

Lucas Peixoto de Almeida Cavalcante

## **Especificação da gramática da linguagem de programação LisC**

Especificação da gramática da linguagem de programação LisC, definida pelo aluno, para a disciplina de Compiladores, correspondente à parte da avaliação da AB2 do semestre de 2019.1, sob orientação do **Prof. Alcino Dall Igna Jr.**

Universidade Federal de Alagoas

Instituto de Computação

Brasil

Maceió/Al  
2019.1

# Sumário

<b>1</b>	<b>INTRODUÇÃO . . . . .</b>	<b>3</b>
<b>2</b>	<b>ALTERAÇÃO NOS TOKENS . . . . .</b>	<b>4</b>
<b>3</b>	<b>GRAMÁTICA ORIGINAL . . . . .</b>	<b>5</b>
<b>4</b>	<b>GRAMÁTICA LL(1) . . . . .</b>	<b>8</b>

# 1 Introdução

Esse documento descreve a gramática construída da linguagem de programação LisC, que deverá ser respeitada na implementação do analisador sintático em C. O analisador sintático escolhido é o preditivo recursivo. Diante da escolha deste analisador sintático, teremos também nesse documento as alterações necessárias na gramática para que a gramática se torne uma  $LL(1)$ , pré-requisito obrigatório de todo analisador sintático preditivo recursivo.

## 2 Alteração nos tokens

A fim de diminuir e simplificar a gramática, foi decidido agrupar alguns tokens, gerando assim novos tokens e eliminando os antigos. O agrupamento pode ser melhor visualizado na **tabela 1**.

Novo token	Relação com tokens eliminados	Expressão regular
OP_ADI	OP_PLUS   OP_MINUS	'-'   '+'
OP_RELACIONAL	OP_EQUAL   OP_DIFF   OP_LEQ   OP_LESS   OP_GEQ   OP_GREATER	'=='   '!='   '<='   '<'   '>='   '>'
OP_LOGIC	OP_OR   OP_AND	'&&'   '  '
OP_MULT_DIV	OP_MULT   OP_DIVIDE	'*'   '/'
OP_FORMAT	FORMAT_FIELD   FORMAT_DECIMAL	'%'   '%%'

Tabela 1 – Agrupamento para a criação de novos tokens.

## 3 Gramática Original

O código 4.1 apresenta a gramática original da linguagem.

```

1 Programa = ListaDecl
2
3 ListaDecl = ListaDecl Decl
4 ListaDecl = Decl
5
6 Decl = DeclVariaveis
7 Decl = DeclFuncao
8
9 DeclVariaveis = DefTipo ListaVariaveis ';'
10
11 DefTipo = 'defint'
12 DefTipo = 'defbool'
13 DefTipo = 'defchar'
14 DefTipo = 'defstring'
15 DefTipo = 'deffloat'
16
17 ListaVariaveis = ListaVariaveis ',' Variavel
18 ListaVariaveis = Variavel
19
20 Variavel = 'id'
21 Variavel = 'id' ExpConcat
22 Variavel = 'id' '[' ExpConcat ']'
23 Variavel = 'id' '[' ExpConcat ']' '{' ListaUni '}'
24
25 ListaUni = ListaUni ',' ExpConcat
26 ListaUni = ExpConcat
27
28 DeclFuncao = 'defun' 'id' '(' ListaParam ')' '(' TipoReturn ')' '{'
    ListaSent '}'
29
30 ListaParam = ListaParam ',' Param
31 ListaParam = Param
32 ListaParam = 'void'
33
34 Param = Tipo 'id'
35 Param = Tipo 'id' '[' ']'
36
37 TipoReturn = 'void'
38 TipoReturn = Tipo
39
40 ListaSent = ListaSent Sent

