



UFPEL

LINGUAGENS FORMAIS:

Especificação e Implementação de um Analisador Léxico-Sintático

Rafael Silveira, Pablo Margreff e Letícia Sampaio

{rsilveira,pmargreff,lsampaio}@inf.ufpel.edu.br

1. INTRODUÇÃO

O Analisador léxico é o programa encarregado do processo de analisar a entrada de caracteres e produzir uma sequência de símbolos chamado "símbolos léxicos" (tokens), que podem ser manipulados mais facilmente pelo analisador sintático.

Já o Analisador Sintático (*parsing*) é o programa encarregado do processo de analisar uma sequência de símbolos de entrada (obtidos do léxico) para determinar sua estrutura gramatical segundo uma determinada gramática formal. Essa análise faz parte de um compilador, junto com o analisador léxico e semântico.

2. METODOLOGIA

Primeiramente, foi desenvolvido o autômato finito com uso da ferramenta Jflap, o qual especifica o analisador léxico de acordo com a linguagem solicitada para este trabalho. Logo após, foi escolhida a linguagem de programação Python¹, por conta de sua alta abstração e baixa curva de aprendizagem. Após a escolha da linguagem de programação, foi transcrito todo o autômato finito para código, com auxílio da ferramenta de controle de versão Git.

Ao fim do desenvolvimento da primeira parte do trabalho, iniciou-se os testes com diversos tipos de entradas de dados. A segunda parte do trabalho, foi desenvolver a gramática livre de contexto, que é a principal parte do analisador sintático. Por fim, foram integrados e testados os códigos das duas partes do trabalho e criada uma interface gráfica, ilustrado na Figura 1, Figura 2 e Figura 3.

