1. Inside program()

a. Check for program symbol, continue if true, otherwise output error

b. Check for identifier, continue of there is an id, otherwise output error, for badId as well

2. Loop

a. looking for dec symbol, exit when find a dec symbol, a begin symbol (that means there is no dec part), an id (begin of dec part without dec symbol) , and an end symbol (kill the program)

3. If there is a dec symbol, call decPart()

4. Inside decPart()

a. if there is no end symbol, call listVar()

b. if there is no end symbol, but if it an id or badId, call checkIdenitifier() to check if it an id or badId. Then get the next symbol and loop listVar() until the next symbol is not comma.

c. Return to decPart()

5. If there is no end symbol, check for colon, then call typePart() (#7b)

6. If there is no end symbol, check for semicolon

a. If there is, is the next symbol id or badId, call checkIdentifier, then call decPart()

b. If there is, is the next symbol begin symbol, output error, then return to program()

7. If there is no semicolon, but the next symbol is an id or badId, call checkIdentifier, then call decPart() (#4)

7a. listVar()

a. Check for end symbol, if not end symbol, check for an id or badId

b. Call checkIdentifier for badId

c. Check next symbol for comma, if it is, call listVar() (#7a), else if it an id or badId, output error and call listVar()

d. Return to respective function call

7b. typePart()

a. Check for end symbol, if not end symbol, check for type symbol

b. Return to decPart() (#4)

8. Return to program() If there is a begin symbol, call statementPart() (#10), otherwise output error

9. If there is end symbol, exit program

10. statementPart()

a. Check for end symbol, if true return to program(), otherwise call statement() (#11)

b. When returned from statement(), Loop

i. check if symbol is not end symbol and is semicolon, call statement() (#11) again

ii. check symbol for end symbol, if not call statement() (#11) again

iii. Exit loop by end symbol

11. statement()

a. Check for end symbol, if true return to statementPart(), otherwise if the symbol is Read symbol (call input() (#12)), Write symbol (call output() (#13)), an id or badId (call assignment() (#14)), and others (return to statementPart() (#10)

b. Return to statementPart()

12. input()

a. Check for end symbol, if true return to statement() (#11)

b. Check for left parenthesis, output error if not

c. Check for end symbol, if true return to statement() (#11), otherwise call listVar() if it is an id or badId and output error if it not

d. Check for right parenthesis, output error if not

e. Return to statement() (#11)

13. output()

a. it is similar to input()

14. assignment()

a. Check for end symbol, if not end symbol, check for assign symbol

b. If it is assign symbol, call expression() (#15), otherwise output error and call expression() #15

c. Return to statement() (#11)

15. expression()

a. Check for end symbol, if not end symbol, check for num, id, badId

b. If it is, call operation() (#16), otherwise output error and call operation() (#16)

c. Return to assignment() (#14)

16. operation()

a. Check for end symbol, if not end symbol, check for id, num, badId, otherwise output error

b. Check for end symbol, if not end symbol, check for addition, substraction, multiply, division

c. If it is, call operation again, otherwise output error

d. Return to expression() (#15)