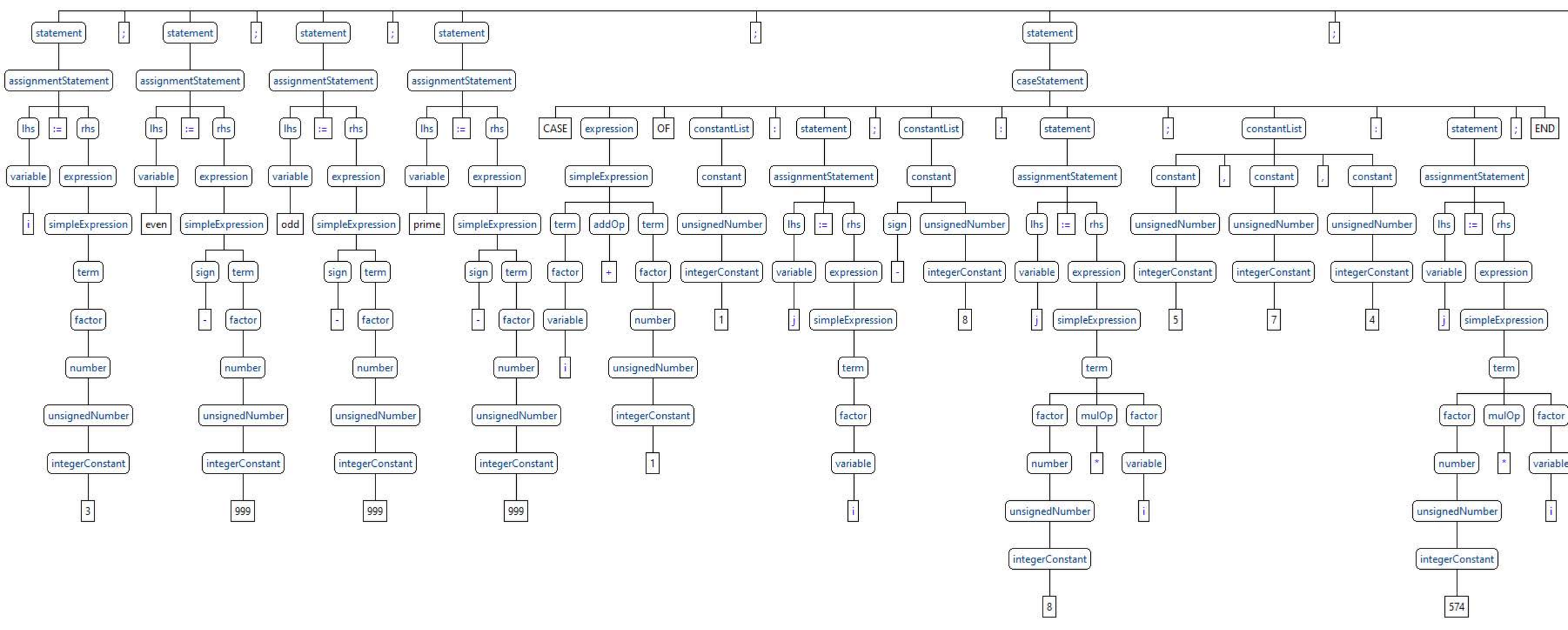
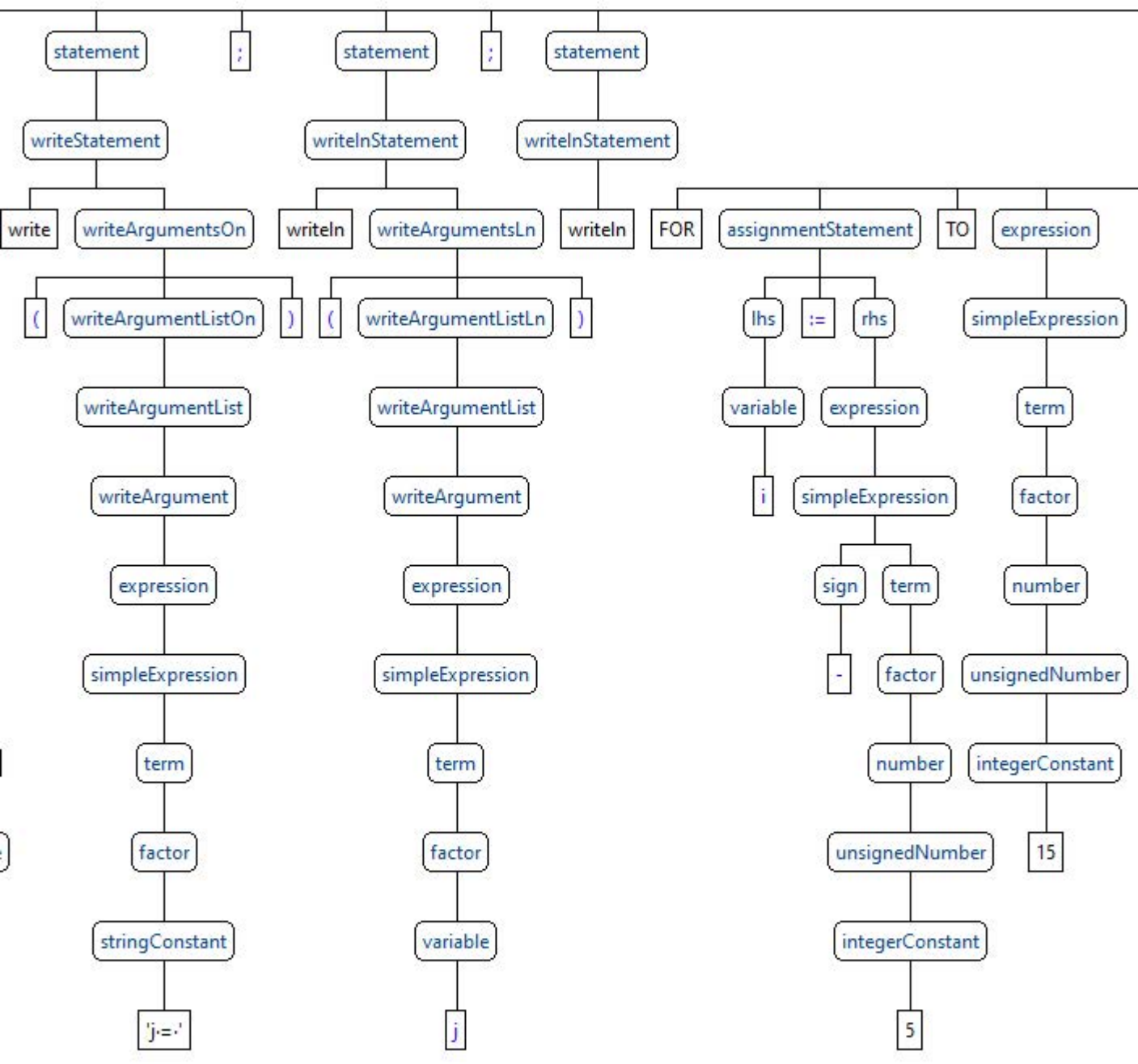
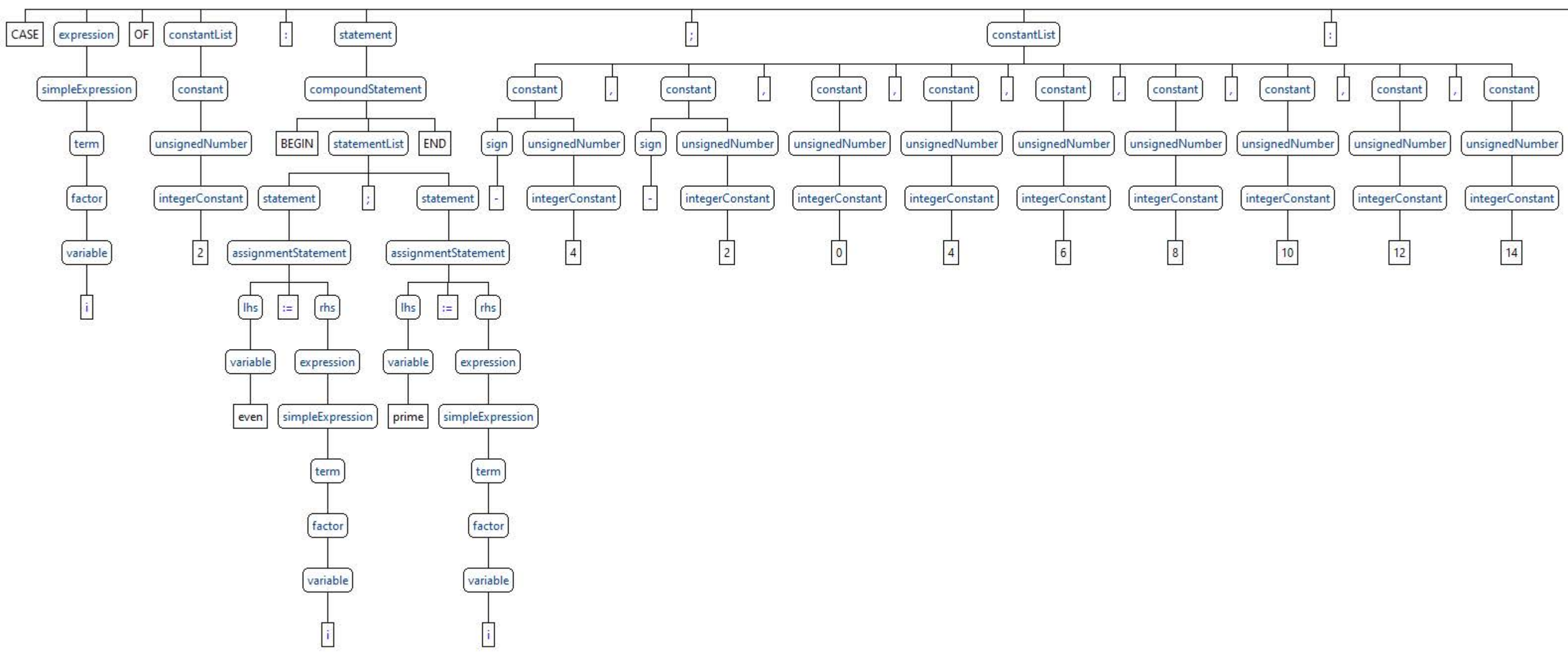
