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 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)
