E

| E "/" E

| E "\*\*" E

| E "%" E

| E "and" E

| E "or" E

| "!" E

| E "==" E

| E "!=" E

| E "<" E

| E ">" E

| E "<=" E

| E ">=" E

| "(" E ")"

| "-" E

| POST\_INCREMENT

| POST\_DECREMENT

| PRE\_INCREMENT

| PRE\_DECREMENT

| integer

| double

| string

| character

| id

| "true"

| "false"

| FUNC\_CALL

| TERNARY