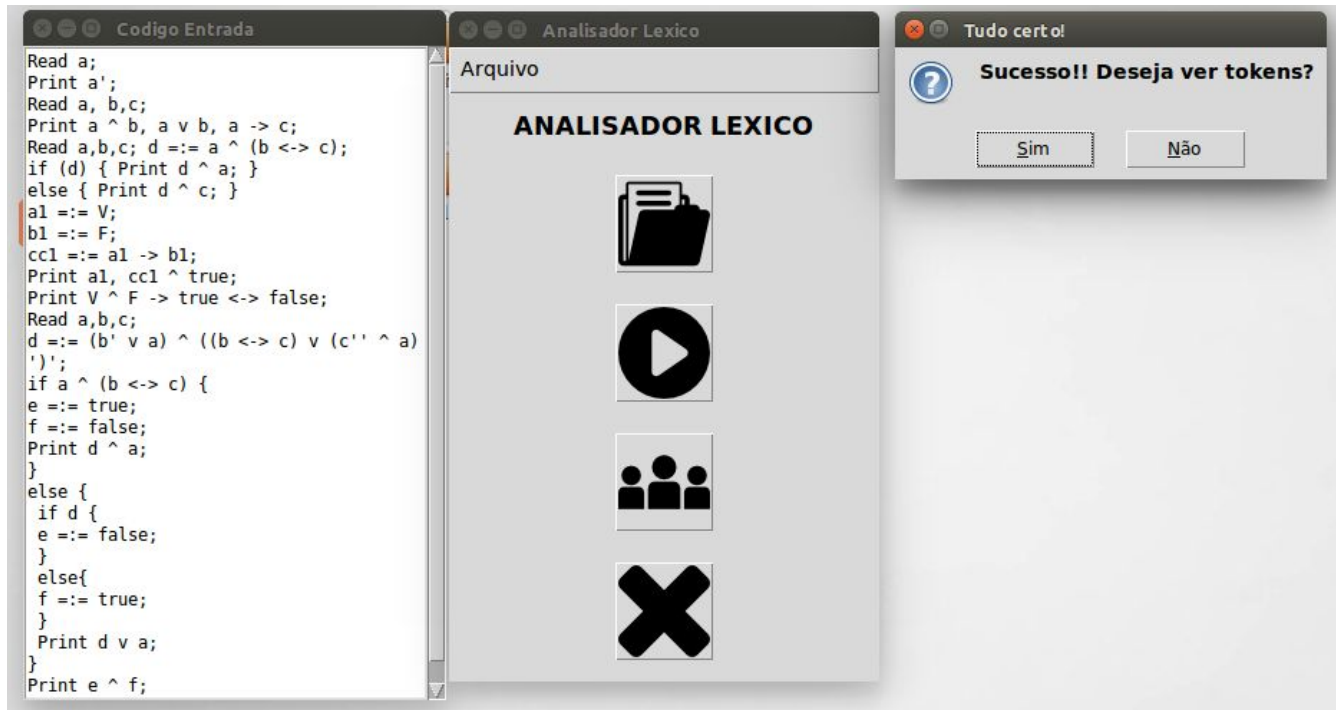


Figura 1: Analisador Léxico Sintático

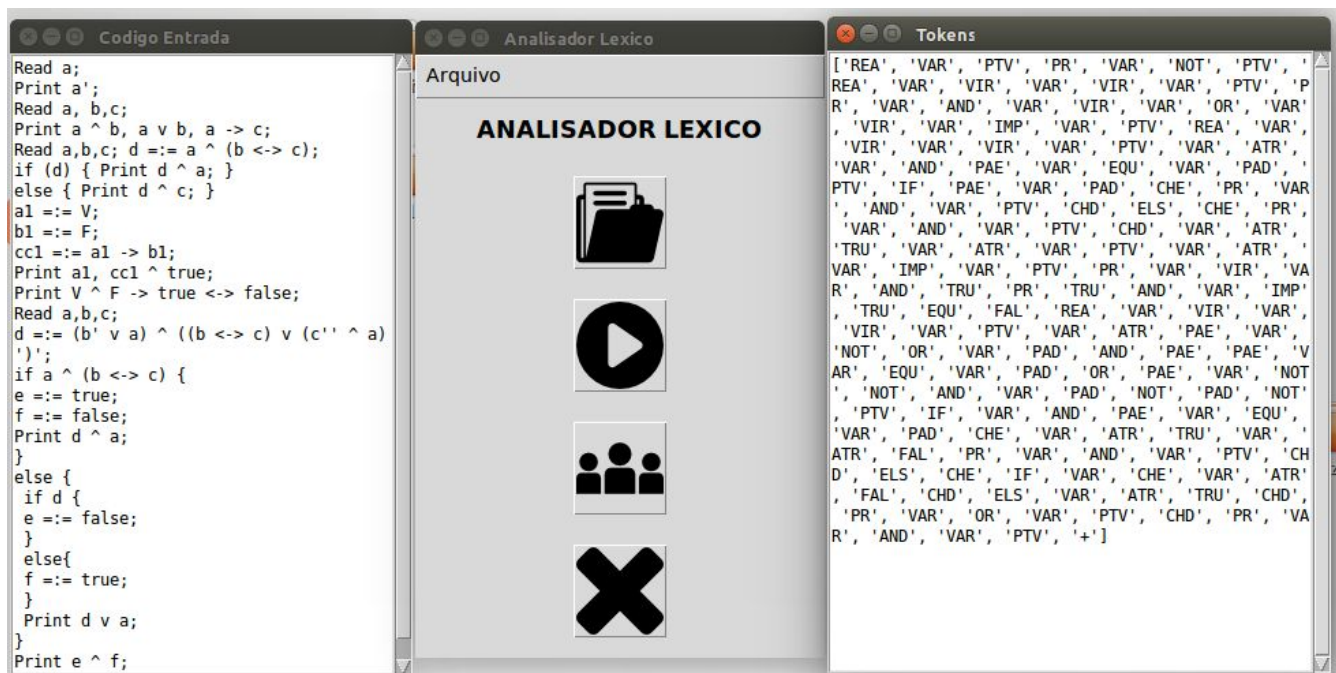


Figura 2: Analisador Léxico Sintático e Tokens

¹ **Python** é uma linguagem de programação de alto nível,, interpretada, imperativa, orientada a objetos, funcional, de tipagem dinâmica e forte. Foi lançada por Guido van Rossum em 1991. Atualmente possui um modelo de desenvolvimento comunitário, aberto e gerenciado pela organização sem fins lucrativos Python Software Foundation.