```

```

41 ListaSent = Sent
42
43 Sent = DeclVariaveis
44 Sent = Comando
45
46 Comando = Continue
47 Comando = Break
48 Comando = Atribuicao
49 Comando = ChamadaFunc
50 Comando = Return
51 Comando = Read
52 Comando = Echo
53 Comando = If
54 Comando = For
55 Comando = While
56
57 Continue = 'continue' ';'
58
59 Break = 'break' ';'
60
61 Atribuicao = 'set' 'id' ExpConcat ';'
62 Atribuicao = 'set' 'id' '[' ExpConcat ']' ExpConcat ';'
63
64 ChamadaFunc = 'id' '(' ListaArg ')' ';'
65
66 Return = 'return' ExpConcat ';'
67
68 Read = 'read' '(' ListaArg ')' ';'
69
70 Echo = 'echo' '(' ExpConcat ')' ';'
71 Echo = 'echo' '(' ExpConcat ')' '(' ListaArg ')' ';'
72
73 If = 'if' '(' ExpBooleana ')' '{' ListaSent '}' Else
74
75 Else = 'else' '{' ListaSent '}'
76 Else = epsilon
77
78 For = 'for' '(' 'id' ',' ExpAritmetica ',' ExpAritmetica ForPasso ')' '{'
    ListaSent '}'
79
80 While = 'while' '(' ExpBooleana ')' '{' ListaSent '}'
81
82 ForPasso = ',' ExpAritmetica
83 ForPasso = epsilon
84
85 ListaArg = ListaArg ',' 'id'
86 ListaArg = 'id'

```

```
87
88 Tipo = 'int'
89 Tipo = 'char'
90 Tipo = 'string'
91 Tipo = 'float'
92 Tipo = 'bool'
93
94 ExpAritmetica = ExpAritmetica 'op_adi' TermAritmetico
95 ExpAritmetica = TermAritmetico
96
97 ExpBooleana = ExpBooleana 'op_logic' TermBooleano
98 ExpBooleana = TermBooleano
99
100 ExpConcat = ExpConcat '<<' ExpAritmetica
101 ExpConcat = ExpBooleana
102
103 TermBooleano = '!' TermBooleano
104 TermBooleano = ExpAritmetica 'op_relacional' ExpAritmetica
105 TermBooleano = ExpAritmetica
106
107 TermAritmetico = TermAritmetico 'op_mult_div' TermFormatacao
108 TermAritmetico = TermFormatacao
109
110 TermFormatacao = TermFormatacao 'op_format' FatorAritmetico
111 TermFormatacao = FatorAritmetico
112
113 FatorAritmetico = '-' FatorAritmetico
114 FatorAritmetico = 'const_float'
115 FatorAritmetico = 'const_char'
116 FatorAritmetico = 'const_bool'
117 FatorAritmetico = 'const_int'
118 FatorAritmetico = 'const_string'
119 FatorAritmetico = '(' ExpBooleana ')',
120 FatorAritmetico = 'id'
121 FatorAritmetico = 'id' '(' ListaArg ')',
122 FatorAritmetico = 'id' '[' ExpAritmetica ']'
```

Código 3.1 – Gramática original da linguagem de programação LisC.

