

# Tarefa 2 - Construção de Compiladores

**Aluno:** Felipe Alves Belisário

**Matrícula:** 11721BCC030

1 -

a)

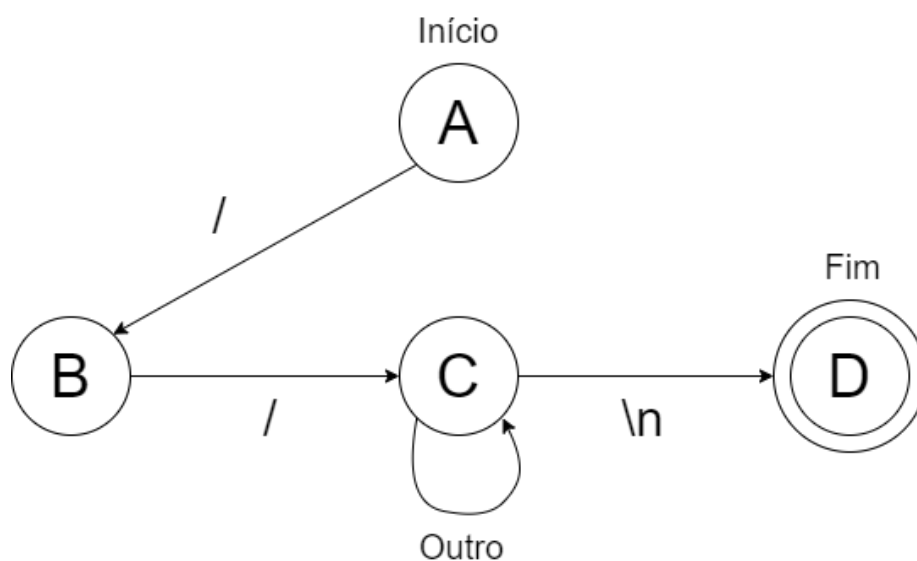
As expressões regulares para os comentários em C dados são as seguintes:

→ Comentários de linha: `// (.)*\n`

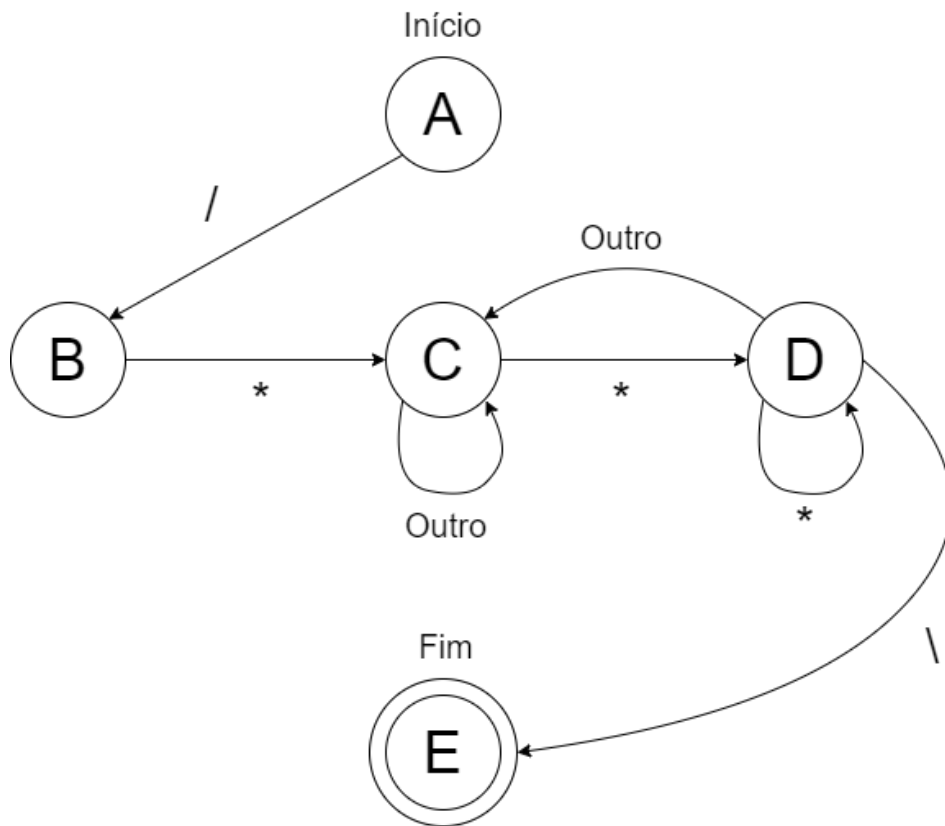
→ Comentários de bloco: `/* (.)**\`

A seguir são os diagramas de transição de cada um, respectivamente:

