

G = (T, N, R, S)

T = {

CLASS | VARNAME | CBRACKETOPEN | CBRACKETCLOSED | SEMICOLON |
BRACKETOPEN | BRACKETCLOSED | COMMA | RETURN | WHILE | IF | ELSE | EQUAL |
NEW | DOT | THIS | SUPER | NOT | PLUSMINUSOPERATOR | BINARYOPERATOR | INT |
TRUE | FALSE | CHAR | STRING | NULL

}

N = {

class | fields | field | methods | method | type | params | param | block | statements | statement |
returnstatement | whilestatement | localvardecl | ifstatement | ifelsestatement | stmtExprExpr |
stmtExprStmt | stmtExpr | assign | new | methodcall | emptymethodcall | expressions |
expression | this | super | localorfieldvar | instvar | unary | binary | integer | bool | char | string |
jnull

}

R = {

class : CLASS VARNAME CBRACKETOPEN CBRACKETCLOSED
| CLASS VARNAME CBRACKETOPEN methods CBRACKETCLOSED
| CLASS VARNAME CBRACKETOPEN fields CBRACKETCLOSED
| CLASS VARNAME CBRACKETOPEN fields methods CBRACKETCLOSED

fields : field fields
| field

field : type VARNAME SEMICOLON

methods : method methods
| method

method : type VARNAME BRACKETOPEN BRACKETCLOSED block
| type VARNAME BRACKETOPEN params BRACKETCLOSED block

type : VARNAME

params : param COMMA params

param : type VARNAME

block : CBRACKETOPEN CBRACKETCLOSED
| CBRACKETOPEN statements CBRACKETCLOSED

statements : statement statements
| statement

statement : block
| returnstatement SEMICOLON
| whilestatement
| localvardecl SEMICOLON
| ifstatement
| ifelsestatement
| stmtExprStmt SEMICOLON

returnstatement : RETURN expression
| RETURN

whilestatement : WHILE BRACKETOPEN expression BRACKETCLOSE statement

localvardecl : type VARNAME

ifstatement : IF BRACKETOPEN expression BRACKETCLOSED statement

ifelsestatement : IF BRACKETOPEN expression BRACKETCLOSED statement ELSE
statement

stmtExprExpr : stmtExpr

stmtExprStmt : stmtExpr

stmtExpr : assign
| new

	methodcall
	emptyMethodCall
assign	: VARNAME EQUAL expression
new	: NEW type BRACKETOPEN BRACKETCLOSED
	NEW type BRACKETOPEN expressions BRACKETCLOSED
methodcall	: expression DOT VARNAME BRACKETOPEN expressions BRACKETCLOSED
	VARNAME BRACKETOPEN expressions BRACKETCLOSED
emptymethodcall	: expression DOT VARNAME BRACKETOPEN BRACKETCLOSED
	VARNAME BRACKETOPEN BRACKETCLOSED
expressions	: expression
	expression COMMA expressions
expression	: this
	super
	localorfieldvar
	instvar
	unary
	binary
	integer
	bool
	char
	string
	jnull
	stmtExprExpr
	OPENBRACKET expression CLOSEDDBRACKED
this	: THIS
super	: SUPER
localorfieldvar	: VARNAME
instvar	: expression DOT VARNAME
unary	: NOT expression
	PLUSMINUSOPERATOR expression
binary	: expression BINARYOPERATOR expression

| expression PLUSMINUSOPERATOR expression

integer : INT

bool : TRUE
| FALSE

char : CHAR

string : STRING

jnull : NULL

}

S = {

class

}