

Autômatos Finitos Determinísticos e Autômatos Finitos Não- Determinísticos

Carlos Timóteo, M. Sc. CAPM

Motivação

- Como determinar se uma *substring* aparece num determinado texto;
- Como determinar se uma expressão foi formada corretamente
(5+1) – 8 `int main(int argc, char* argv){ return 0; }`
- Como procurar por uma *substring* em arquivos de texto no computador;

Definições Iniciais

- Conjunto: É uma coleção de objetos.
 - $L1 = \{ab, c, d\}$
 - $L2 = \{\text{vermelho}, \text{azul}, \text{vermelho}\} = \{\text{vermelho}, \text{azul}\}$
 - $L3 = \{3, 2, 1\} = \{1, 3, 2\}$
- Quantos elementos temos em L4:
 - $L4 = \{\text{azul}, 4, \{\text{preto}, \#\}\}$
- Operações entre conjuntos: União, Intersecção e Diferença

Definições Iniciais

- Propriedades dos Conjuntos
 - Idempotência: $A \cup A = A$
 - Comutatividade: $A \cup B = B \cup A$
 - Associatividade: $(A \cup B) \cup C = A \cup (B \cup C)$
 - Distributividade
 - Absorção
 - Leis de DeMorgan: $A - (B \cup C) = (A - B) \cup (A - C)$

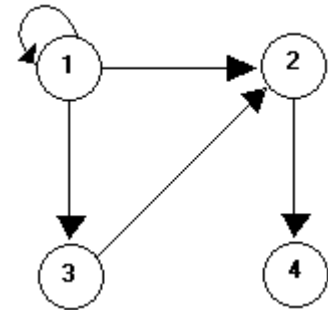
Definições Iniciais

- Conjuntos das Partes:
 - $2\{c,d\} = \{\{c,d\},\{c\},\{d\},\text{vazio}\}$
- Conjunto de todos os subconjuntos:
 - $U\{\{a,b\},\{b,c\},\{c,d\}\} = \{a,b,c,d\}$
- Partição de um conjunto:
 - $\{\{a,b\},\{c\},\{d\}\} \Rightarrow \{a,b,c,d\}$ válido
 - $\{\{a,b\},\{c\},\{d\}\} \Rightarrow \{\{b,c\},d\}$ inválido
- Relação: Um conjunto com uma restrição unária, binária ou terciária.
- Função: injetora, sobrejetora e bijetora

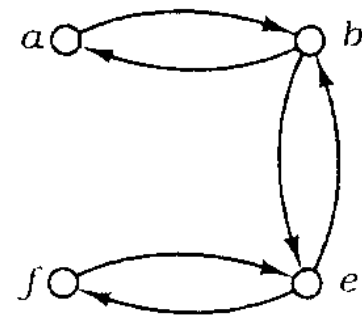
Definições Iniciais

- Grafos dirigidos: composto por arcos e vértices.

– $G = \{(1,1), (1,2), (1,3), (2,4), (3,2)\}$

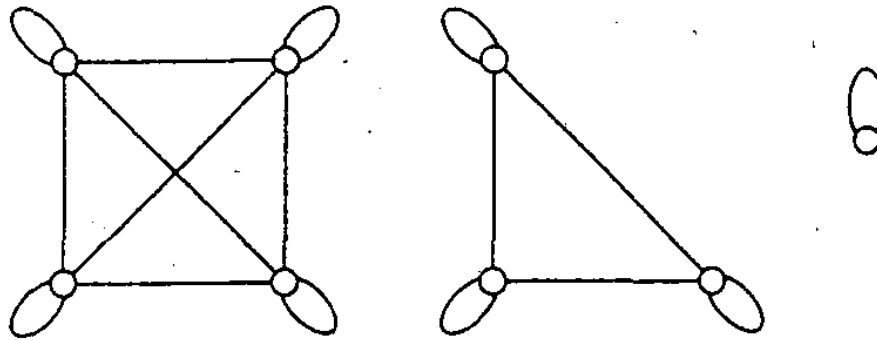


- Relação reflexiva (laços): (n,n)
- Relação simétrica (círculos):
 - $(a,b) (b,a)$
- Relação anti-simétrica



Definições Iniciais

- Relação de Equivalência: Simétrica e reflexiva



- Caminho e comprimento de um caminho:
 - $(a, \dots, d) = (a, b, c, d)$ $\text{comp}=4$
- Conjuntos finitos, infinitos e equinumerosos.

