

PROGRAMA -> DECL PROGRAMA | PROCEDIMENT PROGRAMA | MAIN.

DECL -> DECL\_TUPLA ';' | DECL\_TAUULA ';' | DECL\_TIPUS ';' .

MODIF -> MODIF\_TUPLA ';' | REM\_TUPLA ';' | AFG\_TUPLA ';' | MODIF\_TAUULA ';' |  
REDEF\_TAUULA ';' | REDEF\_TUPLA ';' | MODIF\_TIPUS ';' .

SENTENCIES -> DECL | MODIF | MENTRE | IF | SWITCH | FMENTRE .

PROCEDIMENT -> metode ID '(' ARGS ')' '{' SENTENCIES '}' | metode TIPUS ID '(' ARGS ')' '{'  
SENTENCIES RETURN'}' .

RETURN -> return VALOR .

CRIDA\_PROC -> ID '(' ARGS ')' ';' .

MAIN -> metode main '(' cad arguments '[' ']' ')' '{' M\_1 '}' .

M\_1 -> SENTENCIES M\_1 | CRIDA\_PROC M\_1 | .

ARGS -> ARGS\_1 | .

ARGS\_1 -> ARG ',' ARGS\_1 | ARG .

ARG -> TIPUS ID .

MENTRE -> mentre '(' L ')' fer '{' SENTENCIES '}' .

IF -> if '(' L ')' '{' SENTENCIES '}' IF\_1 .

IF\_1 -> else if '(' L ')' '{' SENTENCIES '}' IF\_1 | else '{' SENTENCIES '}' | .

FMENTRE -> fer '(' SENTENCIES ')' mentre '(' L ')' ';' .

SWITCH -> switch '(' VALOR ')' '{' SWITCH\_1 '}' .

SWITCH\_1 -> case VALOR ':' SENTENCIES break ';' SWITCH\_1 | default ':' SENTENCIES  
break ';' | .

DECL\_TIPUS -> TIPUS A | TIPUS A\_1 .

MODIF\_TIPUS -> ID I\_1 .

TIPUS -> ent | logic | decimal | cadena | car .

A -> A\_1 I .

A\_1 -> ID ',' A\_1 | ID .

I -> '=' E | '=' L | '=' MODIF\_TIPUS | '=' MODIF\_TUPLA | '=' MODIF\_TAUULA .

I\_1 -> '++' | '--' | '+=' E | '-=' E | I .

DECL\_TAUULA -> TIPUS ID '[' ']' '=' NOVA\_TAUULA | TIPUS ID '[' ']' .

NOVA\_TAUULA -> nou ID '[' enter ']' '{' VALORS '}' .

VALORS -> VALOR | VALOR ',' VALORS.

VALOR -> E | L.

MODIF\_TAULA -> ID '[' enter ']' I\_1 .

REDEF\_TAULA -> ID '=' NOVA\_TAULA.

DECL\_TUPLA -> tupla ID '(' ')' NOVA\_TUPLA | tupla ID '(' ')' .

NOVA\_TUPLA -> '=' '(' VALORS ')' .

MODIF\_TUPLA -> ID '('enter')' I\_1.

REM\_TUPLA -> ID '.'remove('VALOR')'.

AFG\_TUPLA -> ID '.'append('VALOR')'.

REDEF\_TUPLA -> ID '=' NOVA\_TUPLA.

E -> F OP.

OP -> '+' F OP | '-' F OP | '\*' F OP | '/' F OP | '%' F OP | .

F -> '('E')' | venter | vdecimal | vcharacter | '-' venter | '+' venter | '-' vdecimal | '+' vdecimal | '-' vcharacter | '+' vcharacter | cadena.

L -> EXPRESIO | EXPRESIO COMP\_LOG L.

COMP\_LOG -> or | and .

EXPRESIO -> '!' '('EXPRESIO')' | '('EXPRESIO')' | EXPRESIO\_1.

EXPRESIO\_1 -> VAL\_BOOLEA COMP\_BOOLEA VAL\_BOOLEA | E COMP\_NUM E | E COMP\_BOOLEA E.

VAL\_BOOLEA -> vlogic | '!' vlogic | '!' ID.

COMP\_NUM -> '<=' | '>=' .

COMP\_BOOLEA -> '==' | '!=' .