# Regular Expression Specifications

The following table outlines the Regular Expressions (RE) used to define the lexical structure of the language.

| Token Category | Regular Expression |
|---|---|
| Keywords | `(start\|finish\|loop\|condition\|declare\|output\|input\|function\|return\|break\|continue\|else)` |
| Identifiers | `[A-Z][a-z0-9_]{0,30}` |
| Integer Literals | `[+-]?[0-9]+` |
| Floating-Point Literals | `[+-]?[0-9]+\.[0-9]{1,6}([eE][+-]?[0-9]+)?` |
| Boolean Literals | `(true\|false)` |
| String Literals | `"([^\"\\\n]\|\\["\\ntr])*"` |
| Character Literals | `'([^'\\\n]\|\\['\\ntr])'` |
| Arithmetic Operators | `(\*\*\|[\+\-\*\/%])` |
| Relational Operators | `(==\|!=\|<=\|>=\|<\|>)` |
| Logical Operators | `(&&\|\|\|!)` |
| Assignment Operators | `(\+=\|\-=\|\*=\|\/=\|=)` |
| Inc/Dec Operators | `(\+\+\|\-\-)` |
| Punctuators | `[\(\)\{\}\[\]\,;:]` |
| Single-line Comment | `##[^\n]*` |
| Multi-line Comment | `#\*([^*]\|\*+[^#])*\*+#` |
| Whitespace | `[\t\r\n]+` |