

# Introdução à Teoria da Computação

## Autômatos Finitos não Determinísticos (AFNs)

**Professor Murilo V. G. da Silva**

Departamento de Informática  
Universidade Federal do Paraná

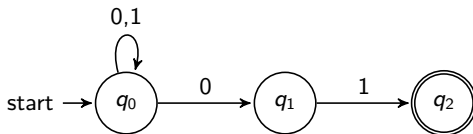
Segundo semestre de 2026

# Autômatos Finitos não Determinísticos (AFNs)

Autômato com uma habilidade “mágica”: adivinhar transição

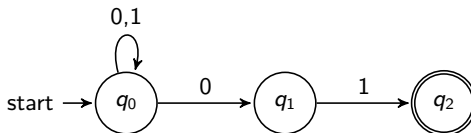
# Autômatos Finitos não Determinísticos (AFNs)

Autômato com uma habilidade “mágica”: adivinhar transição



# Autômatos Finitos não Determinísticos (AFNs)

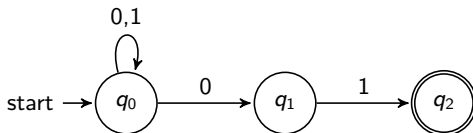
Autômato com uma habilidade “mágica”: adivinhar transição



- Diferenças entre AFDs e AFNs?

# Autômatos Finitos não Determinísticos (AFNs)

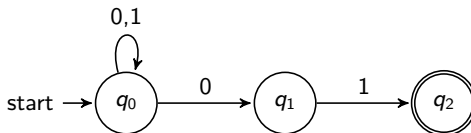
Autômato com uma habilidade “mágica”: adivinhar transição



- Diferenças entre AFDs e AFNs?
- A string 001 é aceita ou rejeitada pelo autômato?

# Autômatos Finitos não Determinísticos (AFNs)

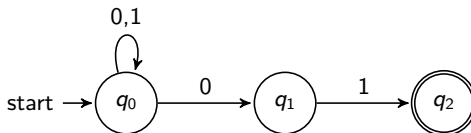
Autômato com uma habilidade “mágica”: adivinhar transição



- Diferenças entre AFDs e AFNs?
- A string 001 é aceita ou rejeitada pelo autômato?
- Quais são as strings **aceitas**?

# Autômatos Finitos não Determinísticos (AFNs)

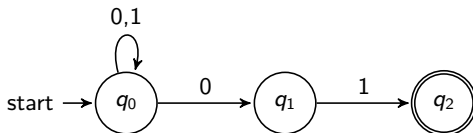
Autômato com uma habilidade “mágica”: adivinhar transição



- Diferenças entre AFDs e AFNs?
- A string 001 é aceita ou rejeitada pelo autômato?
- Quais são as strings **aceitas**?  
⇒ as strings que o autômato consegue aceitar

# Autômatos Finitos não Determinísticos (AFNs)

Autômato com uma habilidade “mágica”: adivinhar transição

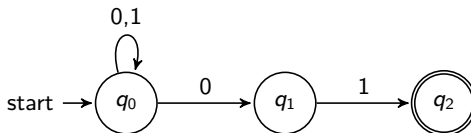


- Diferenças entre AFDs e AFNs?
- A string 001 é aceita ou rejeitada pelo autômato?
- Quais são as strings **aceitas**?

⇒ as strings que o autômato consegue aceitar exemplo: 001

# Autômatos Finitos não Determinísticos (AFNs)

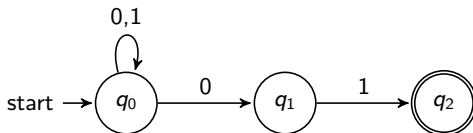
Autômato com uma habilidade “mágica”: adivinhar transição



- Diferenças entre AFDs e AFNs?
- A string 001 é aceita ou rejeitada pelo autômato?
- Quais são as strings **aceitas**?  
⇒ as strings que o autômato consegue aceitar exemplo: 001
- Quais são as strings **rejeitadas**?