## 4 Gramática LL(1)

```
1 Programa = ListaDecl
2
3 ListaDecl = Decl ListaDeclR
4
5 ListaDeclR = Decl ListaDeclR
6 ListaDeclR = epsilon
7
8 Decl = DeclVariaveis
9 Decl = DeclFuncao
10
11 DeclVariaveis = DefTipo ListaVariaveis ';'
12
13 DefTipo = 'defint'
14 DefTipo = 'defbool'
15 DefTipo = 'defchar'
16 DefTipo = 'defstring'
17 DefTipo = 'deffloat'
18
19 ListaVariaveis = Variavel ListaVariaveisR
20
21 ListaVariaveisR = ',' Variavel ListaVariaveisR
22 ListaVariaveisR = epsilon
23
24 Variavel = 'id' VariavelF
25
26 VariavelF = ExpConcat
27 VariavelF = epsilon
28 VariavelF = '[' ExpConcat ']' VariavelFF
29
30 VariavelFF = '{' ListaUni '}'
31 VariavelFF = epsilon
32
33 ListaUni = ExpConcat ListaUniR
34
35 ListaUniR = ',' ExpConcat ListaUniR
36 ListaUniR = epsilon
37
38 DeclFuncao = 'defun' 'id' '(' ListaParam ')' '(' TipoReturn ')' '{'
    ListaSent '}'
39
40 ListaParam = Param ListaParamR
41 ListaParam = 'void'
42
```



```
43 ListaParamR = ',' Param ListaParamR
44 ListaParamR = epsilon
45
46 Param = Tipo 'id' ParamF
47
48 ParamF = '[' ' ' ]'
49 ParamF = epsilon
50
51 TipoReturn = 'void'
52 TipoReturn = Tipo
53
54 ListaSent = Sent ListaSentR
55
56 ListaSentR = Sent ListaSentR
57 ListaSentR = epsilon
58
59 Sent = DeclVariaveis
60 Sent = Comando
61
62 Comando = Continue
63 Comando = Break
64 Comando = Atribuicao
65 Comando = ChamadaFunc
66 Comando = Return
67 Comando = Read
68 Comando = Echo
69 Comando = If
70 Comando = For
71 Comando = While
72
73 Continue = 'continue' ';'
74
75 Break = 'break' ';'
76
77 Atribuicao = 'set' 'id' AtribuicaoF
78
79 AtribuicaoF = ExpConcat ';'
80 AtribuicaoF = '[' ExpConcat ']' ExpConcat ';'
81
82 ChamadaFunc = 'id' '(' ListaArg ')' ';'
83
84 Return = 'return' ExpConcat ';'
85
86 Read = 'read' '(' ListaArg ')' ';'
87
88 Echo = 'echo' '(' ExpConcat ')' EchoF
89
```

```

90 EchoF = '(' ListaArg ')' ';'
91 EchoF = ';'
92
93 If = 'if' '(' ExpBooleana ')' '{' ListaSent '}' Else
94
95 Else = 'else' '{' ListaSent '}'
96 Else = epsilon
97
98 For = 'for' '(' 'id' ',' ExpAritmetica ',' ExpAritmetica ForPasso ')' '{'
    ListaSent '}'
99
100 While = 'while' '(' ExpBooleana ')' '{' ListaSent '}'
101
102 ForPasso = ',' ExpAritmetica
103 ForPasso = epsilon
104
105 ListaArg = 'id' ListaArgR
106
107 ListaArgR = ',' 'id' ListaArgR
108 ListaArgR = epsilon
109
110 Tipo = 'int'
111 Tipo = 'char'
112 Tipo = 'string'
113 Tipo = 'float'
114 Tipo = 'bool'
115
116 ExpAritmetica = TermAritmetico ExpAritmeticaR
117
118 ExpAritmeticaR = 'op_adi' TermAritmetico ExpAritmeticaR
119 ExpAritmeticaR = epsilon
120
121 ExpBooleana = TermBooleano ExpBooleanaR
122
123 ExpBooleanaR = 'op_logic' TermBooleano ExpBooleanaR
124 ExpBooleanaR = epsilon
125
126 ExpConcat = ExpBooleana ExpConcatR
127
128 ExpConcatR = '<<' ExpAritmetica ExpConcatR
129 ExpConcatR = epsilon
130
131 TermBooleano = '!' TermBooleano
132 TermBooleano = ExpAritmetica TermBooleanoF
133
134 TermBooleanoF = 'op_relacional' ExpAritmetica
135 TermBooleanoF = epsilon

```

```
136
137 TermAritmetico = TermFormatacao TermAritmeticoR
138
139 TermAritmeticoR = 'op_mult_div' TermFormatacao TermFormatacaoR
140 TermAritmeticoR = epsilon
141
142 TermFormatacao = FatorAritmetico TermFormatacaoR
143
144 TermFormatacaoR = 'op_format' FatorAritmetico TermFormatacaoR
145 TermFormatacaoR = epsilon
146
147 FatorAritmetico = '-' FatorAritmetico
148 FatorAritmetico = 'const_float'
149 FatorAritmetico = 'const_char'
150 FatorAritmetico = 'const_bool'
151 FatorAritmetico = 'const_int'
152 FatorAritmetico = 'const_string'
153 FatorAritmetico = '(' ExpBooleana ') '
154 FatorAritmetico = 'id' FatorAritmeticoF
155
156 FatorAritmeticoF = '(' ListaArg ') '
157 FatorAritmeticoF = '[' ExpAritmetica ']'
158 FatorAritmeticoF = epsilon
```

Código 4.1 – Gramática original da linguagem de programação LisC.