

Gramática utilizada

No terminales: INI, ENTRADA, ENTCERO, FUNCIONBODY, METODOBODY, EXECBODY, DEC_VAR, DEC_VECT, DEC_LIST, TIPO, TIPO_VECT, TIPO_LIST, LISTAPARAMETROS, PARAMETROS, LISTAVALORES, VALORES, INSTRUCCION, INSCERO, SENTENCIATRANSFERENCIA, SENTENCIACICLO, WHILE, FOR, DOWHILE, IF, IFELSE, ELSEIF, ACTUALIZACION, SWITCH, CASESLIST, DEFAULT, TIPODATO, EXPRESION, CASTEO, TERNARIO, FUNCIONESRESERVADAS, FPRINT, FTOLOWER, FTOUPPER, FLENGTH, FROUND, FTYPEOF, FTOSTRING, LLAMADA.

Terminales: prclase, prdouble, printeger, prboolean, prstring, prchar, prreturn, prprint, prnew, pradd, prif, prelse, prswitch, prfor, prwhile, prcontinue, prbreak, prvoid, princremento, prdecremento, prlower, prupper, prtuncate, prround, prtoCharL, prtoString, prtypeOf, prtrue, prfalse, pror, prand, prsuma, prresta, prmulti, prdiv, menor, menorigual, mayor, mayorigual, igualigual, diferente, exp, modulo, interrogacion, cabre, ccierra, labre, lcierra, id, doble, entero, cadena, EOF.

La gramática se compone de producciones que están colocadas en recursividad por la izquierda. La gramática se implementó con el patrón intérprete a modo de manejar listas de listas en donde se van agregando toda la información necesaria a partir de las mismas producciones.

Las producciones principales del analizador son: INSTRUCCION y EXPRESION.

Instrucción: en esta producción puede venir cualquier sentencia, llamada simple, declaración y asignación dentro de un ámbito en el programa. Pueden venir varias, solamente una o ninguna.

Expresión: en esta producción puede venir cualquier valor que será utilizado posteriormente a su procesamiento para asignar, declarar o utilizar de parámetro para la llamada a una función. Así como una llamada también puede ser una expresión al retornar esta un valor y luego ser utilizada en alguna instrucción.

Los casteos y operadores ternarios pueden ser utilizados como instrucción al momento de declarar o asignar variables, así como también pueden ser tratadas como expresiones al momento de retornar un valor y ser llamada posteriormente en ejecución del programa.

Para las cadenas se hizo uso de estados y que estas puedan contener caracteres especiales precedidas por una barra invertida.

Para poder recuperarse de los errores, se utilizaron producciones de error con algún terminal como recuperación del mismo y luego se añade a una lista de errores que se limpia cada vez que reconoce un EOF.