Lexical Structure

<token> ::= <ident> | <num_lit> | <string_lit> | <boolean_lit>, | <constant> | <reserved>

| <op or separator>

<ident> ::= (<letter> | **_**) (<letter> | **_** | <digit>)*

but not <reserved>, <boolean_lit>, or <constant>

<letter> ::= **a** .. **z** | **A** .. **Z**

<digit> ::= **0** | <nonzero_digit>

<nonzero_digit> ::= **1** .. **9** |

<num_lit> ::= **0** | <nonzero_digit> <digit>*

<string_lit> ::= **“** <printable_char>* **“**

<boolean_lit> := **TRUE** | **FALSE**

<constant> ::= **Z** | **BLACK** | **BLUE** | **CYAN** | **DARK_GRAY** | **GRAY** | **GREEN** | **LIGHT_GRAY** | **MAGENTA** | **ORANGE** | **PINK** | **RED** | **WHITE** | **YELLOW**

<printable_char> **::=** set of “printable” ascii characters, but not DEL. See https://www.ascii-code.com/

<reserved> ::= **image** | **pixel** | **int** | **string** | **void** | **boolean** | **write** | **height** | **width** | **if** | **fi** | **do** | **od** | **red** | **green** | **blue**

<op or separator> ::= **,** | **;** | **?** | **:** | **(** | **)** | **<** | **>** | **[** | **]** | **=** | **==** | **<=** | **>=** | **!** | **&** | **&&** | **|** | **||** |

+ | **-** | * | ** | **/** | **%** | **<:** | **:>** | **^** | **->** | **[]**

<comment> ::= **##** <printable_char> *

<white space> ::= ‘ ‘ | ‘\n’ | ‘\r’ (i.e. space, tab, newline, return)

Σ = { <printable char> | ‘\n’ | ‘\r’ }

Notes

- <white space> and <comment> separate tokens but are otherwise ignored.
- Java character notation has been used for control characters and space in the definition of <comment> and <white space>.
- Metasymbols are in normal, black font. Elements of the alphabet are blue and bold.
- Spaces in the definitions above have been included for readability—they do not translate to spaces in the token.