Técnicas de Prova

- Indução Matemática
- Dedução Matemática
- Abdução Matemática

Dada uma **premissa**, uma **conclusão** e uma regra segundo a qual a **premissa** *implica* na **conclusão**.

Técnicas de Prova

- **Dedução** corresponde a determinar a **conclusão**. Utiliza-se da **regra** e sua **premissa** para chegar a uma **conclusão**.

"Quando chove, a grama fica molhada. Choveu hoje. Portanto, a grama está molhada."

- **Indução** é determinar a **regra**. É aprender a **regra** a partir de diversos exemplos de como a **conclusão** segue da **premissa**.

"A grama ficou molhada todas as vezes em que choveu. Então, se chover amanhã, a grama ficará molhada."

Técnicas de Prova

- **Abdução** significa determinar a ***premissa***.
Usa-se a ***conclusão*** e a *regra* para defender
que a ***premissa*** poderia explicar a ***conclusão***.
"Quando chove, a grama fica molhada. A grama está
molhada, então pode ter chovido."

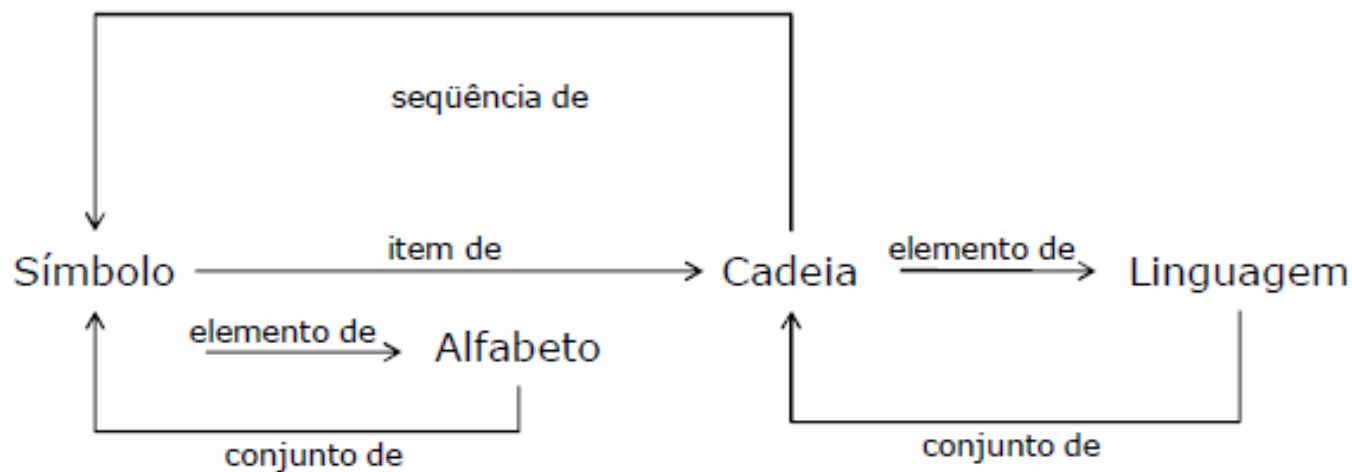
Alfabetos e Linguagens

- **Alfabeto** é um conjunto de símbolos. É um conjunto finito de qualquer tipo.
- Uma **string** em um alfabeto é uma sequência finita de símbolos.
 - *melancia* é uma **string** do alfabeto grego-romano;
 - *01011100* é uma **string** do alfabeto binário;
- Propriedades: concatenação, sufixo, prefixo, inverso..