# Autômatos Finitos não Determinísticos (AFNs)

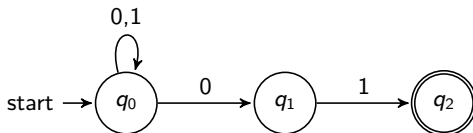
Autômato com uma habilidade “mágica”: adivinhar transição



- Diferenças entre AFDs e AFNs?
- A string 001 é aceita ou rejeitada pelo autômato?
- Quais são as strings **aceitas**?
  - ⇒ as strings que o autômato consegue aceitar exemplo: 001
- Quais são as strings **rejeitadas**?
  - ⇒ as strings que o autômato não consegue aceitar

# Autômatos Finitos não Determinísticos (AFNs)

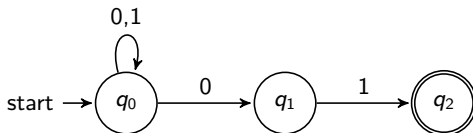
Autômato com uma habilidade “mágica”: adivinhar transição



- Diferenças entre AFDs e AFNs?
- A string 001 é aceita ou rejeitada pelo autômato?
- Quais são as strings **aceitas**?
  - ⇒ as strings que o autômato consegue aceitar exemplo: 001
- Quais são as strings **rejeitadas**?
  - ⇒ as strings que o autômato não consegue aceitar exemplo 000

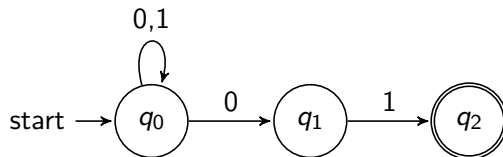
# Autômatos Finitos não Determinísticos (AFNs)

Autômato com uma habilidade “mágica”: adivinhar transição

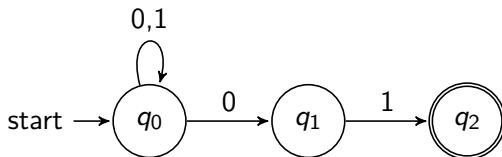


- Diferenças entre AFDs e AFNs?
- A string 001 é aceita ou rejeitada pelo autômato?
- Quais são as strings **aceitas**?
  - ⇒ as strings que o autômato consegue aceitar exemplo: 001
- Quais são as strings **rejeitadas**?
  - ⇒ as strings que o autômato não consegue aceitar exemplo 000
- Não determinismo seria a terminologia correta?

# Diferença entre AFNs e AFDs

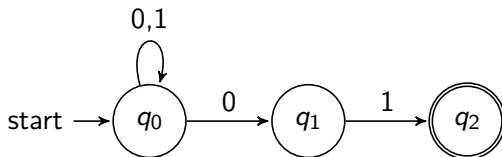


# Diferença entre AFNs e AFDs



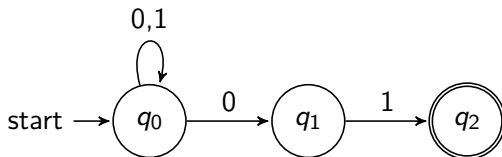
- Única diferença:

# Diferença entre AFNs e AFDs



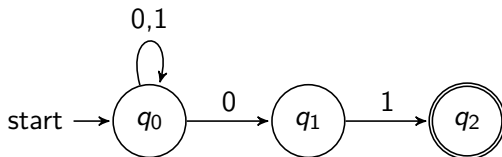
- Única diferença:  $\delta(q, a) = S$ ,

# Diferença entre AFNs e AFDs



- Única diferença:  $\delta(q, a) = S$ ,  
 $S$  é um conjunto de estados

# Diferença entre AFNs e AFDs

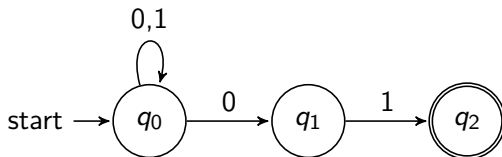


- Única diferença:  $\delta(q, a) = S$ ,  
 $S$  é um conjunto de estados

Por exemplo:

- $\delta(q_0, 0) = \{q_0, q_1\}$

# Diferença entre AFNs e AFDs

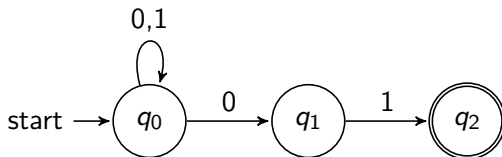


- Única diferença:  $\delta(q, a) = S$ ,  
 $S$  é um conjunto de estados

Por exemplo:

- $\delta(q_0, 0) = \{q_0, q_1\}$        $\delta(q_1, 0) = \emptyset$

# Diferença entre AFNs e AFDs



- Única diferença:  $\delta(q, a) = S$ ,  
 $S$  é um conjunto de estados

Por exemplo:

- $\delta(q_0, 0) = \{q_0, q_1\}$        $\delta(q_1, 0) = \emptyset$        $\delta(q_1, 1) = \{q_2\}$

# Definição Formal de AFN

# Definição Formal de AFN

Def.: Autômato Finito não Determinístico (AFN)

Um **AFN** é uma 5-tupla  $A = (Q, \Sigma, \delta, q_0, F)$ , tal que:

# Definição Formal de AFN

## Def.: Autômato Finito não Determinístico (AFN)

Um **AFN** é uma 5-tupla  $A = (Q, \Sigma, \delta, q_0, F)$ , tal que:

$Q$  é o conjunto de *estados*;

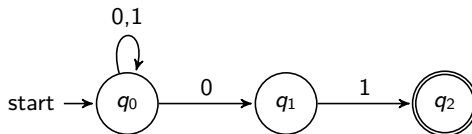
$\Sigma$  é o conjunto de *símbolos de entrada*;

$\delta$  é a *função de transição*  $\delta : Q \times \Sigma \rightarrow \mathcal{P}(Q)$ ;

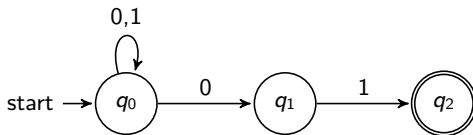
$q_0 \in Q$  é o *estado inicial*;

$F \subseteq Q$  é o conjunto de *estados finais*.

# Exemplo de AFN (formal)

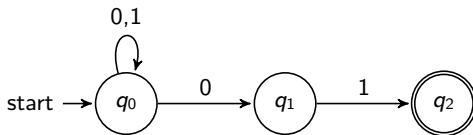


# Exemplo de AFN (formal)



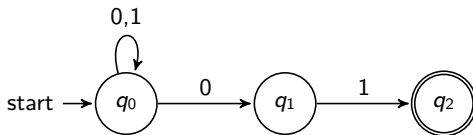
$N = (Q, \Sigma, \delta, q_0, F)$ , tal que

# Exemplo de AFN (formal)



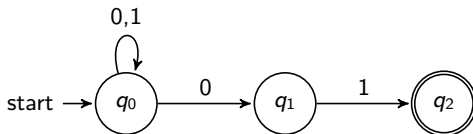
$N = (Q, \Sigma, \delta, q_0, F)$ , tal que  $Q = \{q_0, q_1, q_2\}$ ,

# Exemplo de AFN (formal)



$N = (Q, \Sigma, \delta, q_0, F)$ , tal que  $Q = \{q_0, q_1, q_2\}$ ,  $\Sigma = \{0, 1\}$

