FO(STATLISTP) = FO(STATLIST) = {EOF,}

FO(IDLISTP) = FO(IDLIST) = {],)}

FO(ASSIGNLISTP) = {;, EOF, }, ELSE, END}

FO(EXPRLISTP) = {)}

G(<STATLIST> EOF) = {ASSIGN, PRINT, READ, FOR, IF, }

G(<STATLIST> ::= <STAT> <STATLISTP>) =

{ASSIGN, PRINT, READ, FOR, IF, }

G(<STATLISTP> ::= ; <STAT> <STATLISTP>) =

{;}

G(STATLISTP ::= ε) = {EOF, }

G(<STAT> ::= ASSIGN <ASSIGNLIST>) = {ASSIGN}

G(<STAT> ::= PRINT (<EXPRLIST>)) = {PRINT}

G(<STAT> ::= READ (<IDLIST>)) = {READ}

G(<STAT> ::= FOR <BEXPR> do <STAT>) = {FOR}

G(<STAT> ::= IF (<BEXPR>) <STAT> END) = {IF}

G(<STAT> ::= {<STATLIST>}) = {}

G(<ASSIGNLIST> ::= [<EXPR> to <IDLIST>] <ASSIGNLISTP>) =

{[}

G(<ASSIGNLISTP> ::= [<EXPR> to <IDLIST>] <ASSIGNLISTP>) =

<prog> ::= <statlist> EOF

<statlist> ::= <stat> <statlistp>

<statlistp> ::= ; <stat> <statlistp> | ε

<stat> ::= assign <assignlist>
| print (<exprlist>)
| read (<idlist>)
| for (ID := <expr> ; <bexpr>) do <stat>
| if (<bexpr>) <stat> else <stat> end
| { <statlist> }

<assignlist> ::= [<expr> to <idlist>] <assignlistp>

<assignlistp> ::= [<expr> to <idlist>] <assignlistp> | ε

<idlist> ::= ID <idlistp>

<idlistp> ::= , ID <idlistp> | ε

<bexpr> ::= RELOP <expr> <expr>

<expr> ::= + (<exprlist>) | - <expr> <expr>
| * (<exprlist>) | / <expr> <expr>
| NUM | ID

<exprlist> ::= <expr> <exprlistp>

<exprlistp> ::= , <expr> <exprlistp> | ε

<S> ::= ID := <EXPR> ;
└ ε

<D> ::= ELSE <STAT>
└ ε