→ Comentários de linha:



→ Comentários de bloco:



b)

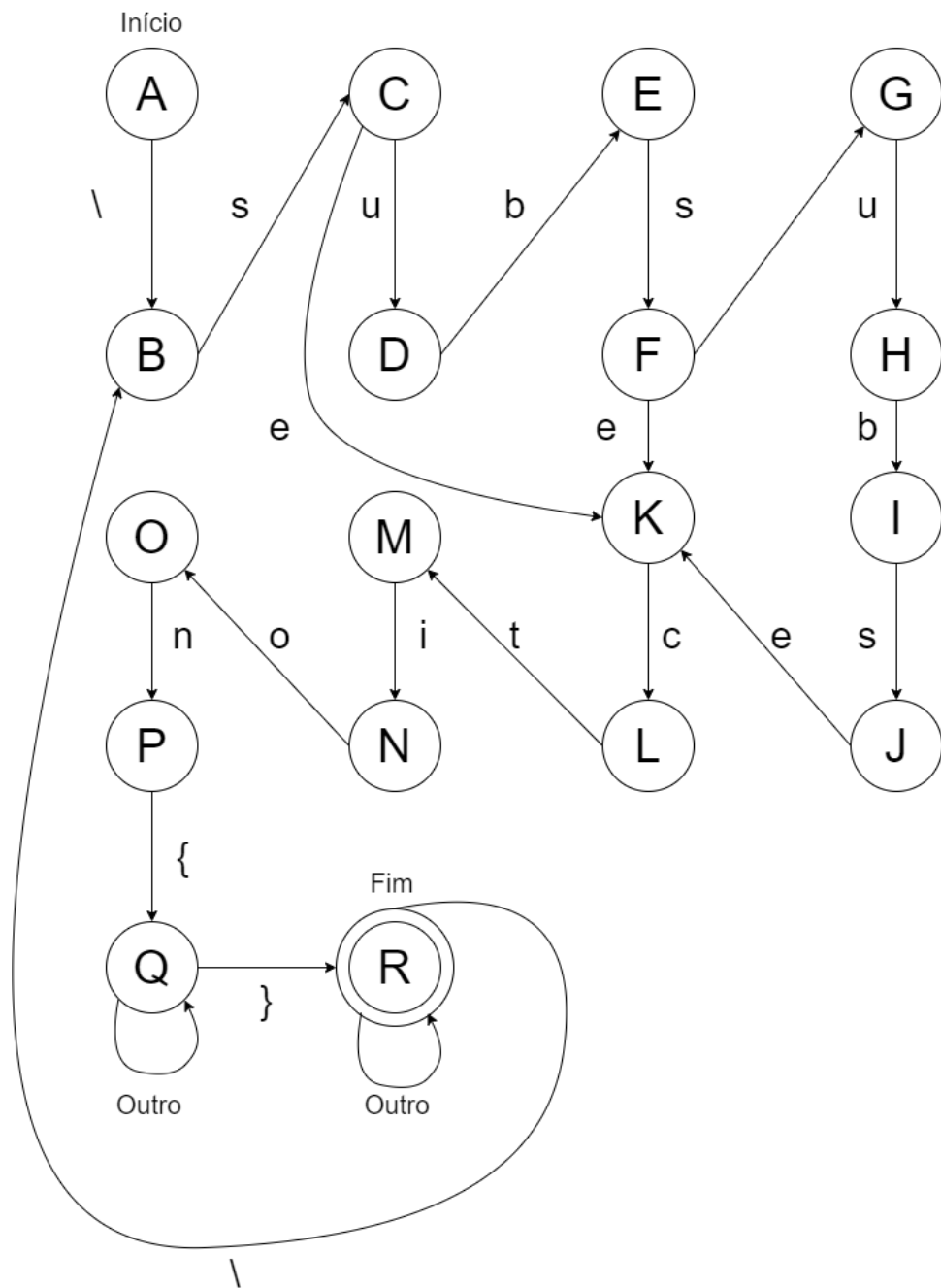
As tags identificadoras do latex das seções e subseções são:

