**Lista de Tokens utilizados**

<OP\_UN> = {!}

<OP\_BIN> = { \* , / , && , || , < , > , <= , >= , == , !=,% }

<OP\_UN\_BIN> = { + , - }

<CARACTER> = ‘( a | b |...| z | 0|...| 9 |...) ‘

<VAR> = (a | b | ... | z)+

<NUM> = (0 | 1 |...| 9)\* ( . ) ? (0 | 1 |...| 9)+

<AP> = { ( }

<FP> = { ) }

<AC> = { { }

<FC> = { } }

<PT\_VIRG> = { ; }

<DOIS\_PT> = {:}

<IGUAL> = { = }

<WHILE> = while

<FOR> = for

<IF> = if

<SWITCH> = switch

<CASE> = case

<DO> = do

<EOF> = EOF

<INCREMENTO> = {++}

<DECREMENDO> = {--}

**Gramática**

programa -> comando programa

| comando

comando -> while

| if

| swicth\_case

| do\_while

| for

| atribuicao <PT\_VIRG>

exp -> operador2 exp R

| <AP> exp <FP> R

| <VAR> R

| <NUM> R

| <SINAL><NUM> R

R -> operador1 exp R | λ

operador1 -> <OP\_UN\_BIN>

| <OP\_BIN>

operador2 -> <OP\_UN\_BIN>

| <OP\_UN>

while -> <WHILE><AP> exp <FP> bloco

bloco -> <AC> programa <FC>

| comando

if -> <IF><AP> exp <FP> bloco

switch\_case -> <SWITCH><AP><VAR><FP><AC> listaCase <FC>

case -> <CASE> identCase <DOIS\_PT> bloco listaCase

listaCase -> λ

| case listaCase

identCase -> <NUM>

| <VAR>

| <CARACTER>

atribuicao -> <VAR><IGUAL> exp

for -> <FOR><AP> atribuicao <PT\_VIRG> exp <PT\_VIRG> atribuicao <FP> bloco

do\_while -> <DO><AC> bloco <FC><WHILE><AP> exp <FP>