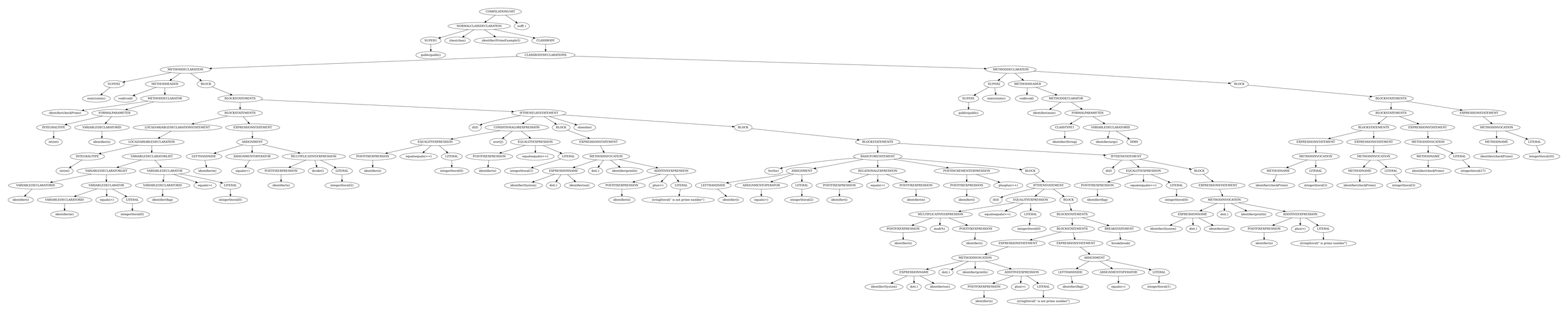

COMPILATIONUNIT
NORMALCLASSDECLARATION
eoff( )
SUPER1
class(class)
identifier(PrimeExample2)
CLASSBODY
public(public)
CLASSBODYDECLARATIONS
METHODDECLARATION
METHODDECLARATION
SUPER2
METHODHEADER
BLOCK
SUPER2
METHODHEADER
BLOCK
static(static)
void(void)
METHODDECLARATOR
BLOCKSTATEMENTS
SUPER1
static(static)
void(void)
METHODDECLARATOR
BLOCKSTATEMENTS
identifier(checkPrime)
FORMALPARAMETER
BLOCKSTATEMENTS
IFTHENELSESTATEMENT
public(public)
identifier(main)
FORMALPARAMETER
BLOCKSTATEMENTS
EXPRESSIONSTATEMENT
INTEGRALTYPE
VARIABLEDECLARATORID
LOCALVARIABLEDECLARATIONSTATEMENT
EXPRESSIONSTATEMENT
if(if)
CONDITIONALOREXPRESSION
BLOCK
else(else)
BLOCK
CLASSTYPE1
VARIABLEDECLARATORID
BLOCKSTATEMENTS
EXPRESSIONSTATEMENT
METHODINVOCATION
int(int)
identifier(n)
LOCALVARIABLEDECLARATION
ASSIGNMENT
EQUALITYEXPRESSION
oror(||)
EQUALITYEXPRESSION
EXPRESSIONSTATEMENT
BLOCKSTATEMENTS
identifier(String)
identifier(args)
DIMS
EXPRESSIONSTATEMENT
EXPRESSIONSTATEMENT
METHODINVOCATION
METHODNAME
LITERAL
INTEGRALTYPE
VARIABLEDECLARATORLIST
LEFTHANDSIDE
ASSIGNMENTOPERATOR
MULTIPLICATIVEEXPRESSION
POSTFIXEXPRESSION
equalsequals(==)
LITERAL
POSTFIXEXPRESSION
equalsequals(==)
LITERAL
METHODINVOCATION
BASICFORSTATEMENT
IFTHENSTATEMENT
METHODINVOCATION
METHODINVOCATION
METHODNAME
LITERAL
identifier(checkPrime)
integerliteral(20)
int(int)
VARIABLEDECLARATORLIST
VARIABLEDECLARATOR
identifier(m)
equals(=)
POSTFIXEXPRESSION
divide(/)
LITERAL
identifier(n)
integerliteral(0)
identifier(n)
integerliteral(1)
EXPRESSIONNAME
dot(.)
identifier(println)
ADDITIVEEXPRESSION
for(for)
ASSIGNMENT
RELATIONALEXPRESSION
POSTINCREMENTEXPRESSION
BLOCK
if(if)
EQUALITYEXPRESSION
BLOCK
METHODNAME
LITERAL
METHODNAME
LITERAL
identifier(checkPrime)
integerliteral(17)
VARIABLEDECLARATORID
VARIABLEDECLARATOR
VARIABLEDECLARATORID
equals(=)
LITERAL
identifier(n)
integerliteral(2)
identifier(System)
dot(.)
identifier(out)
POSTFIXEXPRESSION
plus(+)
LITERAL
LEFTHANDSIDE
ASSIGNMENTOPERATOR
LITERAL
POSTFIXEXPRESSION
equals(=)
POSTFIXEXPRESSION
POSTFIXEXPRESSION
plusplus(++)
IFTHENSTATEMENT
POSTFIXEXPRESSION
equalsequals(==)
LITERAL
EXPRESSIONSTATEMENT
identifier(checkPrime)
integerliteral(1)
identifier(checkPrime)
integerliteral(3)
identifier(i)
VARIABLEDECLARATORID
equals(=)
LITERAL
identifier(flag)
integerliteral(0)
identifier(n)
stringliteral(" is not prime number")
identifier(i)
equals(=)
integerliteral(2)
identifier(i)
identifier(m)
identifier(i)
if(if)
EQUALITYEXPRESSION
BLOCK
identifier(flag)
integerliteral(0)
METHODINVOCATION
identifier(m)
integerliteral(0)
MULTIPLICATIVEEXPRESSION
equalsequals(==)
LITERAL
BLOCKSTATEMENTS
EXPRESSIONNAME
dot(.)
identifier(println)
ADDITIVEEXPRESSION
POSTFIXEXPRESSION
mod(%)
POSTFIXEXPRESSION
integerliteral(0)
BLOCKSTATEMENTS
BREAKSTATEMENT
identifier(System)
dot(.)
identifier(out)
POSTFIXEXPRESSION
plus(+)
LITERAL
identifier(n)
identifier(i)
EXPRESSIONSTATEMENT
EXPRESSIONSTATEMENT
break(break)
identifier(n)
stringliteral(" is prime number")
METHODINVOCATION
ASSIGNMENT
EXPRESSIONNAME
dot(.)
identifier(println)
ADDITIVEEXPRESSION
LEFTHANDSIDE
ASSIGNMENTOPERATOR
LITERAL
identifier(System)
dot(.)
identifier(out)
POSTFIXEXPRESSION
plus(+)
LITERAL
identifier(flag)
equals(=)
integerliteral(1)
identifier(n)
stringliteral(" is not prime number")