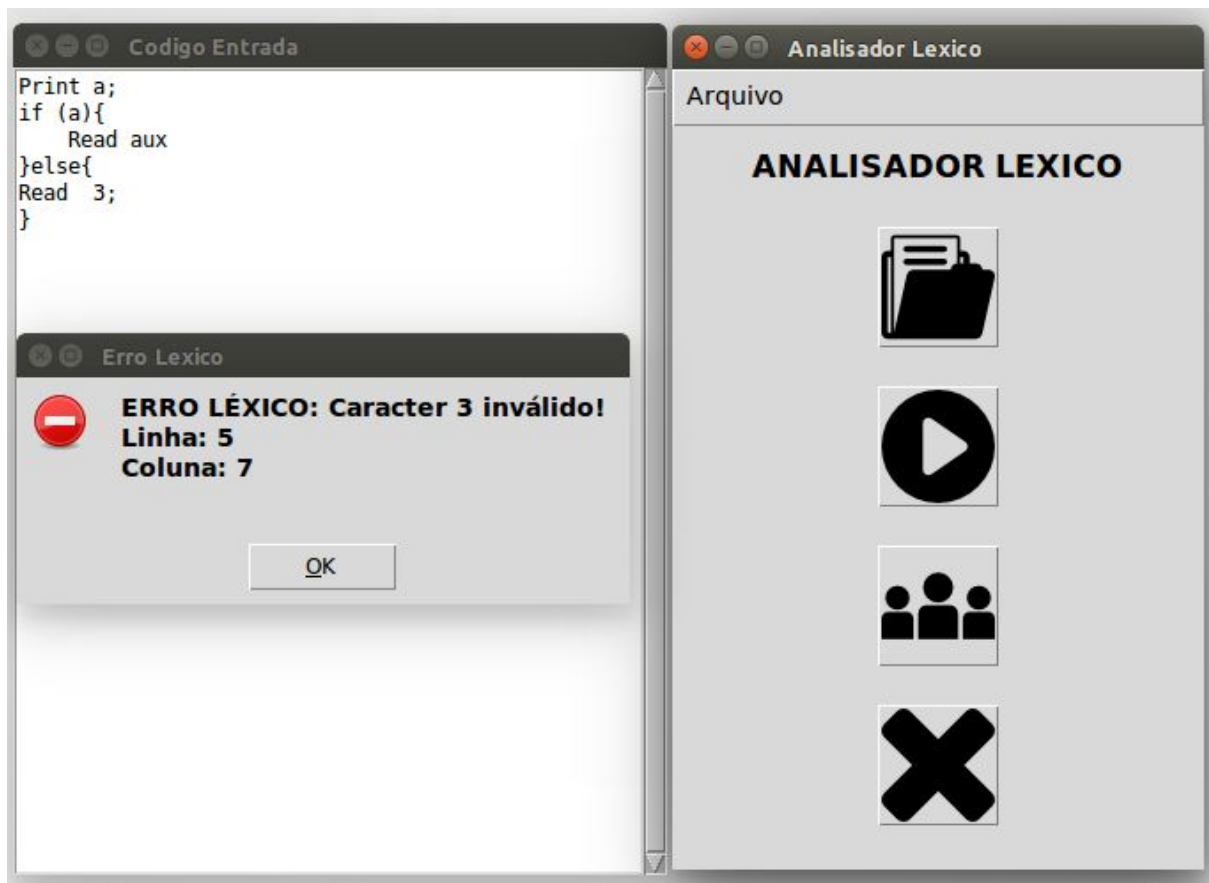