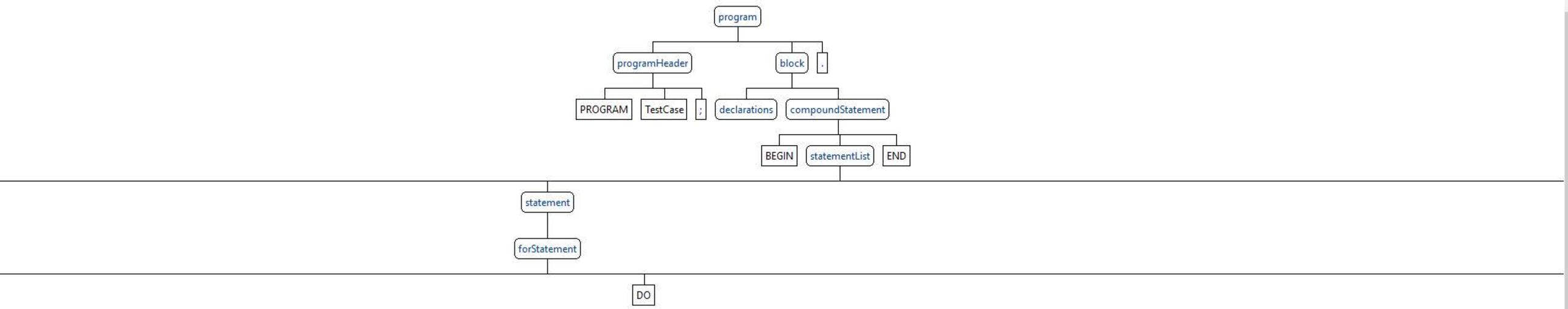
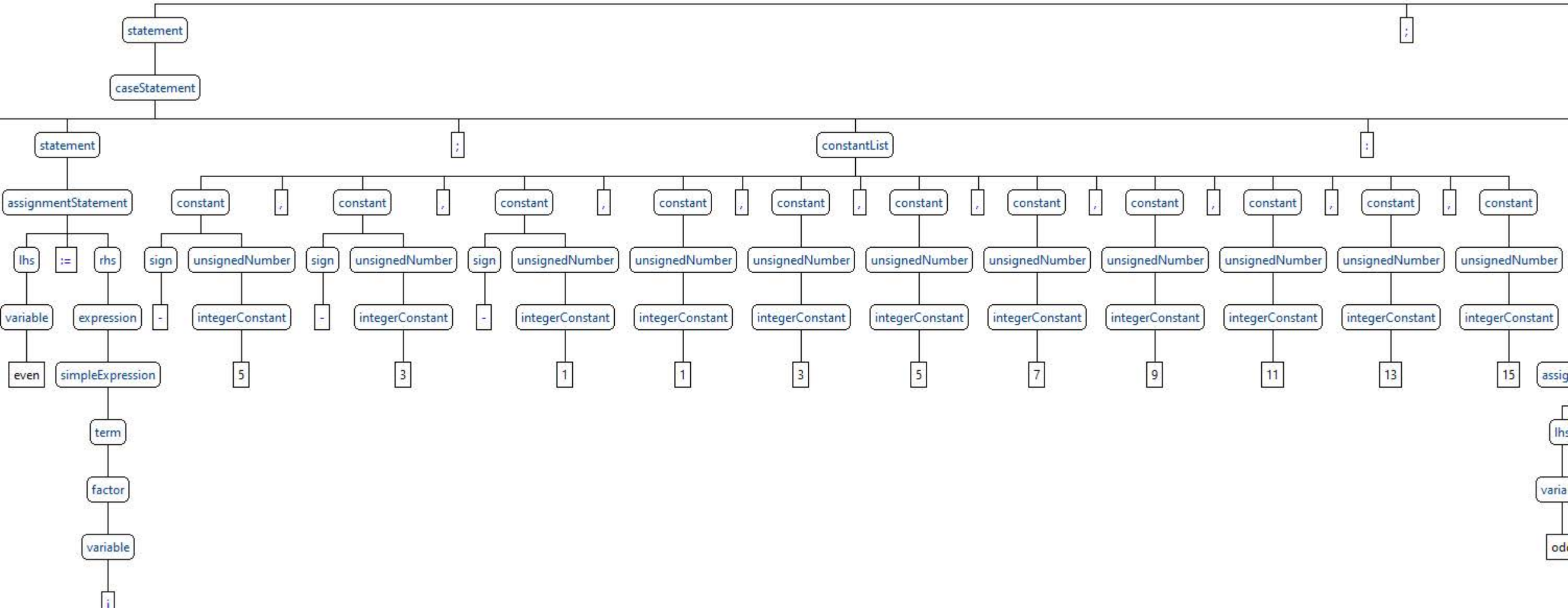
Sorry for the awful cropping. I used the snipping tool to take screenshots of the Parse Tree window, but since it couldn't fit all on one screen, this was the best I could do.

