T = {HASTA, DESDE, MIENTRAS, PARA, INCREMENTO, ENTERO, ( , ) , { , } , principal, DECIMAL, CHAR, BOOLEANO, CADENA, id, + , - , / , \*, ^ , OLN, OL, OR, imprimir, leer, TD, HACER, SI, SINO\_SI, SINO, = }

NT = {P, PRINCIPAL, CODIGO, CODE, C, DECLARAR\_VAR, VAR\_DECLARADA, WH, DWH, IF, FOR, IMPRI, BODY\_IMPRI, FIN\_IMPRI}

--------------------------------------------------------------------------------

P-> PRINCIPAL CODIGO

PRINCIPAL -> principal ()

CODIGO -> {C}

C -> CODE C

| ε

CODE-> DECLARAR\_VAR

| VAR\_DECLARADA

| WH

| DWH

| IF

| FOR

| IMPRI

| LEER

--------------------------------------------------------------------------------

IMPRI -> imprimir BODY\_IMPRI

BODY\_IMPRI -> (VALOR FIN\_IMPRI

FIN\_IMPRI -> + VALOR FIN\_IMPRI

| ) ;

LEER -> leer BODY\_LEER ;

BODY\_LEER -> (id)

--------------------------------------------------------------------------------

WH-> MIENTRAS CONDICION CODIGO

DWH-> HACER\_BLOCK MIENTRAS\_BLOCK

HACER\_BLOCK -> HACER CODIGO

MIENTRAS\_BLOCK -> MIENTRAS CONDICION;

--------------------------------------------------------------------------------

IF-> SI IF\_BODY END\_IF

IF\_BODY -> CONDICION CODIGO

END\_IF -> SINO\_SI IF\_BODY END\_IF |

SINO CODIGO |

ε

--------------------------------------------------------------------------------

FOR-> DESDE\_BLOCK HASTA\_BLOCK INCREMENTO\_BLOCK

DESDE\_BLOCK -> DESDE DECLARAR\_VAR

HASTA\_BLOCK -> HASTA CONDI

INCREMENTO\_BLOCK -> INCREMENTO ENTERO CODIGO

--------------------------------------------------------------------------------

DECLARAR\_VAR -> TD BODY\_VAR FIN\_VAR

BODY\_VAR -> id ASIG

ASIG -> = VALOR

| ε

FIN\_VAR -> , BODY\_VAR FIN\_VAR

| ;

--------------------------------------------------------------------------------

VAR\_DECLARADA -> id CHANGE\_VALOR ;

CHANGE\_VALOR -> OP\_INC\_DEC

| = VALOR

--------------------------------------------------------------------------------

--------------------------------------------------------------------------------

CONDI -> NOT CONDI\_BODY CONDI2

CONDI\_BODY -> VALOR SEG\_DATO

SEG\_DATO -> OR NOT VALOR

| ε

CONDI2 -> OL CONDI\_BODY CONDI2

| ε

NOT -> !

| ε

--------------------------------------------------------------------------------

CONDICION -> (CONDI)

--------------------------------------------------------------------------------

VALOR -> M SUMA

SUMA -> + M SUMA

| - M SUMA

| ε

M-> P MULTI

MULTI -> \* P MULTI

| / P MULTI

| ε

P -> S POTENCIA

POTENCIA -> ^ P

| ε

S -> ASIG\_VALOR

| - ASIG\_VALOR

ASIG\_VALOR ->

ENTERO

| DECIMAL

| CHAR

| BOOLEANO

| CADENA

| id

| (VALOR)

--------------------------------------------------------------------------------