

EBNF terminais

| <u>Terminal</u> | <u>Expressão Regular</u> | <u>Categoria</u> |
|-----------------|---|------------------|
| 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" "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" | |
| relational_op | = ">" "<" ">=" "<=" "==" "!=" | tk_rel_op |
| id | = letter(letter digit)* | tk_id |
| const_int | = digit(digit digit)* | tk_const_int |
| const_float | = digit(.(digit)*) | tk_const_float |
| "int" | = int | tk_kw_int |
| "float" | = float | tk_kw_float |
| "char" | = char | tk_kw_char |
| "if" | = if | tk_kw_if |
| "else" | = else | tk_kw_else |
| "for" | = for | tk_kw_for |
| "while" | = while | tk_kw_while |
| "break" | = break | tk_kw_break |
| "not" | = not | tk_kw_not |
| "and" | = and | tk_kw_and |
| "or" | = or | tk_kw_or |
| "main" | = main | tk_kw_main |
| "function" | = function | tk_kw_func |
| "return" | = return | tk_kw_return |
| "strcmp" | = strcmp | tk_kw_strcmp |
| "strcop" | = strcop | tk_kw_strcop |
| "input" | = input | tk_kw_input |
| "output" | = output | tk_kw_output |
| "+" | = + | tk_op_add |
| "_" | = - | tk_op_sub |
| "*" | = * | tk_op_mul |
| "/" | = / | tk_op_div |
| "=" | = = | tk_op_atrib |
| "(" | = (| tk_open_par |
| ")" | =) | tk_close_par |
| "{" | = { | tk_open_bra |
| "}" | = } | tk_close_bra |
| " " " | = " | tk_open_quo |

“ ”

=

”

tk_close_quo