Alfabetos e Linguagens

- **Estrela de Kleene:** Denotada por L^* é o conjunto de todas as *strings* obtidas pela concatenação de zero ou mais *strings* de L .
 - Se $L=\{01,0,100\}$, então 01001100000 pertence a L^*
- **Linguagem:** Conjunto de todas as strings sobre um alfabeto.

Alfabetos e Linguagens



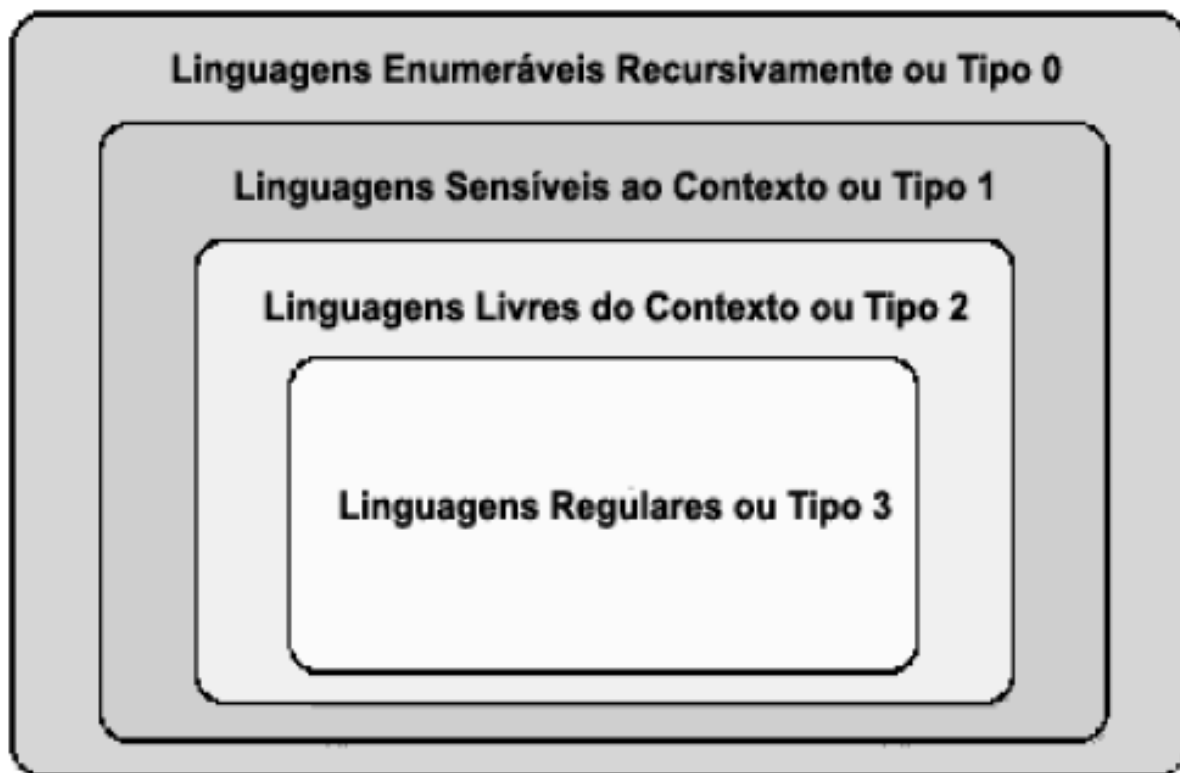
Expressões Regulares

- **Expressão regular:** Descreve uma linguagem finita ou infinita de elementos exclusivamente por meio de símbolos únicos e *. Todas as linguagens regulares sobre um alfabeto podem ser descritas por expressões regulares.

