**Zamudio Pichardo Daniel Balderas Campos José Ángel Calderón Mora Cristian Enrique**

<PROGRAMA> ::= {<DECLARACION\_FUNCIONES>} <FUNCION\_MAIN> {<FUNCION>}

<FUNCION\_MAIN> ::= <TIPO\_DATO> 'main' '(' ')' '{' <CUERPO\_FUNCION> '}'

<DECLARACION\_FUNCIONES> ::= <TIPO\_DATO> <ID> '('<AGUMENTOS\_RECIVIDOS>')' <FIN>

<FUNCION> ::= <TIPO\_DATO> <ID> '('<AGUMENTOS\_RECIVIDOS>')' '{' <CUERPO\_FUNCION> '}'

<CUERPO\_FUNCION> ::= '{' <SENTENCIAS> '}'

<OPERADOR\_ARITMETICO> ::= '+'|'-'|'\*'|'/'

<OPERADOR\_ASIGNACION> ::= '='|'+='|'-='|'/='|'\*='

<OPERADOR\_RELACIONAL> ::= '>'|'=='|'<'|'!='|'>='|'<='

<ID> ::= (<LETRA>|<GION\_BAJO>){<LETRA>|<DIGITO>}

<NUMERO> ::= <DIGITO>{<DIGITO>}

<NUMERO\_FLOAT> ::= <NUMERO>'.'<NUMERO>

<DIGITO> ::= '1'|'2'|'3'|'4'|'5'|'6'|'7'|'8'|'9'

<LETRA\_MINUSCULA> ::= 'a'|'b'|'c'|'d'|'e'|'f'|'g'|'h'|'i'|'j'|'k'|'l'|'m'|'n'|'o'|'p'|'q'|'r'|'s'|'t'|'u' |'v' |'w'|'x'|'y'|'z'

<LETRA\_MAYUSCULA> ::= 'A'|'B'|'C'|'D'|'E'|'F'|'G'|'H'|'I'|'J'|'K'|'L'|'M'|'N'|'O'|'P'|'Q'|'R'|'S'|'T' |'U'|'V'|'W'|'X'|'Y'|'Z'

<LETRA> ::= <LETRA\_MINUSCULA>|<LETRA\_MAYUSCULA>

<CADENA> ::= '”'{<LETRA>|<DIGITO>|'!'|'#'|'$'|'%'|'&'|'/'|'('|')' |'='|'''|'\'|'¿'|'¡'|'+'|'\*'|'{'|'}'|'['|']'|'-'|'\_'|'.'|':'|','|';'}'”'

<TIPO\_DATO> ::= 'int'|'float'

<GION\_BAJO> ::= '\_'

<FIN> ::= ';'

<SENTENCIAS> ::= {<SENTENCIA\_DECLARATORIA>|<SENTENCIA\_EXPRESION>|<SENTENCIA\_CONDICIONAL>|<SENTENCIA\_CICLICA> |<SENTENCIA\_RETORNO>| (<LLAMADA\_FUNCION> <FIN>)|<SENTENCIA\_SALIDA>|<SENTENCIA\_ENTRADA>}

<SENTENCIA\_CONDICIONAL> ::= 'if' '('<EVALUDACION>')''{'<SENTENCIAS>'}' ['else' '{'<SENTENCIAS>'}']

<SENTENCIA\_RETORNO> ::= 'return' (<ID>|<NUMERO>|<NUMERO\_FLOAT>) <FIN>

<SENTENCIA\_EXPRESION> ::= <ID> <OPERADOR\_ASIGNACION> <EXPRESION> <FIN>

<SENTENCIA\_CICLICA> ::= 'while''('<EVALUACION>')' '{'<SENTENCIAS>'}'

<SENTENCIA\_DECLARATORIA> ::= <TIPO\_DATO> <ID> ['=' <EXPRESION>] <FIN>

<SENTENCIA\_SALIDA> ::= 'pritf' '(' <CADENA> ')' <FIN>

<SENTENCIA\_ENTRADA> ::= 'scanf' '(' <ID> ')' <FIN>

<EXPRESION> ::=<ID>|<LLAMADA\_FUNCION>|<NUMERO>|<NUMERO\_FLOAT> [<OPERADOR\_ARITMETICO><EXPRESION>]

<LLAMADA\_FUNCION> ::= <ID> '(' <ARGUMENTOS\_ENVIADOS> ')'

<AGUMENTOS\_ENVIADOS> ::= [(<ID>|<NUMERO>|<NUMERO\_FLOAT>) {',' (<ID>|<NUMERO>|<NUMERO\_FLOAT>)}]

<AGUMENTOS\_RECIVIDOS> ::= [ <TIPO\_DATO> <ID> {',' <TIPO\_DATO> <ID> }]

<EVALUACION> ::= <ID> <OPERADOR\_RELACIONAL> <LLAMADA\_FUNCION>|<NUMERO> |<NUMERO\_FLOAT>

Cada token tiene que estar separado por uno o más espacios ó uno o más tabuladores; ó uno o más salto de línea.

Hay diferenciación entre minúsculas y mayúsculas.

Los elementos terminales están entre comillas simples ‘.

Los corchetes “[]” sirven para representar cosas opcionales.

Los paréntesis “()” sirven para agrupar elementos.

Las llaves “{}” sirven para representar 0 ó muchas veces.

Los elementos no terminales están entre <>

Los ID no se pueden repetir no importa si están en otras funciones.

No se puede usar un ID sin ser declarado, eso incluye funciones, se tiene que declarar la función antes de ser usada.