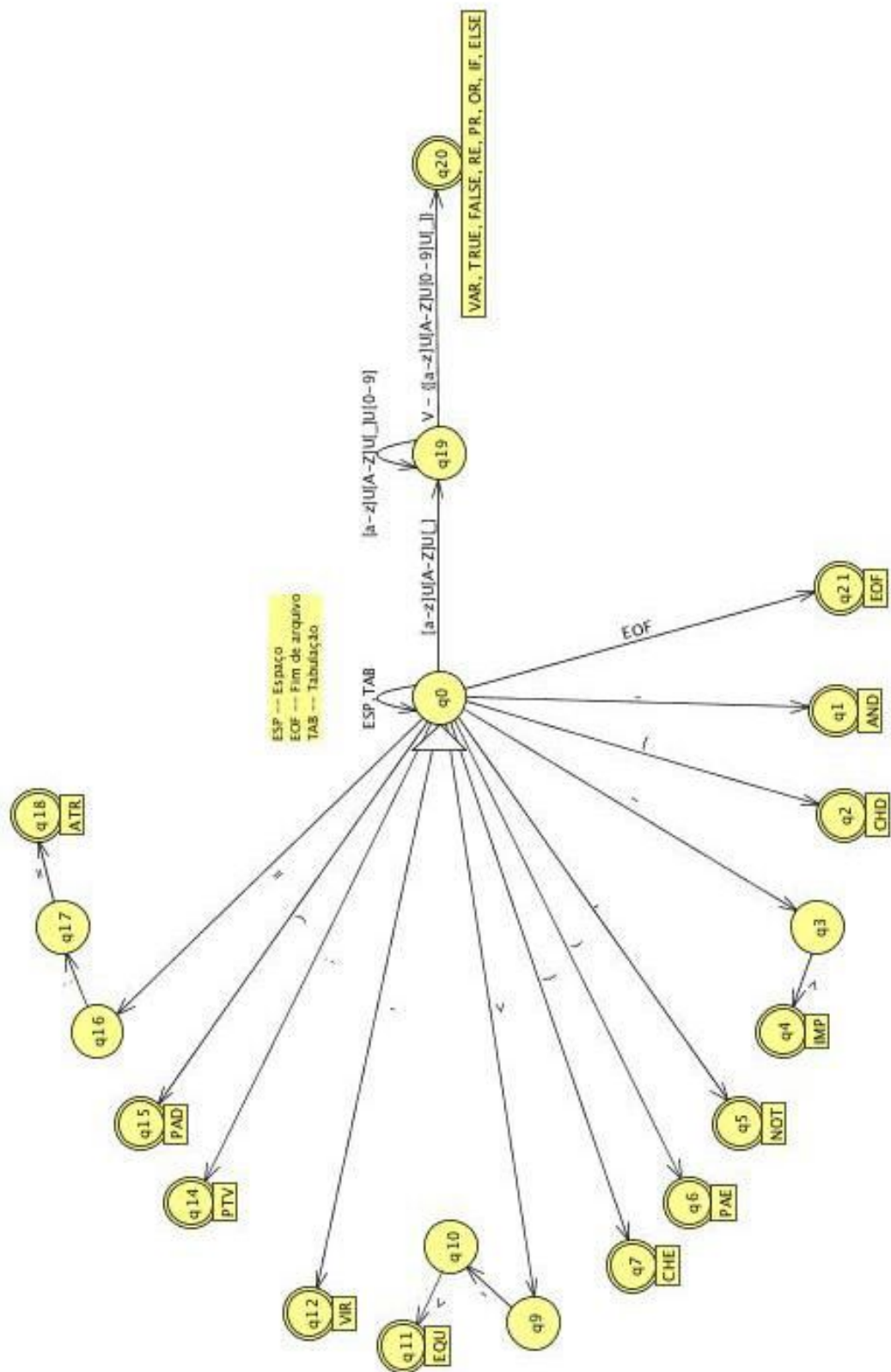


