Abstrakte Algebra für Minijava for Workgroups

Program :: ClassList
ClassList :: Class*
Class :: FieldList MethodList MainMethodList
FieldList :: Field*
MethodList :: Method*
MainMethodList :: MainMethod*
Field :: Type Name
Method :: Type Name ParameterList
MainMethod :: Name Name
ParameterList :: Parameter*
Parameter :: Type Name
Name :: **id**
Type = BasicType | ArrayType
BasicType = PrimitiveType | ClassType
ArrayType :: BasicType **int**
PrimitiveType = int | boolean | void
ClassType :: **id**

Statement = Block | EmptyStatement | IfStatement | ExpressionStatement | WhileStatement
| ReturnStatement
Block :: BlockStatementList
BlockStatementList :: BlockStatement*
BlockStatement = Statement | LocalVariableDeclaration
EmptyStatement :: Ø
IfStatement :: Expression Statement Statement?
ExpressionStatement :: Expression
WhileStatement :: Expression Statement
ReturnStatement :: Expression?
LocalVariableDeclaration :: Type Name Expression?

Expression = PrimaryExpression | MethodInvocation | FieldAccess | ArrayAccess |
BinaryExpression | UnaryExpression
PrimaryExpression = NewArrayExpression | NewObjectExpression | IntLiteral | BoolLiteral |
NullLiteral | ThisLiteral | VarRef
MethodInvocation :: Expression Name ArgumentList
ArgumentList :: Expression*
FieldAccess :: Expression Name
ArrayAccess :: Expression Expression
BinaryExpression :: Expression BINOP Expression
UnaryExpression :: UNOP Expression
BINOP = **=** | **||** | **&&** | **==** | **!=** | **<** | **<=** | **>** | **>=** | **+** | **-** | ***** | **/** | **%**

UNOP = **!** | **-**

NewArrayExpression :: ArrayType Expression
NewObjectExpression :: BasicType
IntLiteral :: **int**
BoolLiteral :: **bool**
NullLiteral :: Ø
ThisLiteral :: Ø
VarRef :: **id**