**Universidade Federal de Santa Catarina**
Campus Araranguá

**Disciplina:** Construção de Compiladores

---

GRAMÁTICA

1) BLOCO::= void main { DCLVAR  DCLFUNC CORPO }
2) DCLVAR::= nomevariavel REPIDENT : TIPO ; LDVAR
3) DCLVAR::= î
4) REPIDENT::= î
5) REPIDENT::= , nomevariavel REPIDENT
6) TIPO::= integer
7) TIPO::= float
8) TIPO::= string
9) TIPO::= char
10) LDVAR::= î
11) LDVAR::= LID : TIPO ; LDVAR
12) LID::= nomevariavel REPIDENT
13) DCLFUNC::= TIPO_RETORNO nomevariavel DEFPAR{ DCLVAR  DCLFUNC CORPO return (VALORRETORNO)} DCLFUNC
14) TIPO_RETORNO::= integer
15) TIPO_RETORNO::= void
16) TIPO_RETORNO::= char
17) TIPO_RETORNO::= float
18) TIPO_RETORNO::= string
19) DCLFUNC::= î
20) VALORRETORNO::= numerointeiro
21) VALORRETORNO::= numerofloat
22) VALORRETORNO::= nomedavariavel
23) VALORRETORNO::= nomedochar
24) VALORRETORNO::= nomedastring
25) VALORRETORNO::= î
26) DEFPAR::= î
27) DEFPAR::= ( PARAM )
28) PARAM::= TIPO  LPARAM
29) LPARAM::= ; TIPO  LPARAM
30) LPARAM::= î
31) CORPO::= inicio COMANDO ; REPCOMANDO fim
32) REPCOMANDO ::= î
33) REPCOMANDO::= COMANDO ; REPCOMANDO
34) COMANDO::= nomevariavel = EXPRESSAO
35) COMANDO::= nomedastring = EXPRESSAO
36) COMANDO::= nomedochar = EXPRESSAO

37) COMANDO::= EXPRESSAO = EXPRESSAO
38) COMANDO::= î
39) COMANDO::= callfuncao nomevariavel PARAMETROS
40) PARAMETROS::= î
41) PARAMETROS::= ( TPARAM REPPAR )
42) REPPAR::= î
43) REPPAR::= , TPARAM REPPAR
44) TPARAM::= numerointeiro
45) TPARAM::= nomedastring
46) TPARAM::= numerofloat
47) TPARAM::= nomedochar
48) TPARAM::= nomevariavel
49) COMANDO::= if (nomevariavel COMPARACAO) { COMANDO ; REPCOMANDO } ELSEPARTE
50) ELSEPARTE::= else { COMANDO ; REPCOMANDO }
51) ELSEPARTE::= î
52) COMANDO::= while ( nomevariavel COMPARACAO) { COMANDO ; REPCOMANDO }
53) COMPARACAO::= == CONTCOMPARACAO
54) COMPARACAO::= != CONTCOMPARACAO
55) COMPARACAO::= > CONTCOMPARACAO
56) COMPARACAO::= >= CONTCOMPARACAO
57) COMPARACAO::= < CONTCOMPARACAO
58) COMPARACAO::= <= CONTCOMPARACAO
59) CONTCOMPARACAO::= numerointeiro
60) CONTCOMPARACAO::= numerofloat
61) CONTCOMPARACAO::= nomedastring
62) CONTCOMPARACAO::= nomedochar
63) CONTCOMPARACAO::= nomevariavel
64) COMANDO::= for (nomevariavel = CONTCOMPARACAO; nomevariavel COMPARACAO; INCREMENTO) { COMANDO ; REPCOMANDO }
65) INCREMENTO::= ++ numerointeiro
66) INCREMENTO::= -- numerointeiro
67) COMANDO::= do { COMANDO ; REPCOMANDO } while ( nomevariavel COMPARACAO)
68) COMANDO::= cin >>  nomevariavel
69) COMANDO::= cout <<  literal SEQCOUT
70) SEQCOUT::= î
71) SEQCOUT::= << nomevariavel SEQUENCIA SEQCOUT
72) SEQCOUT::= << literal SEQCOUT
73) SEQUENCIA::= î
74) SEQUENCIA::= , nomevariavel SEQUENCIA
75) EXPRESSAO::= TERMO REPEXP
76) EXPRESSAO::= callfuncao nomevariavel PARAMETROS
77) REPEXP::= + TERMO REPEXP
78) REPEXP::= - TERMO REPEXP
79) REPEXP::= î
80) TERMO::= FATOR REPTERMO
81) REPTERMO::= î
82) REPTERMO::= * FATOR REPTERMO
83) REPTERMO::= / FATOR REPTERMO
84) FATOR::= numerointeiro
85) FATOR::= numerofloat