Figura 2: Analisador Léxico Sintático acusando erro

3. AUTÔMATO FINITO



3. TABELA DE EQUIVALÊNCIA

SÍMBOLO	SIGNIFICADO	TOKEN
‘	Negação	NOT
^	Conjunção	AND
v	Disjunção	OR
->	Implicação	IMP
<->	Equivalência	EQU
=:=	Atribuição	ATR
,	Vírgula	VIR
{	Chave Esquerda	CHE
}	Chave Direita	CHD
Print	Print	PRI
Read	Read	REA
EOF	Fim de Arquivo	EOF
(Parênteses Esquerdo	PAE
)	Parênteses Direito	PAD
;	Ponto e Vírgula	PTV
[0] U [F] U [False]	Falso	FAL
[1] U [T] U [True]	Verdadeiro	TRU
[a-z] U [A-Z] U [0-9]	Variável	VAR

4. GRAMÁTICA LIVRE DE CONTEXTO

Nesta Seção, poderá ser observado passo a passo o desenvolvimento da gramática deste trabalho.

4.1 GRAMÁTICA INICIAL

$G = (A, B, P, S)$

$A = \{S, E0, E1, E2, A\}$.

$B = \{VAR, NOT, AND, OR, IMP, EQU, ATR, VIR, CHE, CHD, PRI, REA, EOF, PAE, PAD, PTV, FAL, TRU, IF, ELS\}$.

$P = \{$

$S \rightarrow \langle PRI \rangle \mid E0 \langle PTV \rangle \mid \langle PRI \rangle E0 \langle EOF \rangle$

$E0 \rightarrow E1 \mid E0 \langle VIR \rangle E0$

$E1 \rightarrow \langle PAE \rangle E1 \langle PAD \rangle \mid E1 \langle AND \rangle E1 \mid E1 \langle OR \rangle E1 \mid E1 \langle IMP \rangle E1 \mid E1 \langle EQU \rangle E1 \mid E1 \langle NOT \rangle \mid E2$

$E2 \rightarrow \langle TRU \rangle \mid \langle FAL \rangle \mid \langle VAR \rangle$

$S \rightarrow \langle VAR \rangle \langle ATR \rangle E1 \langle PTV \rangle \mid \langle VAR \rangle \langle ATR \rangle E1 \langle EOF \rangle$

$S \rightarrow \langle REA \rangle A \langle PTV \rangle \mid \langle REA \rangle A \langle EOF \rangle$

$A \rightarrow \langle VIR \rangle \langle VAR \rangle \mid \langle VIR \rangle \langle VAR \rangle A$

$S \rightarrow \langle IF \rangle \langle CHE \rangle S \langle CHD \rangle \langle ELS \rangle \langle CHE \rangle S \langle CHD \rangle$

$\}$

4.2 GRAMÁTICA APÓS ELIMINAÇÃO DA AMBIGUIDADE

$G = (A, B, P, S)$

$A = \{S, E0, E1, E2, E3, E4, FINAL\}$.

$B = \{VAR, NOT, AND, OR, IMP, EQU, ATR, VIR, CHE, CHD, PRI, REA, EOF, PAE, PAD, PTV, FAL, TRU, IF, ELS\}$.

$P = \{$

$S \rightarrow PRINT \mid VAR \mid READ \mid S$

$READ \rightarrow \langle REA \rangle E0$

$PRINT \rightarrow \langle PRI \rangle E0$

$VAR \rightarrow \langle VAR \rangle \langle ATR \rangle E1$

$E0 \rightarrow E0 \langle VIR \rangle E1 \mid E1$

$E0^* \rightarrow E0^* \langle VIR \rangle E0^*$

$S^* \rightarrow \langle IF \rangle \langle CHE \rangle S \langle CHD \rangle \langle ELS \rangle \langle CHE \rangle S \langle CHD \rangle$

$E1 \rightarrow E1 \langle EQU \rangle E2 \mid E2$

$E2 \rightarrow E2 \langle IMP \rangle E3 \mid E3$

$$\begin{aligned}
E3 &\rightarrow E3 \text{ <AND> } E4 \mid E3 \text{ <OR> } E4 \mid E4 \\
E4 &\rightarrow E4 \mid \text{ <PAE> } E4 \text{ <PAD> } \mid \text{ FINAL} \\
\text{FINAL} &\rightarrow \text{ <VAR> } \mid \text{ <TRU> } \mid \text{ <FALS>} \\
&\}
\end{aligned}$$

4.2 GRAMÁTICA APÓS ELIMINAÇÃO DE RECURSÃO À ESQUERDA

$G = (A, B, P, S)$

$A = \{S, S', E0, E1, E1^*, E2, E2^*, E3, E3^*, E4, E4^*, \text{FINAL}\}.$

$B = \{ \text{VAR, NOT, AND, OR, IMP, EQU, ATR, VIR, CHE, CHD, PRI, REA, EOF, PAE, PAD, PTV, FAL, TRU, IF, ELS} \}.$

$P = \{$

$S' \rightarrow S \text{ <EOF>}$

$S \rightarrow S \text{ PRINT} \mid \text{ READ} \mid \text{ <IF> <PAE> } E1 \text{ <PAD> <CHE> } S \text{ <CHD> <ELS> <CHE> } S \text{ <CHD>} \mid \text{ VAR}$