# Exemplo de AFN (formal)



$N = (Q, \Sigma, \delta, q_0, F)$ , tal que  $Q = \{q_0, q_1, q_2\}$ ,  $\Sigma = \{0, 1\}$  e

$$\delta(q_0, 0) = \{q_0, q_1\}$$

$$\delta(q_0, 1) = \{q_0\}$$

$$\delta(q_1, 1) = \{q_2\}$$

$$\delta(q_1, 0) = \delta(q_2, 0) = \delta(q_2, 1) = \emptyset$$

# Tabelas de transições

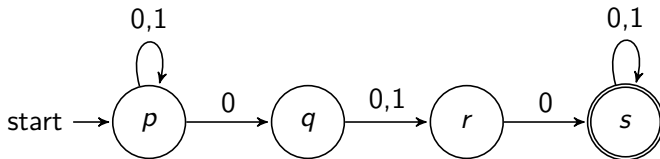
# Tabelas de transições

|                 | 0          | 1           |
|-----------------|------------|-------------|
| $\rightarrow p$ | $\{p, q\}$ | $\{p\}$     |
| $q$             | $\{r\}$    | $\{r\}$     |
| $r$             | $\{s\}$    | $\emptyset$ |
| $*s$            | $\{s\}$    | $\{s\}$     |

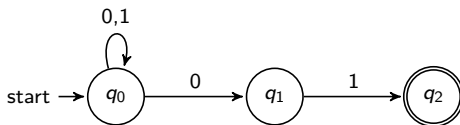
# Tabelas de transições

|                 | 0          | 1           |
|-----------------|------------|-------------|
| $\rightarrow p$ | $\{p, q\}$ | $\{p\}$     |
| $q$             | $\{r\}$    | $\{r\}$     |
| $r$             | $\{s\}$    | $\emptyset$ |
| $*s$            | $\{s\}$    | $\{s\}$     |

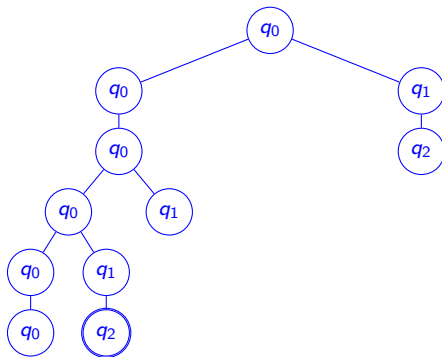
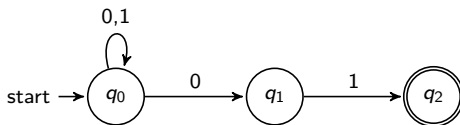
Diagrama correspondente:



# Árvore de computações possíveis para 01001



## Árvore de computações possíveis para 01001



# Árvore de computações possíveis

# Árvore de computações possíveis

Def.:  $(A, q, w)$ -árvores

Seja  $(A, q, w)$  uma tripla tal que  $A$  é um AFN,  $q$  um é estado de  $A$  e  $w$  um string do alfabeto de  $A$ .

# Árvore de computações possíveis

## Def.: $(A, q, w)$ -árvores

Seja  $(A, q, w)$  uma tripla tal que  $A$  é um AFN,  $q$  um é estado de  $A$  e  $w$  um string do alfabeto de  $A$ .

Uma  **$(A, q, w)$ -árvore** é uma árvore enraizada em  $q$  definida a seguir:

# Árvore de computações possíveis

Def.:  $(A, q, w)$ -árvores

Seja  $(A, q, w)$  uma tripla tal que  $A$  é um AFN,  $q$  um é estado de  $A$  e  $w$  um string do alfabeto de  $A$ .

Uma  $(A, q, w)$ -árvore é uma árvore enraizada em  $q$  definida a seguir:

**Base:**  $w = \epsilon$

# Árvore de computações possíveis

## Def.: $(A, q, w)$ -árvores

Seja  $(A, q, w)$  uma tripla tal que  $A$  é um AFN,  $q$  um é estado de  $A$  e  $w$  um string do alfabeto de  $A$ .

Uma  **$(A, q, w)$ -árvore** é uma árvore enraizada em  $q$  definida a seguir:

**Base:**  $w = \epsilon$

A árvore contém apenas o nó raiz  $q$

# Árvore de computações possíveis

## Def.: $(A, q, w)$ -árvores

Seja  $(A, q, w)$  uma tripla tal que  $A$  é um AFN,  $q$  um estado de  $A$  e  $w$  um string do alfabeto de  $A$ .