→ *section:*            /section {texto} texto (\_subsection)\*

→ *subsection:*            /subsection {texto} texto (\_subsubsection)\*

→ *subsubsection:*            /subsubsection {texto} texto

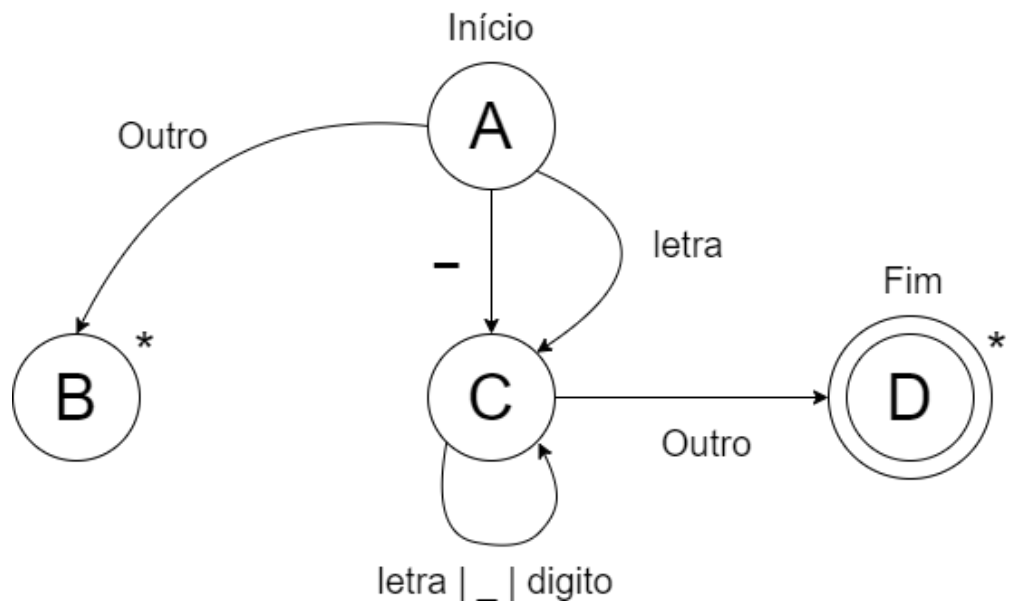
Segue o diagrama de transições:



2 -

a)

Segue o diagrama de transições:



Código:

*Start*

*f* <- A

*while*(*f* = 'A' or *f*='C') *do*

*g* <- nextChar()

*case* (*f*):

        A:

*if* *g* = '\_' or isLetter(*g*) *then*

*f* <- C

*else*

*f* <- B

        B:

*if* *g* = '\_' or isDigit(*g*) or isLetter(*g*) *then*

*f* <- C

*else*

*f* <- D

*end case*

*end while*

*if* *f* = 'B' *then*

```

next--
return "Cadeia de caracteres rejeitada!"
end if
next--

return "Cadeia de caracteres aceita!"

End

```

**b)**

Segue o diagrama de transições:

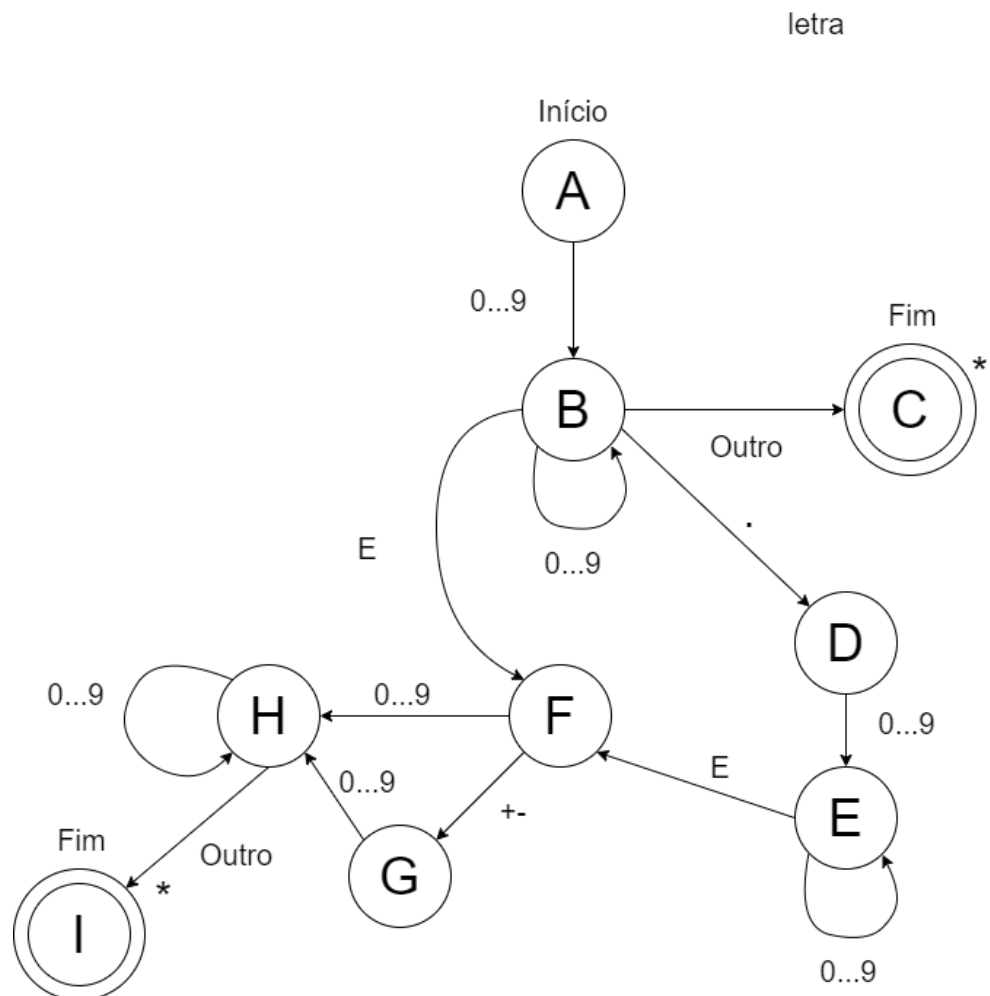


Tabela:

Estado	Final	.	+	-	E	[0...9]	Outro
A	N	-1	-1	-1	-1	1	-1
B	N	4	-1	-1	3	1	2
C	S	-1	-1	-1	-1	-1	-1
D	N	-1	-1	-1	-1	5	-1
E	N	-1	-1	-1	3	5	-1
F	N	-1	6	6	-1	7	-1
G	N	-1	-1	-1	-1	7	-1
H	N	-1	-1	-1	-1	7	8
I	S	-1	-1	-1	-1	-1	-1

Código:

*Start*

*f* <- A

*g* <- nextChar()

*while*( *g* != EOF and final(*f*) = 'N' and *f* != -1) *do*:

*f* <- move(*f*, *g*)

*g* <- nextChar()

*end while*

*if* final(*f*) = 'S' *then*

*return* "Cadeia de caracteres aceita!"

*else*

*return* "Cadeia de caracteres rejeitada!"

*End*

3 -

```
letter [a-zA-Z]
letter_ {letter} | _
digit [0-9]
letterDigit_ {letter_}{digit}
letter {letter}+
suffix "."{letters}
suffixes {sufix}+
emailID {letter_}{letterDigit_}*
email {emailID}"@"{letters}{suffixes}
```

```
digits2 {digito}{2}
digits3 {digits2}{digit}
digits11 {digit}{11}
```

```
cpf ({digits3}"{digits3}"{digits3}"-{digits2})|{digits11}
```

```
%%
{email} {
    printf("email <%s> encontrado!\n", yytext);
}
{cpf} {
    printf("cpf <%s> encontrado!\n", yytext);
}
. printf("Entrada nao conhecida!\n");
%%
```

```
int yywrap(){
    return 0;
}
```

```
int main() {
    yylex();
}
```