Tipos de Formalismos

- Reconhecedores
 - Recebe uma palavra e retorna um valor para dizer se ela é ou não da linguagem
- Geradores
 - Define um conjunto de regras que podem ser combinadas para gerar palavras
- Denotacional
 - Uma expressão que denota de modo geral as palavras da linguagem

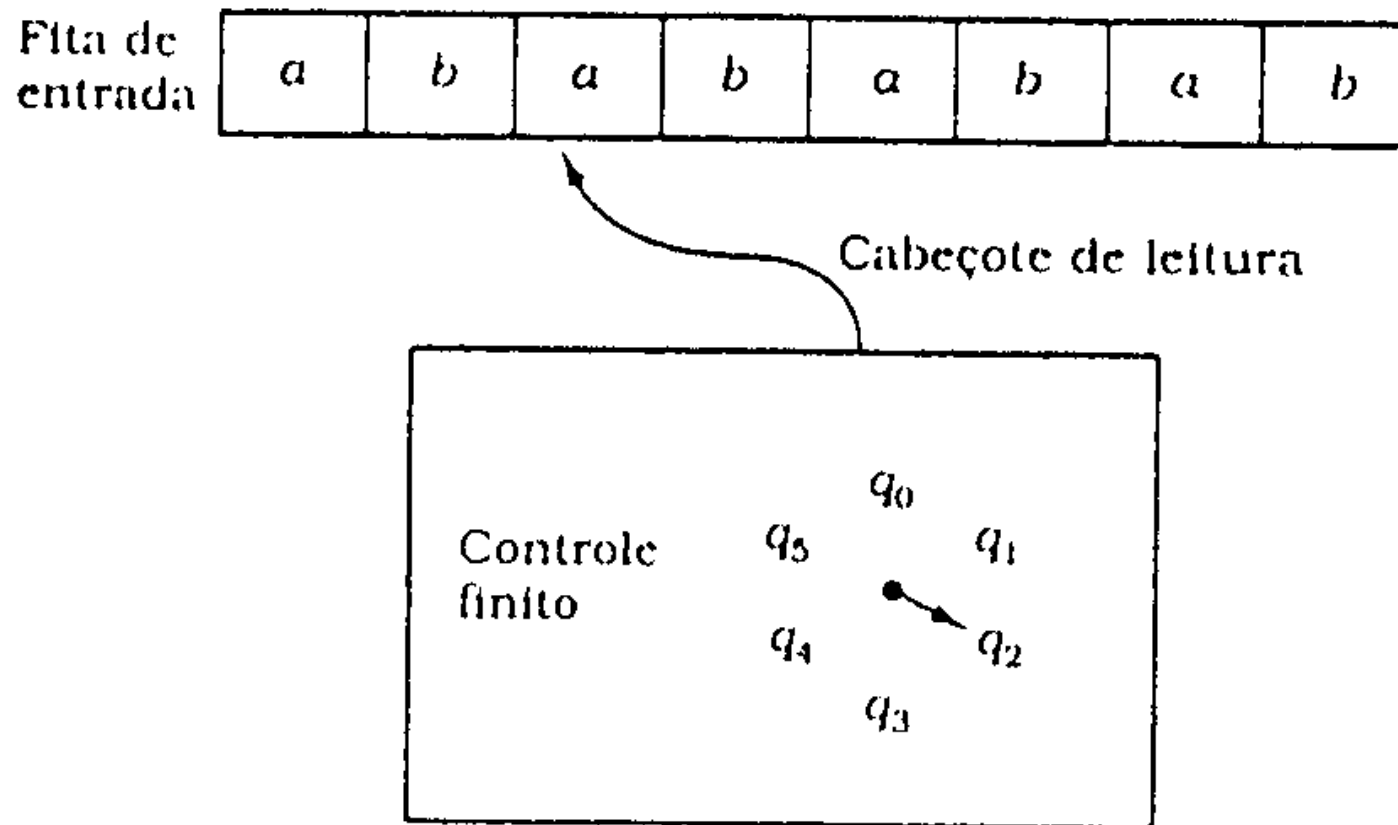
Hierarquia de Chomsky



Autômato Finito Determinístico

- Considere uma máquina capaz de ler símbolo e avançar para o próximo se o símbolo lido for válido.
- Ele tem uma unidade de processamento, recebe como entrada uma cadeia que lhe é fornecida em uma fita de entrada e não produz nenhuma saída real, exceto a indicação de aceitação da entrada.
- Em outras palavras, é um dispositivo de reconhecimento de linguagens.
- Baseada no conceito de máquinas de estados finitos.

