

# **Relatório do Laboratório de CES-41 Compiladores**

14 de março de 2018

**Disciplina: CES-41**

Estudante: Felipe Guimarães

Turma 19.3

**Instituto Tecnológico de Aeronáutica**



\* Os testes foram criados pelo aluno Felipe Uchida.

## I. Questão 1

### Código:

```
1  %{
2  #define ACEITA  1
3  #define OUTRA   2
4  %}
5  delim      [ \t\n\r]
6  ws         {delim}+
7  aceita     0*(10*1)*0*|1*0(1*01*0)*1*
8  string     [^ \t\n\r]+
9  %%
10 {ws}       { ;}
11 {aceita}    {return ACEITA;}
12 {string}    {return OUTRA;}
13 %%
14 main () {
15     int i;
16     while (i = yylex ())
17         switch (i) {
18             case ACEITA:
19                 printf ("%20s: Aceita\n", yytext);
20                 break;
21             case OUTRA:
22                 printf ("%20s: Rejeitada\n", yytext);
23                 break;
24         }
25 }
26
```

### Entrada:

```
1 111 001 00101 11 00 100100 100100001 21 000001 00110001 011100bbb01100 aaa
2 01010111 0110aa011 11a00 010101 asdff 0 1
```

### Saída:

```
1 111 : Rejeitada
2 001 : Rejeitada
3 00101 : Aceita
4 11 : Aceita
5 00 : Aceita
6 100100 : Aceita
7 100100001 : Rejeitada
8 21 : Rejeitada
9 000001 : Aceita
10 00110001 : Aceita
11 011100bbb01100 : Rejeitada
12 aaa : Rejeitada
13 01010111 : Aceita
14 0110aa011 : Rejeitada
15 11a00 : Rejeitada
16 010101 : Aceita
17 asdff : Rejeitada
18 0 : Aceita
19 1 : Rejeitada
20
```

## II. Questão 2

Código:

```
1  %{
2  #define ACEITA 1
3  #define REJEITA 2
4  #define OUTRA 3
5  %}
6  delim      [ \t\n\r]
7  ws         {delim}+
8  rejeita 1*(1*01*0)*1*|0*1(0*10*1)*0*
9  aceita (0*1)*
10 string  [^ \t\n\r]+
11 %%
12 {ws}      { ;}
13 {rejeita} {return REJEITA;}
14 {aceita}  {return ACEITA;}
15 {string}  {return OUTRA;}
16 %%
17 main () {
18     int i;
19     while (i = yylex ())
20         switch (i) {
21             case REJEITA:
22                 printf ("%20s: Rejeitada\n", yytext);
23                 break;
24             case ACEITA:
25                 printf ("%20s: Aceita\n", yytext);
26                 break;
27             case OUTRA:
28                 printf ("%20s: Rejeitada\n", yytext);
29                 break;
30         }
31     }
32 }
```

Entrada:

```
1 111 001 00101 11 00 100100 100100001 21 000001 00110001 011100bbb01100 aaa
2 01010111 0110aa011 11a00 010101 asdff 0 1 00011 0011 01 0000110 0101010101110
```

Saída:

```
1 111 : Rejeitada
2 001 : Rejeitada
3 00101 : Aceita
4 11 : Rejeitada
5 00 : Rejeitada
6 100100 : Rejeitada
7 100100001 : Rejeitada
8 21 : Rejeitada
9 000001 : Rejeitada
10 00110001 : Rejeitada
11 011100bbb01100 : Rejeitada
12 aaa : Rejeitada
13 01010111 : Rejeitada
14 0110aa011 : Rejeitada
15 11a00 : Rejeitada
16 010101 : Rejeitada
17 asdff : Rejeitada
18 0 : Aceita
19 1 : Rejeitada
20 00011 : Aceita
21 0011 : Rejeitada
22 01 : Rejeitada
23 0000110 : Aceita
24 0101010101110 : Aceita
25
```