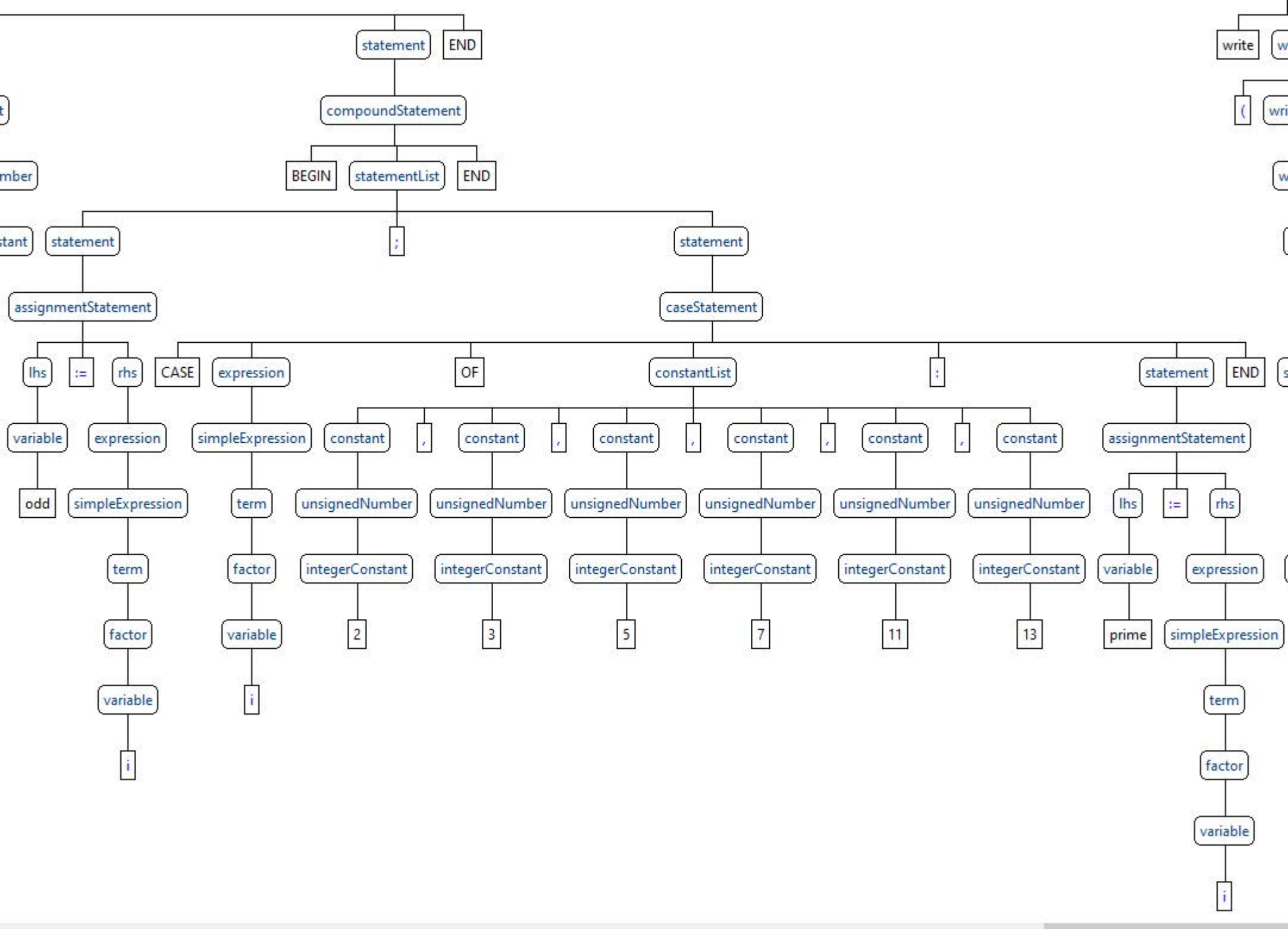

statement
;
assignmentStatement
lhs
:=
rhs
variable
expression
i
simpleExpression
term
factor
number
unsignedNumber
integerConstant
3
even
sign
-
999
odd
prime
caseStatement
CASE
OF
constantList
constant
addOp
+
1
j
8
mulOp
*
5
7
4
574
END

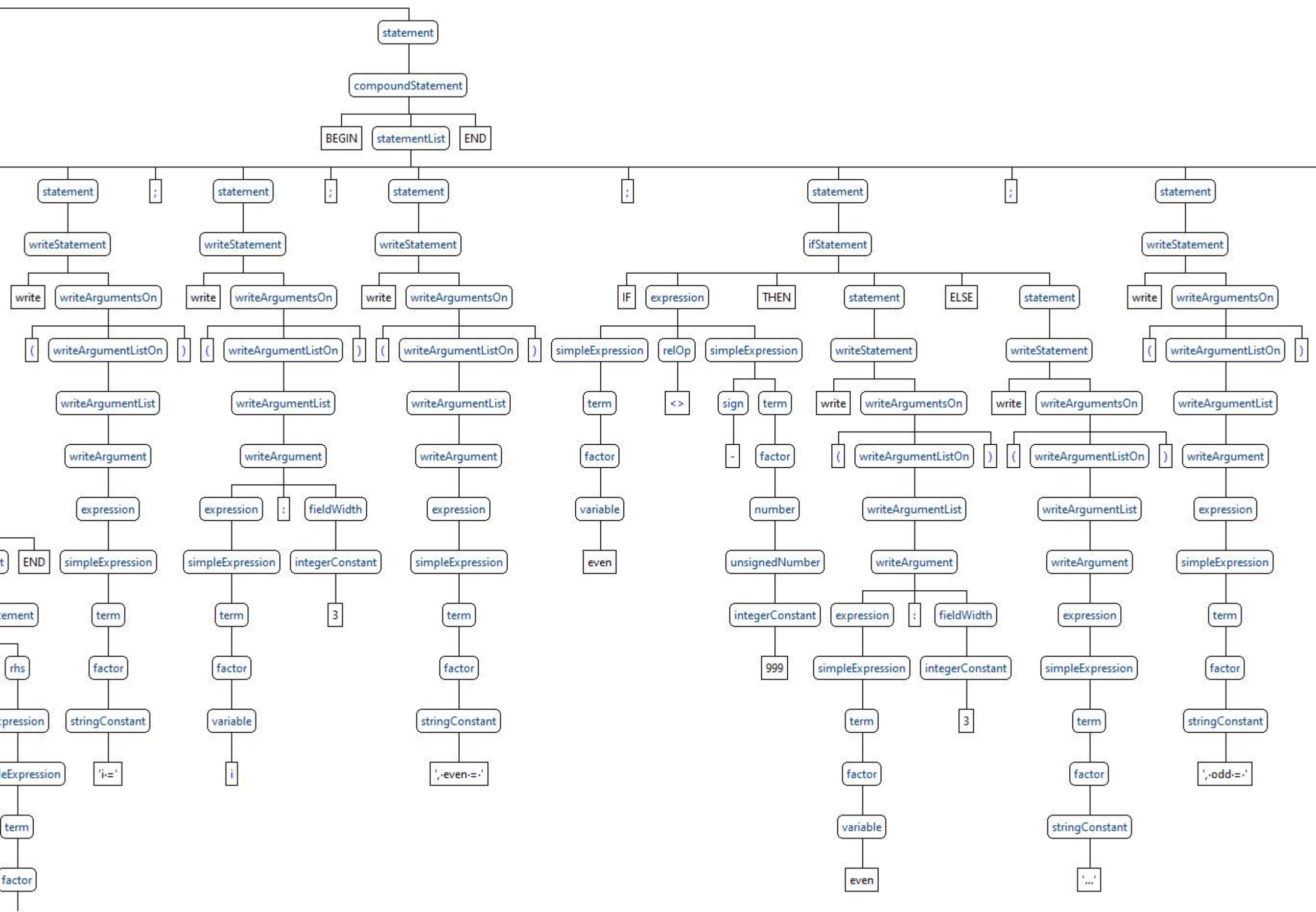

statement
;
statement
;
statement
writeStatement
writeInStatement
writeInStatement
write
writeArgumentsOn
writeln
writeArgumentsLn