$\text{READ} \rightarrow \text{ <REA> } E1$

$\text{PRINT} \rightarrow \text{ <PRI> } E0$

$\text{VAR} \rightarrow \text{ <VAR> <ATR> } E1$

$E0 \rightarrow E1$

$E0 \rightarrow E1 E0^*$

$E0^* \rightarrow \text{ <VIR> } E1 E0^* \mid \text{ <VIR> } E1$

$E1 \rightarrow E2 \mid E2 E1^*$

$E1^* \rightarrow \text{ <EQU> } E2 E1^* \mid \text{ <EQU> } E2$

$E2 \rightarrow E2 \mid E3 E2^*$

$E2^* \rightarrow \text{ <IMP> } E3 E2^* \mid \text{ <IMP> } E3$

$E3 \rightarrow E4 \mid E4 E3^*$

$E3^* \rightarrow \text{ <AND> } E4 E3^* \mid \text{ <AND> } E4 \mid \text{ <OR> } E4 E3^* \mid \text{ <OR> } E4$

$E4 \rightarrow \text{ FINAL} \mid \text{ <PAE> } E1 \text{ <PAD> } \mid \text{ FINAL } E4^* \mid \text{ <OR> } E4 \mid \text{ <PAE> } E1 \text{ <PAD> } E4^*$

$E4^* \rightarrow \text{ NOT } E4^* \mid \text{ NOT}$

$\text{FINAL} \rightarrow \text{ <VAR> } \mid \text{ <TRU> } \mid \text{ <FALS>}$

$\}$

4.3 GRAMÁTICA APÓS FATORAÇÃO

$G = (A, B, P, S')$.

$A = \{S, S', PRINT, READ, VAR, IF, E0, E0', E0^*, E1, E1', E1^*, E2, E2', E2^*, E3, E3', E3^*, E4, E4', E4^*, FINAL\}$.

$B = \{VAR, NOT, AND, OR, IMP, EQU, ATR, VIR, CHE, CHD, PRI, REA, EOF, PAE, PAD, PTV, FAL, TRU, IF, ELS\}$.

$P = \{$

$S' \rightarrow S \langle EOF \rangle$

$S \rightarrow PRINT \mid READ \mid VAR \mid IF$

$PRINT \rightarrow \langle PR \rangle E1 \langle PTV \rangle S$

$READ \rightarrow \langle REA \rangle E0 \langle PTV \rangle S$

$VAR \rightarrow \langle VAR \rangle \langle ATR \rangle E1 \langle PTV \rangle S$

$E0 \rightarrow E1 E0'$

$E0' \rightarrow E0^* \mid \epsilon$

$E0^* \rightarrow \langle VIR \rangle E1 E0'$

$E1 \rightarrow E2 E1'$

$E1' \rightarrow E1^* \mid \epsilon$

$E1^* \rightarrow \langle EQU \rangle E2 E1'$

$E2 \rightarrow E3 E2'$

$E2' \rightarrow E2^* \mid \epsilon$

$E2^* \rightarrow \langle IMP \rangle E3 E2'$

$E3 \rightarrow E4 E3'$

$E3' \rightarrow E3^* \mid \epsilon$

$E3^* \rightarrow \langle AND \rangle E4 E3' \mid \langle OR \rangle E4 E3'$

$E4 \rightarrow FINAL E4' \mid \langle PAE \rangle E1 \langle PAD \rangle E4'$

$E4' \rightarrow E4^* \mid \epsilon$

$E4^* \rightarrow \langle NOT \rangle E4'$

$FINAL \rightarrow \langle VAR \rangle \mid \langle FAL \rangle \mid \langle TRU \rangle$

$IF \rightarrow \langle IF \rangle E1 \langle CHE \rangle S \langle CHD \rangle IF'$

$IF' \rightarrow \langle ELS \rangle \langle CHE \rangle S \langle CHD \rangle S \mid S$

$\}$