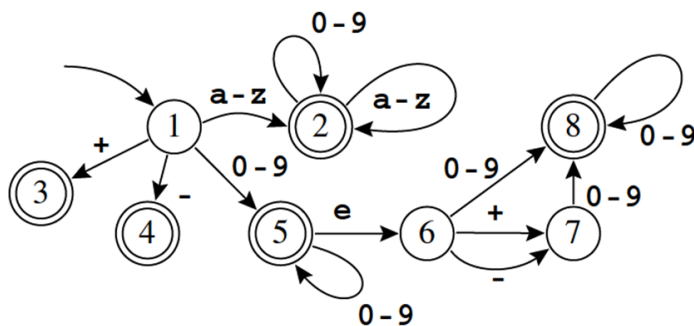


5COP093 - Lista de Exercícios 07

- Utilizando a ferramenta Flex, escreva um arquivo .l para gerar um analisador léxico que implementa o seguinte autômato:



O programa deve ler da entrada padrão e imprimir um *token* por linha e chegar ao fim ao encontrar o símbolo de fim de arquivo. Para caracteres que não façam parte do alfabeto, o programa deve imprimir a mensagem ERRO. Quebras de linha devem ser ignoradas. Exemplo:

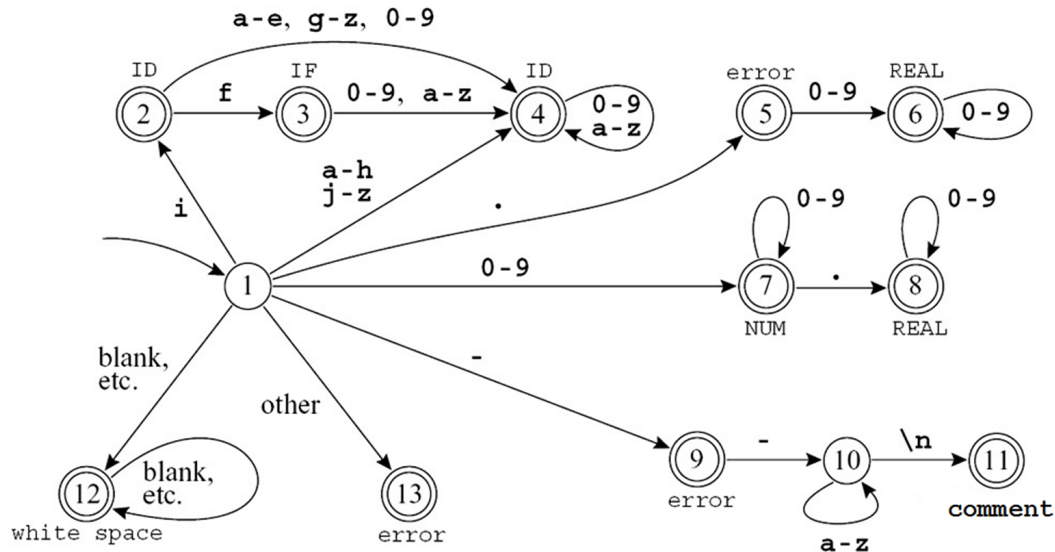
Entrada:

```
+-abcabc@ abc
fgh
```

Saída:

```
+
-
abcabc
ERRO
ERRO
abc
fgh
```

2. Utilizando a ferramenta Flex, escreva um arquivo .l para gerar um analisador léxico para o autômato a seguir:



O programa deve ler da entrada padrão e imprimir um *token* por linha e chegar ao fim ao encontrar o símbolo de fim de arquivo. Quebras de linha que não estejam associadas ao *token comment* devem ser ignoradas. Exemplo:

Entrada:

```
chuchu 666abobrinha@--blabla
0.0
```

Saída:

```
chuchu ID
white space
666 NUM
abobrinha ID
@ error
--blabla comment
0.0 REAL
```

3. Utilizando a ferramenta Flex, escreva um arquivo .l para gerar um analisador léxico que aceite as cadeias geradas pelas expressões regulares abaixo:

$(aba)^+$	(action 1)
$a(b^*)a$	(action 2)
$a b$	(action 3)

Com o programa gerado, reconheça os *tokens* na cadeia **abaabbaba** e diga ao final do reconhecimento como cada *token* é classificado, isto é: (action 1), (action 2) ou (action 3).

4. Utilizando a ferramenta Flex, escreva um arquivo .l para gerar um programa que mostra estatísticas sobre arquivos .c. O programa gerado deve se chamar **cstats** e receber como argumentos na linha de comando os arquivos a serem analisados, conforme o exemplo a seguir:

```
$ cstats ylevel.h ylevel.c ylscan.c ylpase.h ylpase.c
total    blank lines w/   nb, nc    semi- preproc. file
lines    lines comments   lines    colons direct.
-----+-----+-----+-----+-----+-----+-----
    69      10      27      33        8        9 ylevel.h
   237      35      37     165       60       13 ylevel.c
  1730     296     279    1185      391      303 ylscan.c
    54       7       1      47        7       36 ylpase.h
  1290     178     237     915      288     279 ylpase.c
  3380     526     581    2345     754     640 total
```

As informações a serem mostradas são:

- **total lines:** O total de linhas que o arquivo possui.
- **blank lines:** O total de linhas em branco.
- **lines w/ comments:** O total de linhas que possuem comentários, sejam eles de múltiplas linhas ou de uma única linha.
- **nb, nc lines:** O total de linhas que **não** estão em branco e **não** possuem comentários.
- **semi-colons:** O total de vezes que o símbolo ; (ponto-e-vírgula) aparece no arquivo.
- **preproc. direct:** O total de diretivas **#include** e **#define** que estão no arquivo.

O resultado deve ser impresso na tela seguindo o modelo apresentado neste exercício.