writeln
FOR
assignmentStatement
TO
expression
(
writeArgumentListOn
)
(
writeArgumentListLn
)
lhs
:=
rhs
simpleExpression
writeArgumentList
writeArgumentList
variable
expression
term
writeArgument
writeArgument
i
simpleExpression
factor
expression
expression
sign
term
number
simpleExpression
simpleExpression
-
factor
unsignedNumber
term
term
number
integerConstant
factor
factor
unsignedNumber
15
stringConstant
variable
integerConstant
'j·=·'
j
5

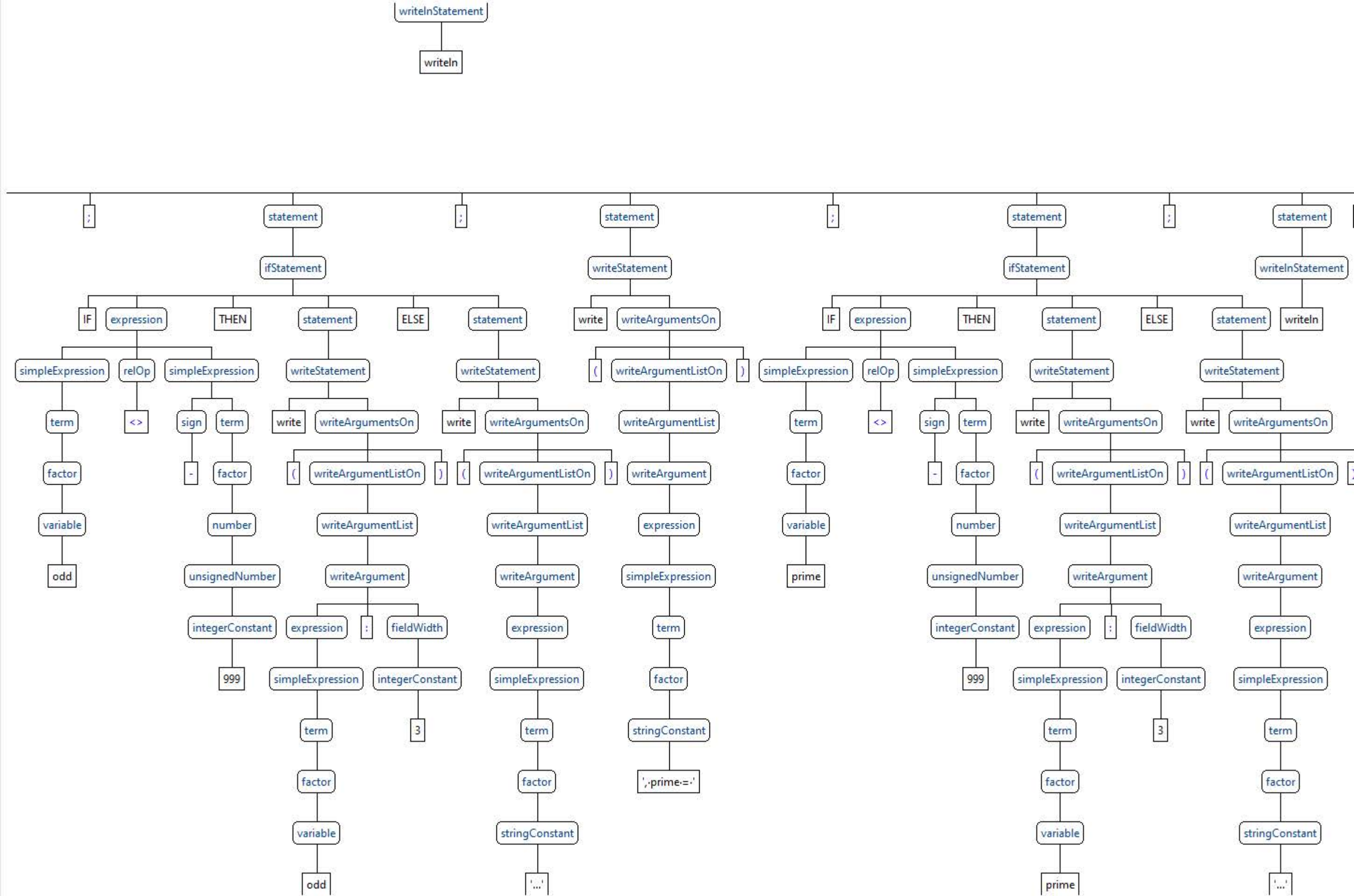

CASE
expression
simpleExpression
term
factor
variable
i
OF
constantList
constant
unsignedNumber
integerConstant
2
:
statement
compoundStatement
BEGIN
statementList
statement
assignmentStatement
lhs
:=
rhs
variable
even
expression
simpleExpression
term
factor
variable
i
;
statement
assignmentStatement
lhs
:=
rhs
variable
prime
expression
simpleExpression
term
factor
variable
i
;
constant
sign
-
unsignedNumber
integerConstant
