CSE4061 – Compiler Design New Grammar

Fatmanur Özdemir 150116046

Eray Ayaz 150116053

Berk Köylü 150116016

1 - Grammar of our programming language:

Start → method decls compoundstmt

decls → declaration ; decls | ε

declaration → type ID

type → int | double | short

methoddecls → declaration , methoddecls | ε

method → { methodStatements } | e

methodStatements → methodStatement methodStatements | e

methodStatement → type ID ( methoddecls ) { statements }

compoundstmt → { statements }

statements → statement statements | ε

statement → ifstmt | whilestmt | assignment | compoundstmt | methodCall | statementDec

statementDec → declaration ;

methodCall → ID ( optparameters ) ;

optparameters → params | ε

params → param A’

A’ → , param A’ | ε

param → ID | intNumber | doubleNumber | shortNumber

ifstmt → if ( booleanExp ) statement else statement

whilestmt → while ( booleanExp ) statement

booleanExp → arithmeticExp booleanOp arithmeticExp

booleanOp → < | > | <= | >= | == | !=

assignment → ID = arithmeticExp ; | ID = unaryExp ;

unaryExp → ++ ID ;

arithmeticExp → multexpr B’

B’ → + multexpr B’ | - multexpr B’ | ε

multexpr → simpleexpr C’

C’ → \* simpleexpr C’ | / simpleexpr C’ | ε

simpleexpr → ID |intNumber | doubleNumber | shortNumber | ( arithmeticExp )

2 - Lexical structure of our programming language:

* Keywords: if, else, while, int, double, short
* Identifiers: An identifier starts with a letter or underscore or dollar sign and continues with letter or digit. Any of the keywords cannot become identifier. Maximum length of the identifier should be at most 64 characters.

* Operator: The followings are operators: + - / \* = == < <= > >= !=
* Delimiters: whitespace
* Numbers: The numbers are defined as follows:

digit ←0|1|2|3|4|5|6|7|8|9

intNumber ← digit+ (the maximum integer number is 127)

doubleNumber ← digit+ (the maximum integer number is 255)

shortNumber ← digit+ (the maximum integer number is 63)