

programa  $\rightarrow$  **programa** ID { corpo } **fimprog**

corpo  $\rightarrow$  declaracao corpo

| comando corpo

| declaracao

| comando

declaracao  $\rightarrow$  tipo listaVar ;

listaVar  $\rightarrow$  ID ( = expressão )? listaVarContinuacao

listaVarContinuacao  $\rightarrow$  , ID ( = expressao )? listaVarContinuacao

|  $\epsilon$

tipo  $\rightarrow$  **inteiro**

| **decimal**

| **texto**

| **logico**

comando  $\rightarrow$  atribuicao ;

| chamadaFuncao ;

| cmdImprimir ;

| cmdLer ;

| cmdSe

| cmdEnquanto

| cmdPara

| cmdRetornar ;

| { corpo }

atribuicao  $\rightarrow$  ID = expressao

| ID é expressao

chamadaFuncao  $\rightarrow$  ID ( listaArg )

| ID ( )

listaArg  $\rightarrow$  expressao listaArgRest

listaArgRest  $\rightarrow$  , expressao listaArgRest

|  $\epsilon$

cmdImprimir  $\rightarrow$  **imprimir** ( expressao )

cmdLer  $\rightarrow$  **ler** ( ID )

cmdSe  $\rightarrow$  **se** ( expressao ) { corpo } ( **senao** { corpo } )?

cmdEnquanto  $\rightarrow$  **enquanto** ( expressao ) { corpo }

cmdPara  $\rightarrow$  **para** ( inicializacao ; condicao ; incremento ) { corpo }

| **para** ( inicializacao ; condicao ; ) { corpo }

| **para** ( inicializacao ; ; incremento ) { corpo }

| **para** ( ; condicao ; incremento ) { corpo }

| **para** ( ; ; ) { corpo }

inicializacao  $\rightarrow$  declaracao

| atribuicao

condicao  $\rightarrow$  expressao

incremento  $\rightarrow$  atribuicao

| expressao

cmdRetornar  $\rightarrow$  **retornar** expressao

| **retornar**

expressao  $\rightarrow$  exprOu

exprOu  $\rightarrow$  exprE exprOuRest

exprOuRest  $\rightarrow$  || exprE exprOuRest

|  $\epsilon$

exprE  $\rightarrow$  exprIgual exprERest

exprERest  $\rightarrow$  && exprIgual exprERest

|  $\epsilon$

exprIgual  $\rightarrow$  exprRel exprIgualRest

exprIgualRest  $\rightarrow$  == exprRel exprIgualRest

| != exprRel exprIgualRest

|  $\epsilon$

exprRel  $\rightarrow$  exprAdd exprRelRest

exprRelRest  $\rightarrow$  < exprAdd exprRelRest

| <= exprAdd exprRelRest

| > exprAdd exprRelRest

| >= exprAdd exprRelRest

|  $\epsilon$

exprAdd  $\rightarrow$  exprMult exprAddRest

exprAddRest  $\rightarrow$  + exprMult exprAddRest

| - exprMult exprAddRest

|  $\epsilon$

$\text{exprMult} \rightarrow \text{exprUnario exprMultRest}$

$\text{exprMultRest} \rightarrow * \text{exprUnario exprMultRest}$

$\quad \mid / \text{exprUnario exprMultRest}$

$\quad \mid \varepsilon$

$\text{exprUnario} \rightarrow ! \text{primario}$

$\quad \mid - \text{primario}$

$\quad \mid \text{primario}$

$\text{primario} \rightarrow ( \text{expressao} )$

$\quad \mid \text{ID} ( \text{listaArg?} )$

$\quad \mid \text{ID} ( )$

$\quad \mid \text{NUM}$

$\quad \mid \text{TEXTO}$

$\quad \mid \textbf{verdadeiro}$

$\quad \mid \textbf{falso}$

$\quad \mid \textbf{nulo}$

$\text{NUM} \rightarrow [0-9]^+ ( ' . ' [0-9]^+ ) ?$

$\text{TEXTO} \rightarrow ' " ' ( \text{qualquer caractere exceto aspas ou quebra de linha, com suporte a } \backslash " \backslash \backslash \backslash n \backslash t ) ' " '$

$\text{ID} \rightarrow \text{letra} ( \text{letra} \mid \text{dígito} \mid ' \_ ' ) ^*$