Aula 4 – Linguagens Regulares

Para estudar as Linguagens Regulares, usaremos os seguintes formalismos:

- *Autômato Finito:* trata-se de um formalismo operacional ou reconhecedor, sendo, basicamente, um *Sistema de Estados Finitos*;

- *Expressão Regular:* trata-se de um formalismo denotacional, também considerado gerador (pois pode-se inferir como construir todas as palavras da correspondente linguagem), o qual é definido a partir de conjuntos (linguagens) básicos e das operações de concatenação e de união;

- *Gramática Regular:* trata-se de um formalismo axiomático ou gerador o qual como o nome indica, é uma gramática, mas com restrições da forma das regras de produção.

De acordo com a hierarquia de Chomsky, as linguagens regulares constituem a classe de linguagens mais simples, sendo possível desenvolver algoritmos de reconhecimento, de geração ou de conversão entre formalismos de pouca complexidade, de grande eficiência e de fácil implementação.

A máquina, que é um procedimento, aceitador, ou reconhecedor, chamada *Autômato Finito (AF).* A palavra *"finito"* é incluída no nome para ressaltar que um **AF** só pode conter uma quantidade finita e limitada de informações, a qualquer momento. Essa informação é representada por um estado da máquina, e só existe um número finito de estados.

Esta restrição faz com que o **AF** seja severamente limitado na classe de linguagens que pode reconhecer, composta apenas pelas Linguagens Regulares.

Duas versões do AF serão estudadas: *Autômato Finito Determinístico (AFD)* e o *Autômato Finito Não Determinístico (AFND)*.

Uma Linguagem Regular pode ser definida de quatro formas:

- Através de uma Gramática Regular (definição);
- Através de um AFD que reconhece a linguagem;
- Através de um AFND que reconhece a linguagem;
- Através de uma Expressão Regular.

4.1 Autômato Finito Determinístico (AFD)

Um Autômato Finito Determinístico (AFD), ou simples Autômato Finito M é uma quíntupla ordenada: $M = (Q, \Sigma, \delta, q0, F)$

Na qual:

- Q é um *conjunto de estados* possíveis do autômato o qual é finito;

- Σ é um *alfabeto de símbolos de entrada*, ou simplesmente, *alfabeto de entrada*;

- δ é uma *função programa* ou simplesmente *programa*, ou ainda *função de transição:*

$$\delta: Q \times \Sigma \rightarrow Q$$

A qual é uma função parcial. Supondo que a função programa é definida para um estado p e um símbolo a, resultando no estado q, então: $\delta(p, a) = q$

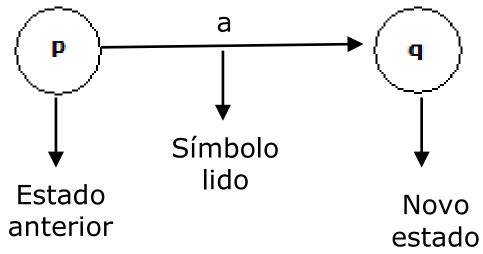
É uma *transição* do autômato;

- q0 é um elemento distinguido de Q, denominado *estado inicial*;

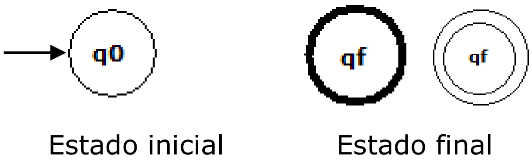
- F é um subconjunto de Q, denominado *conjunto de estados finais*.

Um autômato finito pode ser representado na forma de diagramas no qual:

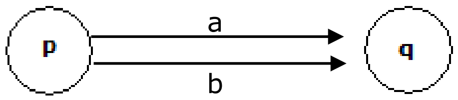
- Estados são nodos, representados por círculos;
- Transições são arestas, ligando os correspondentes dos nodos:

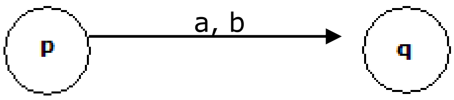


- Estados iniciais e finais são representados de forma distinta dos demais;



- Transições Paralelas (mesmo nodos origem e destino) podem alternativamente ser representados como a figura abaixo:



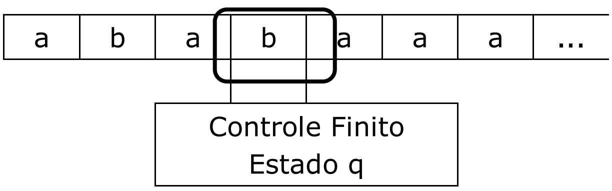


O nome *determinístico* faz referência ao fato de que δ é uma função (também chamada função próximo-estado), que determina precisamente o próximo estado a ser assumido quando a máquina M se encontra no estado q e lê da entrada o símbolo a: o estado $\delta(q,a)$.