### III. Questão 3

Código:

```
1  %{
2  #define ACEITA 1
3  #define REJEITA 2
4  #define OUTRA 3
5  %}
6  delim      [ \t\n\r]
7  ws         {delim}+
8  aceita (0*1)*
9  string    [^ \t\n\r]+
10 %%
11 {ws}      { ;}
12 {aceita}   {return ACEITA;}
13 {string}   {return OUTRA;}
14 %%
15 int check(char* str){
16     int uns = 0;
17     int ind = 0;
18     while(ind < 4){
19         if(str[ind] == '1'){
20             uns++;
21         } else if(str[ind] == '\0'){ //Tamanho menor que 4
22             return 0;
23         }
24         ind++;
25     }
26     if(uns < 3) return 0;
27     while(str[ind] != '\0'){
28         if(str[ind-4] == '1') uns--;
29         if(str[ind] == '1') uns++;
30         if(uns < 3) return 0;
31         ind++;
32     }
33     return 1;
34 }
35
36 main () {
37     int i;
38     while (i = yylex ())
39         switch (i) {
40             case ACEITA:
41                 if(check(yytext))
42                     printf ("%20s: Aceita\n", yytext);
43                 else
44                     printf ("%20s: Rejeitada\n", yytext);
45                 break;
46             case OUTRA:
47                 printf ("%20s: Rejeitada\n", yytext);
48                 break;
49         }
50 }
51
```

Entrada:

```
1  0111 1011 1101 1110 1111
2  01101 11011 1101 111011 11101
3  aa0011 b111 c1111 1111q
```

## Saída:

```
1 0111 : Aceita
2 1011 : Aceita
3 1101 : Aceita
4 1110 : Aceita
5 1111 : Aceita
6 01101 : Rejeitada
7 11011 : Aceita
8 1101 : Aceita
9 111011 : Aceita
10 11101 : Aceita
11 aa0011 : Rejeitada
12 b111 : Rejeitada
13 c1111 : Rejeitada
14 1111q : Rejeitada
15
```

