



**UEM – UNIVERSIDADE ESTADUAL DE MARINGÁ**  
**BACHARELADO EM INFORMÁTICA**  
Linguagens Formais e Autômatos – Professor Maurílio

# **Trabalho de Linguagens Formais e Autômatos**

## **Autômato Finito Determinístico**

Larissa Zanata Moraes

**RA: 99495**

Maringá

2019

## Introdução

Para a implementação desse trabalho foi utilizado a linguagem de programação Java na versão 8, a IDE Eclipse, e a JDK versão 8 também. A fim de contemplar o objetivo, também foi usado os princípios básicos da orientação a objeto.

## Objetivos/Justificativa

O programa implementado inicialmente deve receber do usuário uma descrição formal de um AFD , sendo ela composta por  $\{E, \Sigma, \delta, i, F\}$ :

**E:** conjunto finito dos estados;

**$\Sigma$ :** alfabeto da linguagem;

**$\delta$ :** função de transição;

**i:** estado inicial;

**F:** Conjunto dos estados finais;

E por fim recebe também do usuário, a palavra a ser testada.

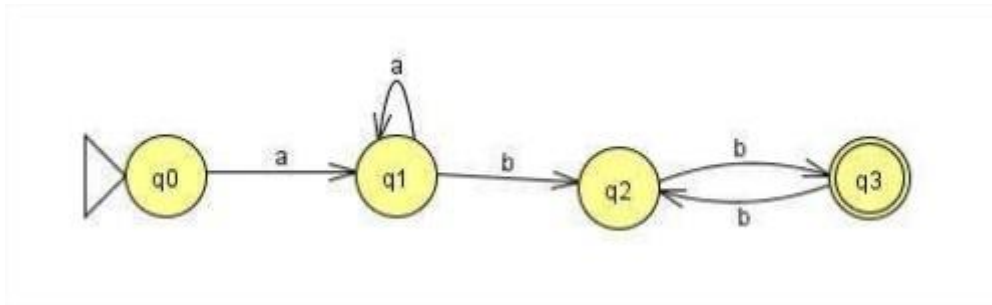
Seu objetivo é verificar se essa palavra é aceita ou rejeitada conforme a descrição formal do automato inserido no programa.

## Autômato Finito Determinístico

Um Autômato servem para dizer se uma palavra pertence ou não a uma Linguagem.

Com o Autômato pronto, e testando ele com uma palavra ele irá responder sim ou não dizendo se a palavra testada pertence a Linguagem a qual o Autômato reconhece.

Exemplo:



A descrição formal desse Autômato seria:

$E: \{q0, q1, q2, q3\}$

$\Sigma: \{a,b\}$

$\delta:$

	a	b
q0	q1	x
q1	q1	q2
q2	x	q3
q3	x	q2

$i: q0$

$F: \{q3\}$

Exemplos de palavras aceitas por esse autômato: { abb, aaabb, abbbb, aabb, aabbbbbbb, ... }

## Decisões de projeto para a implementação

### o Estrutura de dados utilizada

Para esse primeiro trabalho não foi preciso implementar uma estrutura de dados para satisfazer alguma parte, apenas foi usado as estruturas já existentes na linguagem Java. Também foi usado o básico da orientação a objeto para a criação da classe Autômato que dará origem aos objetos conforme forem criados.

### o Dentre outras questões relativas à implementação

Para a execução do programa existe alguns observações importantes a serem levadas em conta:

1 – Para o preenchimento dos conjuntos de estados, alfabetos, etc, se houver mais de um item, deve-se colocar os caracteres separados por virgula ‘,’. Exemplos:

E:

a0,q1,q2

$\Sigma$ :

a,b,c

```
Insira o alfabeto:
a,b
Insira os estados:
q0,q1,q2
Insira o estado Final:
q2
Insira o estado Inicial:
q1
```

*Imagem 1 – conjuntos sendo informados no programa em execução*

2 – Para o preenchimento da função de transição (no programa chamada de tabela), onde não existe transição deve-se colocar o sinal de ‘x’. Exemplo:

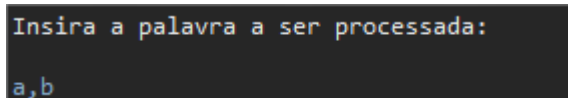
$\delta$ :

	a	b
q0	q1	x
q1	q1	q2
q2	x	q3
q3	x	q2

```
Entre com o valor matriz [q0][a]:
q1
Entre com o valor matriz [q0][b]:
x
Entre com o valor matriz [q1][a]:
x
Entre com o valor matriz [q1][b]:
q2
Entre com o valor matriz [q2][a]:
x
Entre com o valor matriz [q2][b]:
x
```

*Imagem 2 – Valores da função de transição sendo informados no programa.*

3 – Para o preenchimento da palavra a ser testada, deve-se separar os caracteres por vírgula ‘,’ caso tenham mais de um. Exemplo:



```
Insira a palavra a ser processada:  
a,b
```

*Imagem 3 – Palavra sendo inserida para ser processada no programa.*

## Conclusão

Dentro do arquivo compactado existe:

- Um arquivo com extensão .jar: lfa.jar
- Um arquivo com extensão .exe: lfa.exe

**Observação:** A extensão .exe não está funcionando corretamente, tive alguns impedimentos para fazê-lo executar no meu ambiente, aconselho rodar o arquivo com extensão .jar usando o seguinte comando via terminal:

Java -jar lfa.jar

- O projeto com o código fonte.
- O relatório do trabalho.

## Referências

DEVEMEDIA. Aprenda a trabalhar com vetores bidimensionais. Disponível em: <<https://www.devmedia.com.br/matrizes-aprenda-a-trabalhar-com-vetores-bidimensionais-revista-easy-java-magazine-22/25766>>. Acesso em 22 de Novembro de 2019.

GUJ. Matriz: percorrer, imprimir, comparar. Disponível em: <<https://respostas.guj.com.br/10611-matriz-percorrer-imprimir-comparar>>. Acesso em 23 de Novembro de 2019.