Jason Chan CS152 Phase 2 Grammar in Backus–Naur Form

```
⟨prog_start⟩          ::= ϵ
                        |  ⟨function⟩ ⟨prog_start⟩

⟨function⟩            ::= 'FUNCTION' ⟨identifier⟩ 'SEMICOLON' 'BEGIN_PARAMS'
                           ⟨declarations⟩ 'END_PARAMS' 'BEGIN_LOCALS' ⟨declarations⟩
                           'END_LOCALS' 'BEGIN_BODY' ⟨statements⟩ 'END_BODY'

⟨identifier⟩          ::= 'IDENT'

⟨identifiers⟩         ::= ⟨identifier⟩
                        |  ⟨identifier⟩ 'COMMA' ⟨identifiers⟩

⟨declaration⟩         ::= ⟨identifiers⟩ 'COLON' 'ARRAY' 'L_SQUARE_BRACKET'
                           'NUMBER' 'R_SQUARE_BRACKET' 'OF' 'INTEGER'
                        |  ⟨identifiers⟩ 'COLON' 'INTEGER'

⟨declarations⟩        ::= ϵ
                        |  ⟨declaration⟩ 'SEMICOLON' ⟨declarations⟩

⟨statements⟩          ::= ⟨statement⟩ 'SEMICOLON' ⟨statements⟩
                        |  ⟨statement⟩ 'SEMICOLON'

⟨statement⟩           ::= ⟨var⟩ 'ASSIGN' ⟨expression⟩
                        |  'IF' ⟨bool-expr⟩ 'THEN' ⟨statements⟩ 'ENDIF'
                        |  'IF' ⟨bool-expr⟩ 'THEN' ⟨statements⟩ 'ELSE' ⟨statements⟩
                           'ENDIF'
                        |  'WHILE' ⟨bool-expr⟩ 'BEGINLOOP' ⟨statements⟩ 'ENDLOOP'
                        |  'DO' 'BEGINLOOP' ⟨statements⟩ 'ENDLOOP' 'WHILE'
                           ⟨bool-expr⟩
                        |  'FOR' ⟨var⟩ 'ASSIGN' 'NUMBER' 'SEMICOLON' ⟨bool-expr⟩
                           'SEMICOLON' ⟨var⟩ 'ASSIGN' ⟨expression⟩ 'BEGINLOOP'
                           ⟨statements⟩ 'ENDLOOP'
                        |  'READ' ⟨vars⟩
                        |  'WRITE' ⟨vars⟩
                        |  'CONTINUE'
                        |  'RETURN' ⟨expression⟩
```

```
⟨bool-expr⟩              ::= ⟨relation-and-expr⟩
                          |  ⟨relation-and-expr⟩ 'OR' ⟨bool-expr⟩

⟨relation-and-expr⟩      ::= ⟨relation-expr⟩
                          |  ⟨relation-expr⟩ 'AND' ⟨relation-and-expr⟩

⟨relation-expr⟩          ::= ⟨expression⟩ ⟨comp⟩ ⟨expression⟩
                          |  'TRUE'
                          |  'FALSE'
                          |  'L_PAREN' ⟨bool-expr⟩ 'R_PAREN'
                          |  'NOT' ⟨expression⟩ ⟨comp⟩ ⟨expression⟩
                          |  'NOT' 'TRUE'
                          |  'NOT' 'FALSE'
                          |  'NOT' 'L_PAREN' ⟨bool-expr⟩ 'R_PAREN'

⟨comp⟩                   ::= 'EQ'
                          |  'NEQ'
                          |  'LT'
                          |  'GT'
                          |  'LTE'
                          |  'GTE'

⟨expressions⟩            ::= ⟨expression⟩
                          |  ⟨expression⟩ 'COMMA' ⟨expressions⟩

⟨expression⟩             ::= ⟨multiplicative-expr⟩
                          |  ⟨multiplicative-expr⟩ 'ADD' ⟨expression⟩
                          |  ⟨multiplicative-expr⟩ 'SUB' ⟨expression⟩

⟨multiplicative-expr⟩    ::= ⟨term⟩
                          |  ⟨term⟩ 'MULT' ⟨multiplicative-expr⟩
                          |  ⟨term⟩ 'DIV' ⟨multiplicative-expr⟩
                          |  ⟨term⟩ 'MOD' ⟨multiplicative-expr⟩

⟨term⟩                   ::= ⟨var⟩
                          |  'NUMBER'
                          |  'L_PAREN' ⟨expression⟩ 'R_PAREN'
                          |  'SUB' ⟨var⟩
```

```
                      |  'SUB' 'NUMBER'
                      |  'SUB' 'L_PAREN' ⟨expression⟩ 'R_PAREN'
                      |  ⟨identifier⟩ 'L_PAREN' ⟨expressions⟩ 'R_PAREN'
                      |  ⟨identifier⟩ 'L_PAREN' 'R_PAREN'

⟨vars⟩               ::= ⟨var⟩
                      |  ⟨var⟩ 'COMMA' ⟨vars⟩

⟨var⟩                ::= ⟨identifier⟩
                      |  ⟨identifier⟩ 'L_SQUARE_BRACKET' ⟨expression⟩
                         'R_SQUARE_BRACKET'
```