

Universidade Federal de Alagoas  
Instituto de Computação  
*Ciência da Computação*

---

## Nova - Especificação de Tokens v2

Rubens Pessoa

23 de outubro de 2016



# Sumário

<b>Sumário</b>	<b>i</b>
<b>1 Linguagem de Programação a ser utilizada na implementação</b>	<b>1</b>
<b>2 Enumeração com as categorias</b>	<b>1</b>
2.1 Enum . . . . .	1
<b>3 Expressões Regulares Auxiliares</b>	<b>2</b>
3.1 Expressões Regulares . . . . .	2



# 1 Linguagem de Programação a ser utilizada na implementação

A linguagem de programação que será utilizada para implementar os analisadores léxico e sintático será o Java, em sua versão 1.8.

## 2 Enumeração com as categorias

### 2.1 Enum

---

```
1
2 public enum TokenCategory {
3
4     EOF(-1),
5     ID(1),
6     CTE_FLOAT(2),
7     CTE_INT(3),
8     CTE_STR(4),
9     OP_ATR(5),
10    OP_REL1(6),
11    OP_REL2(7),
12    OP_AD(8),
13    OP_MULT(9),
14    OP_MOD(10),
15    COMMENT(11),
16    VECTOR_AUX(12),
17    OP_AND(13),
18    OP_OR(14),
19    OP_NOT(15),
20    PR_IF(16),
21    PR_ELSE(17),
22    PR_SHOOT(18),
23    PR_WHILE(19),
24    PR_FOR(20),
25    PR_INT(21),
26    PR_FLOAT(22),
27    PR_BOOL(23),
28    PR_STRING(24),
29    BOOL_VALUE(25),
30    SP(26),
```

```

31     AB_PAR(27),
32     FEC_PAR(28),
33     AB_CH(29),
34     FEC_CH(30),
35     AB_COL(31),
36     FEC_COL(32),
37     PR_VOID(33),
38     PR_IO(34),
39     PR_MAIN(35);
40
41
42     private int value;
43
44     TokenCategory(int value) {
45         this.value = value;
46     }
47     public int getValue() {
48         return this.value;
49     }
50 }

```

---

## 3 Expressões Regulares Auxiliares

### 3.1 Expressões Regulares

---

```

1     PR_IO = "readIn_|_printOut";
2     PR_VOID = "void";
3     PR_MAIN = "main";
4     PR_IF = "if";
5     PR_ELSE = "else";
6     PR_WHILE = "while";
7     PR_FOR = "for";
8     PR_SHOOT = "shoot";
9     PR_STRING = "string";
10    PR_INT = "int";
11    PR_FLOAT = "float";
12    PR_BOOL = "bool";
13    BOOL_VALUE = "True_|_False";
14    OP_ATR = "=";

```

```

15      OP_REL1 = "<_ | >_ | <=_ | >=" ;
16      OP_REL2 = "=_ | !=" ;
17      OP_AD = "+_ | -" ;
18      OP_MULT = "*_ | _/" ;
19      OP_MOD = "%0" ;
20      OP_AND = "and" ;
21      OP_OR = "or" ;
22      OP_NOT = "not" ;
23      SP = "_ ; _ | _ , " ;
24      AB_PAR = "(" ;
25      FEC_PAR = ")" ;
26      AB_COL = "[" ;
27      FEC_COL = "]" ;
28      AB_CH = "{" ;
29      FEC_CH = "}" ;
30      VECTOR_AUX = "::" ;
31      ID = "[a-zA-Z][_a-zA-Z0-9]*\\w*" ;
32      CTE_FLOAT = "[+|-]?([0-9]*\\.?[0-9]+)" ;
33      CTE_INT = "[0-9]+" ;
34      CTE_STR = "[a-zA-Z_]?\\\"(\\.|\\\"|\\' )*\\\" \" ;
35      COMMENT = "#[a-zA-Z][_a-zA-Z0-9]*" ;

```

---