Gramática Livre de Contexto LL(1) da Linguagem L

S → **DeclararSA** | λ

A → **ComandosA** | λ

Declarar → "final" "id" "<-" **B** const ";" | **C** "id" **ListaIds** ";"

B → "-" | λ

C → "int" | "boolean" | "byte" | "string"

ListaIds → **Atrib D**

Atrib → "<-" **B** "const" | λ

D → "," "id" **Atrib** | λ

Comando → **Atribuicao** | **Repeticao** | **Teste** | **Nulo** | **Leitura** | **Escrita**

Atribuicao → "id" "<-" **Expressao** ";"

Repeticao → "while" "("**Expressao**")" **BlocoWhile**

BlocoWhile → "begin" **E** "endwhile" | **Comando**

E → **ComandoE** | λ

Teste → "if" "("**Expressao**")" **G**

G → **Blocoif** | **Comando BlocoElse**

Blocoif → "begin" **H** "endif" "else" "begin" **H** "endelse"

H → **ComandoH** | **Comando**

BlocoElse → "else" **Comando** | λ

Nulo → ";"

Leitura → "readln""(""id"")"";"

Escrita → "write""("**ListaExpressoes**")"";" | "writeln""("**ListaExpressoes**")"";"

ListaExpressoes → **ExpressaoI**

I → "," **ExpressaoI** | λ

Expressao → **ExpJ**

J → "=" **Exp** | "!=" **Exp** | "<" **Exp** | ">" **Exp** | "<=" **Exp** | ">=" **Exp** | λ

Exp → **KTL**

K → "+" | "-" | λ

L → "+" **TL** | "-" **TL** | "||" **TL** | λ

T → **FM**

M → "*" **F** | "&&" **F** | "/" **F** | λ

F → "!" **F** | "("**Expressao**")" | **B** const | "id"