

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

---

## Nova - Especificação de Tokens

Nelson Douglas      Rubens Pessoa

3 de agosto 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 e Lexemas</b>                 | <b>2</b> |
| 3.1 Expressões Regulares Auxiliares . . . . .                      | 2        |
| 3.2 Lexemas . . . . .  | 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 public enum Categories {
2     id ,
3     cteFloat ,
4     cteInt ,
5     cteStr ,
6     opAtr ,
7     opMeq ,
8     opMaq ,
9     opMeIgq ,
10    opMaIgq ,
11    OpIg ,
12    OpDif ,
13    opAd ,
14    opSub ,
15    opMul ,
16    opAnd ,
17    opOr ,
18    opNeg ,
19    prIf ,
20    prElse ,
21    prShoot ,
22    prWhile ,
23    prFor ,
24    prInt ,
25    prFloat ,
26    prBool ,
27    prString ,
28    prTrue ,
29    prFalse ,
30    sp ,
```

```

31         abPar ,
32         fcPar ,
33         abCh ,
34         fcCh ,
35         abCo ,
36         fcCo
37     }

```

---

## 3 Expressões Regulares Auxiliares e Lexemas

### 3.1 Expressões Regulares Auxiliares

---

```

1     letra = "a" | "b" | ... | "z" | "A" | "B" | ... | "Z" |
2     digito = "0" | "1" | ... | "9" |
3     espacoEmBranco = " "

```

---

### 3.2 Lexemas

---

```

1     id = letra (letra | digito)*
2     cteFloat = "-" ? digito+("." digito+)?
3     cteInt = "-" ? digito+
4     cteStr = "\"" (digito | letra | _espacoEmBranco)* "\""
5     opAt = "="
6     opMe = "<"
7     opMa = ">"
8     opIg = "=="
9     opDif = "!="
10    opMei = "<="
11    opMai = ">="
12    opAd = "+"
13    opSub = "-"
14    opMul = "*" | "/"
15    opAnd = "and"
16    opOr = "or"
17    opNeg = "not"
18    prIf = "if"
19    prElse = "else"

```

```
20      prShoot = "shoot"
21      prWhile = "while"
22      prFor = "for"
23      prInt = "int"
24      prChar = "char"
25      prFloat = "float"
26      prBool = "bool"
27      prString = "string"
28      prTrue = "True",
29      prFalse = "False",
30      abPar = "("
31      fcPar = ")"
32      abCh = "{"
33      fcCh = "}"
34      abCo = "["
35      fcCo = "]"
36      sp = ";" | ","
```

---