Gramática

<PROGRAM> -> class (id) <DECLARA> <ESTATUTOS> endclass

<DECLARA> -> declare <AUX1> of <TIPO> ; <AUX2>

<AUX1> -> id <AUX3>

<AUX3> -> <AUX4>

<AUX3> -> £

<AUX4> -> [ cte entera ] <AUX5>

<AUX5> -> <AUX4>

<AUX5> -> £

<AUX2> -> <DECLARA>

<AUX2> -> £

<TIPO> -> int

<TIPO> -> float

<TIPO> -> char

<TIPO> -> string

<ESTATUTOS> -> <AUX6> ;

<ESTATUTOS> -> £

<AUX6> -> <EST\_ASIG>

<AUX6> -> <EST\_IF>

<AUX6> -> <EST\_WHILE>

<AUX6> -> <EST\_DO>

<AUX6> -> <EST\_READ>

<AUX6> -> <EST\_WRITE>

<EST\_ASIG> -> <ASIG> = <EXPR>

<ASIG> -> id <AUX7>

<AUX7> -> <AUX8> \*DIM\_ASIG\*

<AUX7> -> £

<AUX8> -> <EXPR> <AUX9>

<AUX9> -> , <AUX8>

<AUX9> -> £

<EST\_IF> -> if ( <EXPR> ) <ESTATUTOS>

<EST\_IF> -> <AUX10> endif

<AUX10> -> else <ESTATUTOS>

<AUX10> -> £

<EST\_WHILE> -> while ( <EXPR> ) <ESTATUTOS> endwhile

<EST\_DO> -> do <ESTATUTOS> dowhile ( <EXPR> ) endo

<EST\_READ> -> read ( <AUX11> )

<AUX11> -> id <AUX12>

<AUX12> -> , <AUX11>

<AUX12> -> £

<EST\_WRITE> -> write ( <AUX13> )

<AUX13> -> <EXPR> <AUX14>

<AUX14> -> , <AUX13>

<AUX14> -> £

<EXPR> -> <EXPR2> <AUX15>

<AUX15> -> || <EXPR>

<AUX15> -> £

<EXPR2> -> <EXPR3> <AUX16>

<AUX16> -> && <EXPR2>

<AUX16> -> £

<EXPR3> -> <AUX17> <EXPR4>

<AUX17> -> !

<AUX17> ->£

<EXPR4> -> <EXPR5> <AUX18>

<AUX18> -> <OPEREL> <EXPR5>

<AUX18> -> £

<OPEREL> -> ==

<OPEREL> -> !=

<OPEREL> -> <

<OPEREL> -> <=

<OPEREL> -> >

<OPEREL> -> >=

<EXPR5> -> <TERM> <AUX19>

<AUX19> -> <AUX20> <EXPR5>

<AUX19> -> £

<AUX20> -> +

<AUX20> -> -

<TERM> -> <FACT> <AUX21>

<AUX21> -> <AUX22> <TERM>

<AUX21> -> £

<AUX22> -> \*

<AUX22> -> /

<AUX22> -> %

<FACT> -> <ASIG>

<FACT> -> cteentera

<FACT> -> ctefloat

<FACT> -> ctenotacion

<FACT> -> ctecaracter

<FACT> -> ctestring

<FACT> -> ( <EXPR> )