De forma simplificada, podemos dizer que um AFD aceita uma cadeia se, partindo do *estado inicial*, e mudando de estado de acordo com a *função de transição*, o AFD atinge um *estado final* ao terminar de ler a cadeia.

Uma das maneiras de visualizar o funcionamento de um AFD é através de um *controle finito* que lê símbolos de uma *fita* de entrada (onde se encontra a cadeia de entrada), sequencialmente, da esquerda para a direita.

Os elementos do conjunto de estados Q representam os estados possíveis do controle finito. A operação se inicia no estado inicial i, lendo o primeiro símbolo da fita de entrada. Por conveniência, considera-se que a cabeça de leitura se move sobre a fita.



A Figura acima representa um AFD cujo controle está no estado q, e que está lendo o quarto símbolo da cadeia de entrada, um b.

Exemplo 1: Considere o AFD M = (Q, Σ, δ, i, F), onde temos:

Q = {q0, q1, q2, q3}

Σ = {a, b}

i = q0

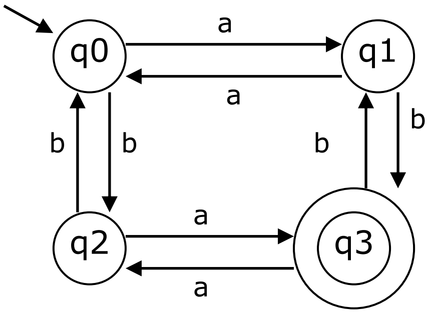
F = {q3}

e onde a função de transição δ: {q0, q1, q2, q3} x {a, b} → {q0, q1, q2, q3} é dada pela tabela abaixo:

δ	a	b
q0	q1	q2
q1	q0	q3
q2	q3	q0
q3	q2	q1

M = ({q0, q1, q2, q3}, {a, b}, q0, {q3})

Alternativamente, podemos representar o AFD M por um D*iagrama de Transições,* ou *Diagrama de Estados*, como o da figura abaixo.



Note que o Diagrama de Transições determina completamente o Autômato M, através de algumas convenções:

- Os estados são os nós do grafo, ou seja, Q = {q0, q1, q2, q3};

- O estado inicial é indicado pela seta, ou seja, i = q0;

- Os estados finais são indicados pelo círculo duplo: q3 – que é o único estado final, ou seja, F = {q3};

- As transições são as indicadas pelas arestas: δ(q0, a) = q1, δ(q0, b) = q2, δ(q1, a) = q0, ..., ou seja, δ é a mesma função representada pela tabela acima.

Qual a Linguagem desse AFD?

L = {w | |w| é par e w ∈ {a,b}}

Exemplo: Autômato Finito: aa ou bb como subpalavra

Considere a seguinte linguagem sobre o alfabeto {a, b}:

L = {w | w possui aa ou bb como subpalavra}

O autômato finito:

M = ({a, b}, {q0, q1, q2, q3}, δ, q0, {q3})

Onde δ é dada pela tabela abaixo, reconhece a Linguagem L.

δ	a	b
q0	q1	q2
q1	q3	q2
q2	q1	q3
q3	q3	q3

Construa o Autômato Finito Determinístico (AFD):

Exercícios:

1. Desenvolva Autômatos Finitos Determinísticos que reconheçam as seguintes linguagens sobre o Σ = {a, b}:

a) {w | w possui aaa como subpalavra}

b) {w | w possui número ímpar de a e número ímpar de b}

2. Construa o Autômato Finito Determinístico para a seguinte Linguagem:

L = $\{xba^n \mid x \in \{a, b\}^*, n \geq 0$ e x tem número par de as$\}$