

Universidade Federal de Alagoas  
Instituto de Computação  
Compiladores

Onicla - Especificação dos tokens

Ramon Basto Callado Lima Lopes  
José Rubens da Silva Brito

Maceió  
2020.1

## **Sumário**

<b>1. Linguagem para desenvolvimento</b>	<b>3</b>
<b>2. Enumeração com as categorias de Tokens</b>	<b>3</b>
<b>3. Expressões Regulares Auxiliares</b>	<b>4</b>
<b>4. Expressões Regulares dos Lexemas</b>	<b>4</b>

## 1. Linguagem para desenvolvimento

A linguagem de programação Onicla possui os seus analisadores léxico e sintático desenvolvidos na linguagem de programação Java. Onde o Java Development Kit na versão 11.0.10.

## 2. Enumeração com as categorias de Tokens

```
public enum TokenCategory {  
    EOF,  
    ID,  
    CTE_FLOAT,  
    CTE_INT,  
    CTE_CHAR,  
    CTE_CADCHA,  
    CTE_BOOL,  
    OP_ATR,  
    OP_REL,  
    OP_AD,  
    OP_SUB,  
    OP_MULT,  
    OP_DIV,  
    OP_RES,  
    OP_CONCAT,  
    OP_GREATER,  
    OP_LESS,  
    OP_GREATEQ,  
    OP_LESSEQ,  
    OP_NOT,  
    OP_NOTUN,  
    PR_AND,  
    PR_OR,  
    PR_FUNC,  
    PR_REFOUND,  
    PR_IF,  
    PR_ELSE,  
    PR_WHILE,  
    PR_REPEAT,  
    PR_INTEGER,  
    PR_FLOAT,  
    PR_CHARRAY,  
    PR_CHARAC,  
    PR_BOOL,
```

```
PR_INPUT,  
PR_PRINT,  
PR_PRINTNL,  
PR_PRINTL,  
PR_TRUE,  
PR_FALSE,  
PR_NULL,  
PR_BEGIN,  
PR_END,  
PR_VOID,  
PR_MAIN,  
AB_PAR,  
FEC_PAR,  
AB_COL,  
FEC_COL,  
TERMINAL,  
SEP,  
ER_ID,  
ER_CHAR,  
ER_NUMBER,  
ER_KEYWORD,  
ER_SYMBOL;  
  
}
```

### 3. Expressões Regulares Auxiliares

letras\_minusculas = '[:lower:]'

letras\_maiusculas = '[:upper:]'

digitos = '[:digit:]'

alfanumerico = '[:alnum:]'

simbolo = '[:punct:][-][\']'

ans = ' [{alfanumerico}{simbolo}]'

cadCarac = '\ {ans}+[:space:]+[:word:]+\''

#### 4. Expressões Regulares dos Lexemas

Final do arquivo	
EOF	<<EOF>>

Identificadores	
ID	[[letras_miniscula]{alfanumerico}+{1,15}]

Palavras reservadas	
<i>PR_VOID</i>	'Void'
<i>PR_MAIN</i>	'Main'
<i>PR_IF</i>	'If'
<i>PR_ELSE</i>	'Else'
<i>PR_WHILE</i>	'While'
<i>PR_REPEAT</i>	'Repeat'
<i>PR_FLOAT</i>	'Float'
<i>CTE_FLOAT</i>	[[digitos]+'.'{digitos}+]
<i>PR_INT</i>	'Integer'
<i>CTE_INT</i>	[[digitos}+]
<i>PR_CHARRAY</i>	'Characterarray'

<i>CTE_CADCHA</i>	[[cadCarac]]
<i>CTE_CHAR</i>	[\{ans}{1}\]
<i>PR_CHARAC</i>	‘Character’
<i>PR_BEGIN</i>	‘Begin’
<i>PR_END</i>	‘End’
<i>PR_INPUT</i>	‘Input’
<i>PR_PRINT</i>	‘Print’
<i>PR_PRINTNL</i>	‘Printnl’
<i>PR_PRINTL</i>	‘Printl’
<i>PR_BOOL</i>	‘Bool’
<i>CTE_BOOL</i>	‘True’   ‘False’
<i>PR_TRUE</i>	‘True’
<i>PR_FALSE</i>	‘False’
<i>PR_NULL</i>	‘Null’
<i>PR_AND</i>	‘And’
<i>PR_OR</i>	‘Or’

<b>Operadores</b>	
<i>OP_ATR</i>	‘=’
<i>OP_REL</i>	‘==’, ‘!=’
<i>OP_AD</i>	‘+’
<i>OP_SUB</i>	‘_’

OP_MULT	'*
OP_DIV	'/'
OP_RES	'%'
OP_GREATER	'>'
OP_LESS	'<'
OP_GREATEQ	'<='
OP_LESSEQ	'>='
OP_NOT	'!'
OP_NOTUN	'~'
OP_CONCAT	'^'

Delimitadores	
AB_PAR	'('
FEC_PAR	')'
AB_COL	'['
FEC_COL	']'
TERMINAL	'.,'
SEP	' ,'

Erros Léxicos	
<i>ER_ID</i>	Identificador mal formado
<i>ER_CHAR</i>	Caracter ou cadeia mal formado
<i>ER_NUMBER</i>	Número mal formado
<i>ER_KEYWORD,</i>	Palavra chave mal formada
<i>ER_SYMBOL</i>	Símbolo desconhecido