Autômato Finito Determinístico



Máquinas de estados finitos

- Conjunto finito de estados
 - Tem um estado inicial
- Mudança de estados
 - Depende do estado atual e de certa entrada

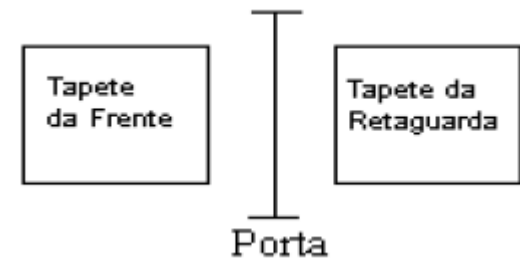
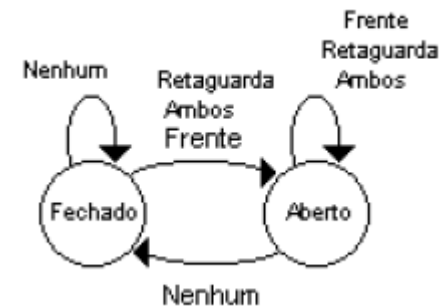


Tabela de Transição

	NENHUM	FRENTE	RETAGUARDA	AMBOS
FECHADO	FECHADO	ABERTO	ABERTO	ABERTO
ABERTO	FECHADO	ABERTO	ABERTO	ABERTO



AFD – Definição Formal

Um Autômato Finito Determinístico (AFD) M é uma 5-upla:

$$M = (Q, \Sigma, \delta, q_0, F), \text{ onde}$$

Q : conjunto finito de estados do autômato;

Σ : alfabeto de símbolos de entrada;

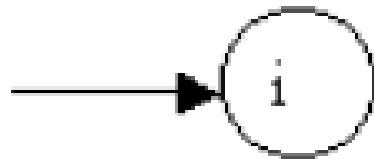
δ : função programa ou função de transição (parcial)

$\delta: Q \times \Sigma \rightarrow Q$. Significa dizer que permanecendo em um estado e lendo um símbolo do alfabeto faz o autômato passar para outro estado ou mesmo ficar no mesmo

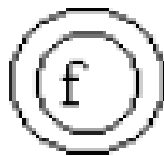
q_0 : estado inicial ($q_0 \in Q$)

F : conjunto de estados finais ou estados de aceitação
($F \subseteq Q$)

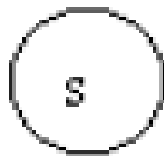
AFD – Representação Gráfica



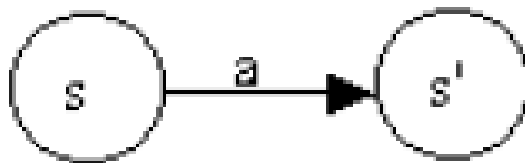
Estado inicial



Estado final



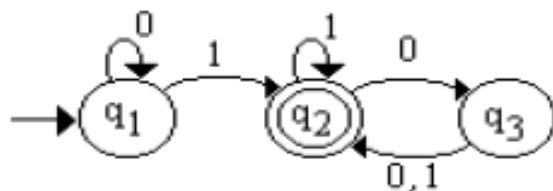
Restantes estados



Transição $\delta(s,a)=s'$

AFD – Representação Gráfica

Um autômato finito M_1 : (diagrama de estados)



M_1 tem **3 estados**, q_1 , q_2 , q_3 ;

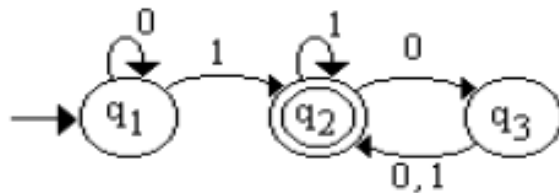
M_1 **inicia** no estado q_1 ;

M_1 tem um **estado final**, q_2 ;

Os arcos que vão de um estado p/ outro chamam-se **transições**.