4
,
constant
sign
-
unsignedNumber
integerConstant
2
,
constant
unsignedNumber
integerConstant
0
,
constant
unsignedNumber
integerConstant
4
,
constant
unsignedNumber
integerConstant
6
,
constant
unsignedNumber
integerConstant
8
,
constant
unsignedNumber
integerConstant
10
,
constant
unsignedNumber
integerConstant
12
,
constant
unsignedNumber
integerConstant
14
constantList
:

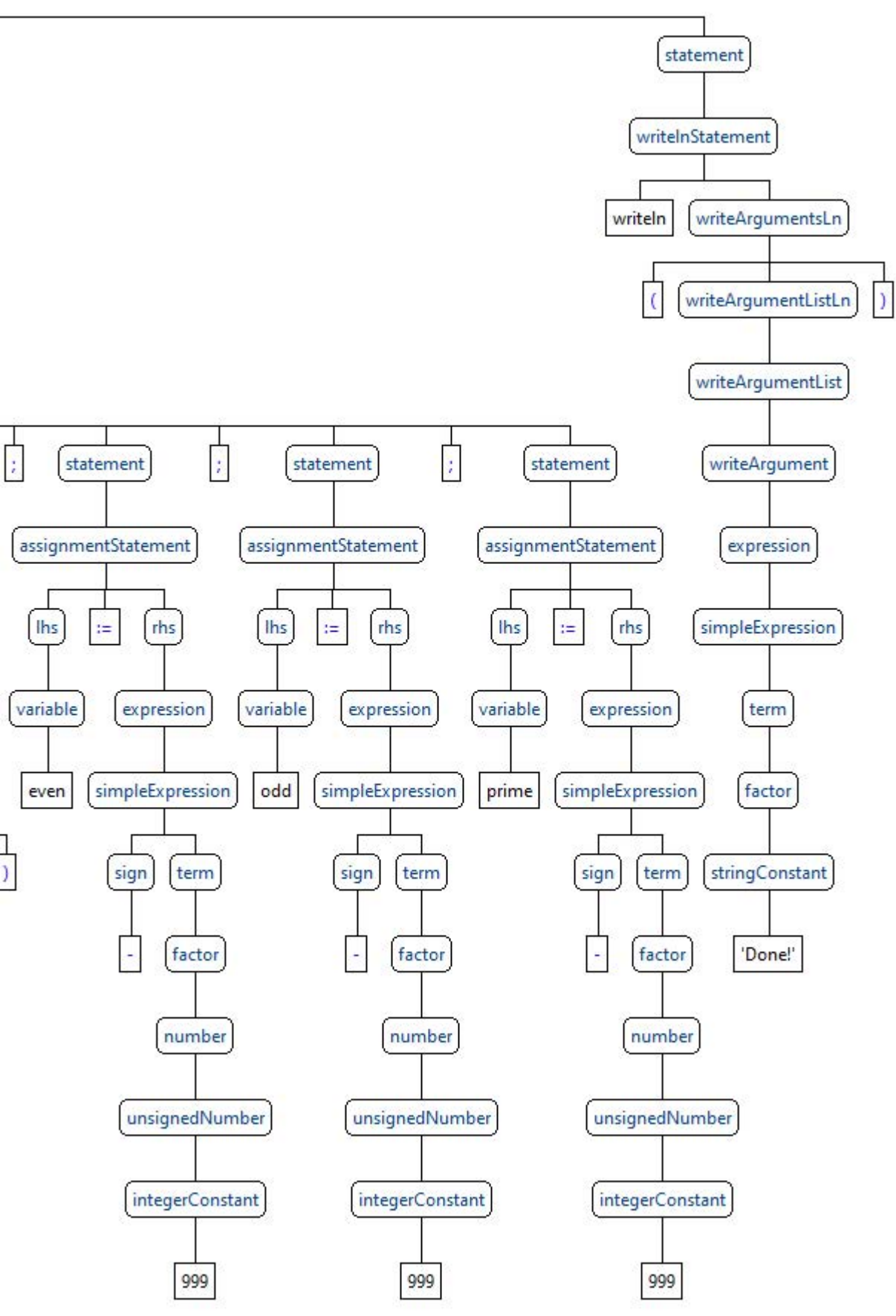

program
programHeader
block
.
