Program → main ( ) Compound-stmt

Compound-stmt → { InnerCompound-stmt }

InnerCompound-stmt → Variable statements

Variable → type id S; Variable | ε

S → , id S | = (Num | ID) Expression S | ε

statments → Stmt statments | ε

Stmt → ID Operator (+ | - | ε) (Num | ID) Expression ; | InnerCompound-stmt | If-Stmt |
selector-Stmt | until-stmt | loop-stmt | write-stmt | read-stmt

If-stmt → if ((Num | ID) Expression) so Compound-stmt Else-stmt

Else-stmt → else Compound-stmt | ε

Selector-Stmt → selector : ID { (selectStmt) (otherStmt)}

selectStmt → select NUM : compundStmt selectStmt | lambda

otherStmt → other : compundStmt | lambda

until-stmt → until (((Num | ID)   Expression) Compound-stmt

loopStmt → (ID & (Num | ID) expression & ID(++ | --)) compound_stmt

writeStmt → write (string stringValue);

stringValue → , id | ε

readStmt → read ((int|char|float) , id);

Expression → Operator Operand| lambda

type → int | void | float

Operator → RelOp | ArithOp

RelOp → <= | < | >= | > | <> | == | + | - | * | / | ++ | -- | += | -= | /=