AFD – Exemplo Funcionamento

Exemplo: entrada 1101



1. Inicia no estado q_1 .
2. Lê 1, segue transição de q_1 p/ q_2 .
3. Lê 1, segue transição de q_2 p/ q_2 .
4. Lê 0, segue transição de q_2 p/ q_3 .
5. Lê 1, segue transição de q_3 p/ q_2 .
6. Pára c/ saída reconhece.

AFD - Propriedades

- Um AFD nunca entra em loop infinito
- Novos símbolos da entrada são lidos a cada aplicação da função programa, o processo de reconhecimento de qualquer cadeia pára de duas maneiras: **aceitando** ou **rejeitando** uma entrada.

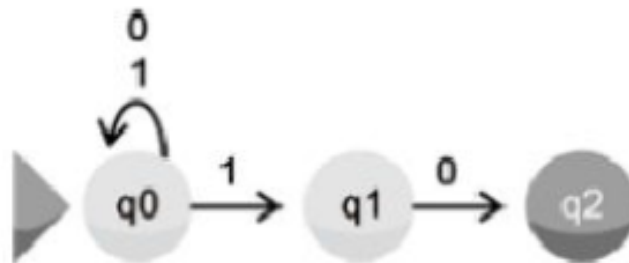
Autômatos Finitos Não-Determinísticos

- Adiciona o não-determinismo aos autômatos finitos.
 - Capacidade de mudar de estado de forma apenas parcialmente determinada pelo estado corrente e pelo símbolo de entrada.
 - Podemos ter zero, uma ou mais transições de estado com o mesmo símbolo de entrada.

AFD vs AFND

- Determinístico
 - Transições bem definidas
 - Função de transição leva a um único estado
 - Sequência de estados é única para cada palavra
- Não-determinístico
 - Transições ambíguas
 - Função de transição leva a vários estados alternativos
 - Várias sequências possíveis

Autômatos Finitos Não Determinísticos



- Um estado pode ter zero, um ou mais arcos "saindo" para cada símbolo do alfabeto;
- zero, um ou mais arcos podem sair de cada estado rotulados com λ .

AFND – Definição Formal

Um Autômato Finito Não-Determinístico (AFND) é uma 5-tupla onde:

1. Q é um conjunto finito de estados;
2. Σ é um alfabeto finito;
3. $\delta : Q \times \Sigma_{\lambda} \rightarrow P(Q)$ é a função de transição;
4. $q_0 \in Q$ é o estado inicial; e
5. $F \subseteq Q$ é o conjunto de estados de aceitação.

Uma sequência de entrada $a,b,c\dots d$ é aceita por um AFND se existe uma sequência de transições, correspondendo a sequência de entrada, que leva do estado inicial a algum dos estados finais.

Equivalência AFND/AFD

- De um AFD é possível criar um AFND equivalente?
 - Trivial de mostrar
 - Basta criar um AFND cuja função leva a conjuntos unitários
- De um AFND é possível criar um AFD equivalente?
 - Dado $\mathbf{M} = (\mathbf{T}, \mathbf{Q}, \delta, \mathbf{q}_0, \mathbf{F})$ não-determinístico, construir \mathbf{M}' determinístico
 - Veremos como fazer...

Próxima Aula

- Expressões Regulares
- Transformações
 - AFD \rightarrow AFND
 - AFND \rightarrow AFD
 - AFND \rightarrow AFD \rightarrow ER
 - ER \rightarrow AFND \rightarrow AFD