PROGRAM
TestCase
;
declarations
compoundStatement
BEGIN
statementList
END
statement
forStatement
DO


statement
caseStatement
;
statement
;
constantList
:
assignmentStatement
constant
,
constant
,
constant
,
constant
,
constant
,
constant
,
constant
,
constant
,
constant
,
constant
,
constant
,
constant
,
constant
lhs
:=
rhs
sign
unsignedNumber
sign
unsignedNumber
sign
unsignedNumber
unsignedNumber
unsignedNumber
unsignedNumber
unsignedNumber
unsignedNumber
unsignedNumber
unsignedNumber
unsignedNumber
unsignedNumber
unsignedNumber
variable
expression
-
integerConstant
-
integerConstant
-
integerConstant
integerConstant
integerConstant
integerConstant
integerConstant
integerConstant
integerConstant
integerConstant
integerConstant
integerConstant
integerConstant
even
simpleExpression
5
3
1
1
3
5
7
9
11
13
15
term
factor
variable


statement
END
compoundStatement
BEGIN
statementList
END
statement
;
statement
assignmentStatement
caseStatement
lhs
:=
rhs
CASE
expression
OF
constantList
:
statement
END
variable
expression
simpleExpression
constant
,
constant
,
constant
,
constant
,
constant
,
constant
assignmentStatement
odd
simpleExpression
term
unsignedNumber
unsignedNumber
unsignedNumber
unsignedNumber
unsignedNumber
unsignedNumber
lhs
:=
rhs
term
factor
integerConstant
integerConstant
integerConstant
integerConstant
integerConstant
integerConstant
variable
expression
factor
variable
2
3
5
7
11
13
prime
simpleExpression
variable
term
factor
variable
i
i
write
(
term
factor
variable
i


statement
compoundStatement
BEGIN
statementList
END
statement
;
statement
;
statement
;
statement
;
statement
writeStatement
writeStatement
writeStatement
ifStatement
writeStatement
write
writeArgumentsOn
write
writeArgumentsOn
write
writeArgumentsOn
IF
expression
THEN
statement
ELSE
statement
write
writeArgumentsOn
(
writeArgumentListOn
)
(
writeArgumentListOn
)
(
writeArgumentListOn
)
simpleExpression
relOp
simpleExpression
writeStatement
writeStatement
(
writeArgumentListOn
)
writeArgumentList
writeArgumentList
writeArgumentList
term
<>
sign
term
write
writeArgumentsOn
write
writeArgumentsOn
writeArgumentList
writeArgument
writeArgument
writeArgument
factor
-
factor
(
writeArgumentListOn
)
(
writeArgumentListOn
)
writeArgument
expression
expression
:
fieldWidth
expression
variable
number
writeArgumentList
writeArgumentList
expression
END
simpleExpression
simpleExpression
integerConstant
simpleExpression
even
unsignedNumber
writeArgument
writeArgument
simpleExpression
term
term
3
term
integerConstant
expression
:
fieldWidth
expression
term
rhs
factor
factor
factor
999
simpleExpression
integerConstant
simpleExpression
factor
stringConstant
variable
stringConstant
term
3
term
stringConstant
'i·='
i
',·even·=·'
factor
factor
',·odd·=·'
term
variable
stringConstant
factor
even
'...'


writeInStatement
writeln
;
statement
;
statement
;
statement
;
statement
ifStatement
writeStatement
ifStatement
writeInStatement
IF
expression
THEN
statement
ELSE
statement
write
writeArgumentsOn
IF
expression
THEN
statement
ELSE
statement
writeln
simpleExpression
relOp
simpleExpression
writeStatement
writeStatement
(
writeArgumentListOn
)
simpleExpression
relOp
simpleExpression
writeStatement
writeStatement
term
<>
sign
term
write
writeArgumentsOn
write
writeArgumentsOn
writeArgumentList
term
<>
sign
term
write
writeArgumentsOn
write
writeArgumentsOn
factor
-
factor
(
writeArgumentListOn
)
(
writeArgumentListOn
)
writeArgument
factor
-
factor
(
writeArgumentListOn
)
(
writeArgumentListOn
variable
number
writeArgumentList
writeArgumentList
expression
variable
number
writeArgumentList
writeArgumentList
odd
unsignedNumber
writeArgument
writeArgument
simpleExpression
prime
unsignedNumber
writeArgument
writeArgument
integerConstant
expression
:
fieldWidth
expression
term
integerConstant
expression
:
fieldWidth
expression
999
simpleExpression
integerConstant
simpleExpression
factor
999
simpleExpression
integerConstant
simpleExpression
term
3
term
stringConstant
term
3
term
factor
variable
factor
stringConstant
factor
variable
factor
stringConstant
odd
'...'
',·prime·=·'
prime
'...'


statement
writeInStatement
writeln
writeArgumentsLn
(
writeArgumentListLn
)
writeArgumentList
writeArgument
expression
simpleExpression
term
factor
stringConstant
'Done!'
;
statement
;
statement
;
statement
assignmentStatement
lhs
:=
rhs
variable
even
expression
simpleExpression
sign
-
term
factor
number
unsignedNumber
integerConstant
999
assignmentStatement
lhs
:=
rhs
variable
odd
expression
simpleExpression
sign
-
term
factor
number
unsignedNumber
integerConstant
999
assignmentStatement
lhs
:=
rhs
variable
prime
expression
simpleExpression
sign
-
term
factor
number
unsignedNumber
integerConstant
999