# Ledgard Syntax

The syntax of the ledgard programming language is defined by the following BNF grammar:

```
⟨program⟩               ::= ‘program’ ⟨decl-list⟩ ‘begin’ ⟨stmt-list⟩ ‘end’ ‘;’

⟨decl-list⟩             ::= ⟨declaration⟩ | ⟨decl-list⟩ ⟨declaration⟩

⟨declaration⟩           ::= ⟨identifier-list⟩ ‘:’ ⟨type⟩ ‘;’

⟨identifier-list⟩       ::= ⟨identifier⟩ | ⟨identifier-list⟩ ‘,’ ⟨identifier⟩

⟨type⟩                  ::= ⟨simple-type⟩ | ⟨array-type⟩

⟨simple-type⟩           ::= ‘integer’ | ‘boolean’

⟨array-type⟩            ::= ‘array’ ‘[’ ⟨bounds⟩ ‘]’ ‘of’ ⟨type⟩

⟨bounds⟩                ::= ⟨integer-literal⟩ ‘..’ ⟨integer-literal⟩

⟨stmt-list⟩             ::= ⟨statement⟩ | ⟨stmt-list⟩ ⟨statement⟩

⟨statement⟩             ::= ⟨assignment-stmt⟩ | ⟨exchange-stmt⟩ | ⟨if-stmt⟩ | ⟨loop-stmt⟩ | ⟨input-stmt⟩ |
                            ⟨output-stmt⟩

⟨assignment-stmt⟩       ::= ⟨variable⟩ ‘:=’ ⟨expression⟩ ‘;’

⟨exchange-stmt⟩         ::= ⟨variable⟩ ‘:=:’ ⟨variable⟩ ‘;’

⟨if-stmt⟩               ::= ‘if’ ⟨expression⟩ ‘then’ ⟨stmt-list⟩ ‘end’ ‘if’ ‘;’
                          | ‘if’ ⟨expression⟩ ‘then’ ⟨stmt-list⟩ ‘else’ ⟨stmt-list⟩ ‘end’ ‘if’ ‘;’

⟨loop-stmt⟩             ::= ‘while’ ⟨expression⟩ ‘loop’ ⟨stmt-list⟩ ‘end’ ‘loop’ ‘;’

⟨input-statement⟩       ::= ‘input’ ⟨variable-list⟩ ‘;’

⟨output-statement⟩      ::= ‘output’ ⟨variable-list⟩ ‘;’

⟨variable-list⟩         ::= ⟨variable⟩ | ⟨variable-list⟩ ‘,’ ⟨variable⟩

⟨expression⟩            ::= ⟨operand⟩ | ⟨operand⟩ ⟨operator⟩ ⟨operand⟩

⟨operand⟩               ::= ⟨variable⟩ | ⟨integer-literal⟩ | ⟨boolean-literal⟩ | ‘(’ ⟨expression⟩ ‘)’ | ‘not’ ⟨operand⟩

⟨variable⟩              ::= ⟨variable⟩ | ⟨variable⟩ ‘[’ ⟨expression⟩ ‘]’

⟨boolean-literal⟩       ::= ‘true’ | ‘false’

⟨operator⟩              ::= ‘<’ | ‘<=’ | ‘==’ | ‘<>’ | ‘>=’ | ‘>’ | ‘+’ | ‘-’ | ‘*’ | ‘/’ | ‘and’ | ‘or’
```

An $\langle integer - literal \rangle$ in Ledgard is simply a string of the digits 0-9, and an $\langle identifier \rangle$ is an uppercase letter followed by any number of uppercase letters, digits, and underscores. Note that the assignment operator is ‘:=’ and the equality operation is ‘==’. A solitary ‘=’ is invalid.

A comment in Ledgard begins with the symbol ‘$--$’ and continues to the end of the line.