## IV. Questão 4

### Código:

```
1 %{
2 #include <string.h>
3 #define ID 1
4 #define CTINT 2
5 #define CTHEX 3
6 #define CTREAL 4
7 #define OPREL 5
8 #define MENOR 1
9 #define MENIG 2
10 #define MAIOR 3
11 #define MAIG 4
12 #define IGUAL 5
13 #define DIFER 6
14 #define OPAD 6
15 #define MAIS 1
16 #define MENOS 2
17 #define OPMULT 7
18 #define VEZES 1
19 #define DIV 2
20 #define ABPAR 8
21 #define FPAR 9
22 #define INVAL 10
23
24 union {
25     char string[50];
26     int atr, valor;
27     float valfloat;
28     char carac;
29 } yylval;
30
31 %}
32
33 delim [ \t\n\r]
34 ws {delim}+
35 digito [0-9]
36 ctint {digito}+
37 cthex 0x{digito}+
38 ctreal {digito}+\.{digito}*
39 l_maiuscula [A-Z]
40 l_minuscula [a-z]
41 id {l_maiuscula}({l_minuscula}|{digito}|(_{l_maiuscula}))*({l_minuscula}|{digito})+)?
```

```

43 %%
44 {ws}      {;}
45 {id}      {strcpy (yylval.string, yytext); return ID;}
46 {ctint}   {yylval.valor = atoi(yytext); return CTINT;}
47 {cthex}   {yylval.valor = strtol(yytext, NULL, 16); return CTHEX;}
48 {ctreal}  {yylval.valfloat = strtod(yytext, NULL); return CTREAL;}
49 "<"      {strcpy(yylval.string, "MENOR"); return OPREL;}
50 "<="     {strcpy(yylval.string, "MENIG"); return OPREL;}
51 ">"      {strcpy(yylval.string, "MAIOR"); return OPREL;}
52 ">="     {strcpy(yylval.string, "MAIG"); return OPREL;}
53 "=="      {strcpy(yylval.string, "IGUAL"); return OPREL;}
54 "!="      {strcpy(yylval.string, "DIF"); return OPREL;}
55 "+"       {strcpy(yylval.string, "MAIS"); return OPAD;}
56 "-"       {strcpy(yylval.string, "MENOS"); return OPAD;}
57 "*"       {strcpy(yylval.string, "VEZES"); return OPMULT;}
58 "/"       {strcpy(yylval.string, "DIV"); return OPMULT;}
59 "("       {return ABPAR;}
60 ")"       {return FPAR;}
61 "."       {yylval.carac = yytext[0]; return INVALID;}
62 %%
63
64 main () {
65     int i;
66     printf ("\n  texto |  tipo  | atributo \n");
67     printf ("-----\n");
68     while (i = yylex ()) {
69         printf ("%10s|", yytext);
70         switch (i) {
71             case ID:
72                 printf ("%10s|%10s|", "ID", yylval.string); break;
73             case CTINT:
74                 printf ("%10s|%10d|", "CTINT", yylval.valor); break;
75             case CTHEX:
76                 printf ("%10s|%10d|", "CTHEX", yylval.valor); break;
77             case CTREAL:
78                 printf ("%10s|%10f|", "CTREAL", yylval.valfloat); break;
79             case OPREL:
80                 printf ("%10s|%10s|", "OPREL", yylval.string); break;
81             case OPAD:
82                 printf ("%10s|%10s|", "OPAD", yylval.string); break;
83             case OPMULT:
84                 printf ("%10s|%10s|", "OPMULT", yylval.string); break;
85             case ABPAR:
86                 printf ("%10s|", "ABPAR"); break;
87             case FPAR:
88                 printf ("%10s|", "FPAR"); break;
89             case INVALID:
90                 printf ("%10s|%10c|", "INVALID", yylval.carac); break;
91         }
92         printf ("\n");
93     }
94 }
95

```

**Entrada:**

```

1  ()
2  151265
3  12a35
4  _
5  H11 H1000
6  +-* /
7  <
8  <=
9  >
10 >=
11 ==
12 !=
13 <
14 <q=
15 >
16 >e=
17 =l=
18 412. 55.441
19 Afchvghj
20 B Cd B C
21 G T4 G T265
22 AAA
23

```

Saída:

1	texto	tipo	atributo
2	-----		
3	(	ABPAR	
4	)	FPAR	
5	151265	CTINT	151265
6	12	CTINT	12
7	a	INVAL	a
8	35	CTINT	35
9	_	INVAL	-
10	_	INVAL	
11	H11	CTHEX	17
12	H1000	CTHEX	4096
13	+	OPAD	MAIS
14	-	OPAD	MENOS
15	*	OPMULT	VEZES
16	/	OPMULT	DIV
17	<	OPREL	MENOR
18	<=	OPREL	MENIG
19	>	OPREL	MAIOR
20	>=	OPREL	MAIG
21	==	OPREL	IGUAL
22	!=	OPREL	DIFER
23	<	OPREL	MENOR
24	<	OPREL	MENOR
25	q	INVAL	q
26	=	INVAL	=
27	>	OPREL	MAIOR
28	>	OPREL	MAIOR
29	e	INVAL	e
30	=	INVAL	=
31	=	INVAL	=
32	1	CTINT	1
33	=	INVAL	=
34	412.	CTREAL	412.000000
35	55.441	CTREAL	55.441000
36	Afchvghj	ID	Afchvghj
37	B_Cd	ID	B_Cd
38	B	ID	B
39	_	INVAL	
40	C	ID	C
41	G_T4	ID	G_T4
42	G_T265	ID	G_T265
43	A	ID	A
44	A	ID	A
45	A	ID	A
46			