86) FATOR::= nomevariavel
87) FATOR::= nomedastring
88) FATOR::= nomedochar
89) FATOR::= ( EXPRESSAO )

| Cód. | Token |
|---|---|
| 1 | while |
| 2 | void |
| 3 | string |
| 4 | return |
| 5 | numerointeiro |
| 6 | numerofloat |
| 7 | nomevariavel |
| 8 | nomedochar |
| 9 | nomedavariavel |
| 10 | nomedastring |
| 11 | main |
| 12 | literal |
| 13 | integer |
| 14 | int |
| 15 | inicio |
| 16 | if |
| 17 | î |
| 18 | for |
| 19 | float |
| 20 | fim |
| 21 | else |
| 22 | double |
| 23 | do |
| 24 | cout |
| 25 | cin |
| 26 | char |
| 27 | callfuncao |
| 28 | >> |
| 29 | >= |
| 30 | > |
| 31 | == |
| 32 | = |
| 33 | <= |
| 34 | << |
| 35 | < |
| 36 | ++ |
| 37 | + |
| 38 | } |
| 39 | { |
| 40 | ; |
| 41 | : |

| | |
|---|---|
| 42 | / |
| 43 | , |
| 44 | * |
| 45 | ) |
| 46 | ( |
| 47 | $ |
| 48 | != |
| 49 | -- |
| 50 | - |

| Cód | Símbolo |
|---|---|
| 51 | BLOCO |
| 52 | DCLVAR |
| 53 | DCLFUNC |
| 54 | CORPO |
| 55 | REPIDENT |
| 56 | TIPO |
| 57 | LDVAR |
| 58 | REPIDEN |
| 59 | LID |
| 60 | TIPO_RETORNO |
| 61 | DEFPAR |
| 62 | VALORRETORNO |
| 63 | PARAM |
| 64 | LPARAM |
| 65 | COMANDO |
| 66 | REPCOMANDO |
| 67 | EXPRESSAO |
| 68 | PARAMETROS |
| 69 | TPARAM |
| 70 | REPPAR |
| 71 | COMPARACAO |
| 72 | ELSEPARTE |
| 73 | CONTCOMPARACAO |
| 74 | INCREMENTO |
| 75 | SEQCOUT |
| 76 | SEQUENCIA |
| 77 | EXPSIMP |
| 78 | REPEXPSIMP |
| 79 | TERMO |
| 80 | REPEXP |
| 81 | FATOR |
| 82 | REPTERMO |

