**Context free grammar**

**Program Structure**

Program → int main ( ) { DeclList StmtList }

**Declarations**

DeclList → Declaration ; DeclList | ε

Declaration → IDENTIFIER : Type ;

Type → int | char | StructDecl

StructDecl → struct IDENTIFIER { DeclList }

**Statements**

StmtList → Stmt ; StmtList | ε

Stmt → SimplStmt | StructStmt

**Simple Statements**

SimplStmt → AssignStmt | IOStmt

**Assignment Statement**

AssignStmt → IDENTIFIER := Expression ;

Expression → Expression AddOp Term | Term

Term → Term MulOp Factor | Factor

Factor → ( Expression ) | IDENTIFIER | IntConst | CharConst

AddOp → + | -

MulOp → \* | / | %

**Input/Output Statement**

IOStmt → read ( IDENTIFIER ) ;

| write ( OutputContent ) ;

OutputContent → IDENTIFIER | StringConst

**Structured Statements**

StructStmt → CmpdStmt

| IfStmt

| WhileStmt

**Compound Statement**

StructStmt → CmpdStmt | IfStmt | WhileStmt

CmpdStmt → { StmtList }

**Conditional Statement**

IfStmt → if ( Condition ) Stmt

| if ( Condition ) Stmt else Stmt

Condition → Expression RelOp Expression

RelOp → < | <= | = | != | >= | >

**Loop Statement**

WhileStmt → while ( Condition ) Stmt

**Tokens**

IDENTIFIER → Letter IDENT\_Tail

IDENT\_Tail → Letter IDENT\_Tail | Digit IDENT\_Tail | ε

IntConst → Sign Digit Int\_Tail

Sign → + | - | ε

Int\_Tail → Digit Int\_Tail | ε

CharConst → ' SingleChar '

SingleChar → Letter | Digit

StringConst → " StringBody "

StringBody → Letter StringBody | Digit StringBody | ε

Letter → a | b | c | d | e | f | g | h | i | j | k | l | m | n | o | p | q | r | s | t | u | v | w | x | y | z

Digit → 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9

AddOp → + | -

MulOp → \* | / | %

RelOp → < | <= | = | != | >= | >

Keyword → int | main | char | struct | if | else | while | read | write

SpecialChar → ( | ) | { | } | : | ; | :=

**Nonterminals**

N = { Program, DeclList, Declaration, Type, StructDecl,

StmtList, Stmt, SimplStmt, AssignStmt, Expression, Term,

Factor, AddOp, MulOp, IOStmt, OutputContent, StructStmt,

CmpdStmt, IfStmt, Condition, RelOp, WhileStmt, IDENTIFIER,

IDENT\_Tail, IntConst, Sign, Int\_Tail, CharConst, SingleChar,

StringConst, StringBody, Letter, Digit}

**Terminals**

ε = { int, main, char, struct, if, else, while, read, write,

(, ), {, }, :, ;, :=, <, <=, =, !=, >=, >, +, -, \*, /, %,

', ", a, b, c, ..., z, 0, 1, 2, ..., 9 }