Extension : Recursive procedure

#################################################################

New Grammar:

<program> ::= PROGRAM <declarations> <procedures> BEGIN <statementSequence> END

<declarations> ::= VAR ident AS <type> SC <declarations>

| ε

<type> ::= INT | BOOL

<procedures> ::= <procedure><procedures>

| ε

<procedure> ::== PROCEDURE ident LP <parameters> RP SC <declarations> BEGIN <statementSequence> END

<parameters> ::= <parameter> <moreparameter>

| ε

<moreparameter> ::= SC <parameter><parameters>

| ε

<parameter> ::= ident COL <type>

<statementSequence> ::= <statement> SC <statementSequence>

| ε

<statement> ::= <assignment>

| <ifStatement>

| <whileStatement>

| <writeInt>

<assignment> ::= ident ASGN <assignment'>

<assignment'> ::= <expression> | READINT

<ifStatement> ::= IF <expression> THEN <statementSequence> <elseClause> END

<elseClause> ::= ELSE <statementSequence>

| ε

<whileStatement> ::= WHILE <expression> DO <statementSequence> END

<writeInt> ::= WRITEINT <expression>

<expression> ::= <simpleExpression><expression'>

<expression'>::= COMPARE <expression> | ε

<simpleExpression> ::= <term> <simpleExpression'>

<simpleExpression'> ::= ADDITIVE <simpleExpression>

| ε

<term> ::= <factor> <term'>

<term'> ::= MULTIPLICATIVE <term> | ε

<factor> ::= ident

| num

| boollit

| LP <expression> RP