Especificação GLC

<programa>::= begin <opcional id> <opcional def> start <lista de comandos> end .

<opcional id>::= identificador | ε

<opcional def>::= define <declaração de variaveis> | ε

<tipo>::= num | real | text | flag

<valor>::= constante inteira | constante real | constante literal

<declaração de variáveis>::= <lista de identificadores>: <tipo> <declaração de variaveis’>

<declaração de variáveis’> ::= = <valor> ; | [<constante inteira>] <declaração de variaveis’’’> | ;

<declaração de variaveis’’> ::= = {<valores>}; | ε

<lista de identificadores>::= identificador <lista de identificadores’>

<lista de identificadores>::= , <lista de identificadores> | ε

<valores>::= <valor><valores’>

<valores’>::= , <valores> | ε

<lista de comandos>::= <comando><lista comandos’>

<lista comandos’>::= <lista de comandos> | ε

<comando>::= <comando de atribuição> | <comando entrada dados> | <comando seleção> | <comando repetição>

<expressão>::= <expressão aritmética ou lógica> <expressão’>

<expressão’>::= == <expressão aritmética ou lógica> | != <expressão aritmética ou lógica> | << <expressão aritmética ou lógica> | >> <expressão aritmética ou lógica> | <<= <expressão aritmética ou lógica> | >>= <expressão aritmética ou lógica> | ε

<expressão aritmética ou lógica>::= <termo2> <menor prioridade>

<menor prioridade>::= + <termo2> <menor prioridade> | - <termo2> <menor prioridade> | | <termo2> <menor prioridade> | ε

<termo2>::= <termo1> <media prioridade>

<media prioridade>::= \* <termo1> <media prioridade> | / <termo1> <media prioridade> | % <termo1> <media prioridade> | %% <termo1> <media prioridade> | & <termo1> <media prioridade> | & <termo1> <media prioridade> | ε

<termo1>::= <elemento> <maior prioridade>

<maior prioridade>::= \*\* <elemento> <maior prioridade> | ε

<elemento>::= identificador | constante inteira | constante real | constante literal | true | false | (<expressão>) | ! (<expressão>)

<comando atribuição>::= set identificador <comando atribuição’>

<comando atribuição’>::= = <expressão> ; | [constante inteira] = <expressão> ;

<comando entrada dados>::= read ( identificador <cmd dados>

<cmd dados>::= ); | [constante inteira]);

<comando saída dados>::= show ( <lista de id e/ou const>);

<lista de id e/ou const>::= <valor><lista> | identificador <lista>

<lista>::= , < lista de id e/ou const > | ε

<comando seleção>::= if <expressão booleana> then <lista de comandos> <else> end;

<else>::= else <lista de comandos> | ε

<comando repetição>::= loop while <expressão> <op booleano> <cmd1> | <valor/id> <op booleano><cmd1> | ! <expressão booleana>

<cmd1>::= <expressão> | <valor/id>

<op booleano>::= == | != | << | >> | <<= |>>= | & | “| |”