**Tabela de Parsing**

```
tabParsing[51][2] = 1;
tabParsing[52][2] = 3;
tabParsing[52][3] = 3;
tabParsing[52][7] = 2;
tabParsing[52][13] = 3;
tabParsing[52][15] = 3;
tabParsing[52][19] = 3;
tabParsing[52][26] = 3;
tabParsing[52][47] = 3;
tabParsing[53][2] = 13;
tabParsing[53][3] = 13;
tabParsing[53][13] = 13;
tabParsing[53][15] = 19;
tabParsing[53][19] = 13;
tabParsing[53][26] = 13;
tabParsing[54][15] = 31;
tabParsing[55][41] = 4;
tabParsing[55][43] = 5;
tabParsing[56][3] = 8;
tabParsing[56][13] = 6;
tabParsing[56][19] = 7;
tabParsing[56][26] = 9;
tabParsing[57][2] = 10;
tabParsing[57][3] = 10;
tabParsing[57][7] = 11;
tabParsing[57][13] = 10;
tabParsing[57][15] = 10;
tabParsing[57][19] = 10;
tabParsing[57][26] = 10;
tabParsing[57][47] = 10;
tabParsing[59][7] = 12;
tabParsing[60][2] = 15;
tabParsing[60][3] = 18;
tabParsing[60][13] = 14;
tabParsing[60][19] = 17;
tabParsing[60][26] = 16;
tabParsing[61][39] = 26;
tabParsing[61][46] = 27;
tabParsing[62][3] = 24;
tabParsing[62][5] = 20;
tabParsing[62][6] = 21;
tabParsing[62][8] = 23;
tabParsing[62][9] = 22;
tabParsing[62][45] = 25;
tabParsing[63][3] = 28;
tabParsing[63][13] = 28;
tabParsing[63][19] = 28;
tabParsing[63][26] = 28;
```

```
tabParsing[64][40] = 29;
tabParsing[64][45] = 30;
tabParsing[65][1] = 52;
tabParsing[65][5] = 37;
tabParsing[65][6] = 37;
tabParsing[65][7] = 37;
tabParsing[65][8] = 37;
tabParsing[65][10] = 37;
tabParsing[65][16] = 49;
tabParsing[65][18] = 64;
tabParsing[65][23] = 67;
tabParsing[65][24] = 69;
tabParsing[65][25] = 68;
tabParsing[65][27] = 39;
tabParsing[65][40] = 38;
tabParsing[65][46] = 37;
tabParsing[66][20] = 33;
tabParsing[66][38] = 33;
tabParsing[67][5] = 75;
tabParsing[67][6] = 75;
tabParsing[67][7] = 75;
tabParsing[67][8] = 75;
tabParsing[67][10] = 75;
tabParsing[67][27] = 76;
tabParsing[67][46] = 75;
tabParsing[68][32] = 40;
tabParsing[68][40] = 40;
tabParsing[68][45] = 40;
tabParsing[68][46] = 41;
tabParsing[69][5] = 44;
tabParsing[69][6] = 46;
tabParsing[69][7] = 48;
tabParsing[69][8] = 47;
tabParsing[69][10] = 45;
tabParsing[70][43] = 43;
tabParsing[70][45] = 42;
tabParsing[71][29] = 56;
tabParsing[71][30] = 55;
tabParsing[71][31] = 53;
tabParsing[71][33] = 58;
tabParsing[71][34] = 57;
tabParsing[71][48] = 54;
tabParsing[72][21] = 50;
tabParsing[72][40] = 51;
tabParsing[73][5] = 59;
tabParsing[73][6] = 60;
tabParsing[73][7] = 63;
tabParsing[73][8] = 62;
tabParsing[73][10] = 61;
tabParsing[74][36] = 65;
```

```
tabParsing[74][49] = 66;
tabParsing[75][34] = 72;
tabParsing[75][40] = 70;
tabParsing[76][34] = 73;
tabParsing[76][40] = 73;
tabParsing[76][43] = 74;
tabParsing[79][5] = 80;
tabParsing[79][6] = 80;
tabParsing[79][7] = 80;
tabParsing[79][8] = 80;
tabParsing[79][10] = 80;
tabParsing[79][46] = 80;
tabParsing[80][32] = 79;
tabParsing[80][37] = 77;
tabParsing[80][40] = 79;
tabParsing[80][45] = 79;
tabParsing[80][50] = 78;
tabParsing[81][5] = 84;
tabParsing[81][6] = 85;
tabParsing[81][7] = 86;
tabParsing[81][8] = 88;
tabParsing[81][10] = 87;
tabParsing[81][46] = 89;
tabParsing[82][32] = 81;
tabParsing[82][37] = 81;
tabParsing[82][40] = 81;
tabParsing[82][42] = 83;
tabParsing[82][44] = 82;
tabParsing[82][45] = 81;
tabParsing[82][50] = 81;
```