Uma  **$(A, q, w)$ -árvore** é uma árvore enraizada em  $q$  definida a seguir:

**Base:**  $w = \epsilon$

A árvore contém apenas o nó raiz  $q$

**Indução:**  $w$  é da forma  $ax$ , tal que  $a \in \Sigma$  e  $x \in \Sigma^*$

# Árvore de computações possíveis

## Def.: $(A, q, w)$ -árvores

Seja  $(A, q, w)$  uma tripla tal que  $A$  é um AFN,  $q$  um estado de  $A$  e  $w$  um string do alfabeto de  $A$ .

Uma  **$(A, q, w)$ -árvore** é uma árvore enraizada em  $q$  definida a seguir:

**Base:**  $w = \epsilon$

A árvore contém apenas o nó raiz  $q$

**Indução:**  $w$  é da forma  $ax$ , tal que  $a \in \Sigma$  e  $x \in \Sigma^*$

Seja  $\delta$  a função de transição do AFN.

# Árvore de computações possíveis

## Def.: $(A, q, w)$ -árvores

Seja  $(A, q, w)$  uma tripla tal que  $A$  é um AFN,  $q$  um é estado de  $A$  e  $w$  um string do alfabeto de  $A$ .

Uma  **$(A, q, w)$ -árvore** é uma árvore enraizada em  $q$  definida a seguir:

**Base:**  $w = \epsilon$

A árvore contém apenas o nó raiz  $q$

**Indução:**  $w$  é da forma  $ax$ , tal que  $a \in \Sigma$  e  $x \in \Sigma^*$

Seja  $\delta$  a função de transição do AFN. Se  $\delta(q, a) = \{p_1, p_2, \dots, p_k\}$ , então então a árvore contém o nó  $q$  e

# Árvore de computações possíveis

## Def.: $(A, q, w)$ -árvores

Seja  $(A, q, w)$  uma tripla tal que  $A$  é um AFN,  $q$  um estado de  $A$  e  $w$  um string do alfabeto de  $A$ .

Uma  **$(A, q, w)$ -árvore** é uma árvore enraizada em  $q$  definida a seguir:

**Base:**  $w = \epsilon$

A árvore contém apenas o nó raiz  $q$

**Indução:**  $w$  é da forma  $ax$ , tal que  $a \in \Sigma$  e  $x \in \Sigma^*$

Seja  $\delta$  a função de transição do AFN. Se  $\delta(q, a) = \{p_1, p_2, \dots, p_k\}$ , então a árvore contém o nó  $q$  e os filhos de  $q$  são  $p_1, p_2, \dots, p_k$ , estes sendo raízes de  $(A, p_i, x)$ -árvores.

# Árvore de computações possíveis

## Def.: $(A, q, w)$ -árvores

Seja  $(A, q, w)$  uma tripla tal que  $A$  é um AFN,  $q$  um estado de  $A$  e  $w$  um string do alfabeto de  $A$ .

Uma  **$(A, q, w)$ -árvore** é uma árvore enraizada em  $q$  definida a seguir:

**Base:**  $w = \epsilon$

A árvore contém apenas o nó raiz  $q$

**Indução:**  $w$  é da forma  $ax$ , tal que  $a \in \Sigma$  e  $x \in \Sigma^*$

Seja  $\delta$  a função de transição do AFN. Se  $\delta(q, a) = \{p_1, p_2, \dots, p_k\}$ , então a árvore contém o nó  $q$  e os filhos de  $q$  são  $p_1, p_2, \dots, p_k$ , estes sendo raízes de  $(A, p_i, x)$ -árvores.

**Exercício:** Considere o autômato  $N$  do slide anterior.

# Árvore de computações possíveis

## Def.: $(A, q, w)$ -árvores

Seja  $(A, q, w)$  uma tripla tal que  $A$  é um AFN,  $q$  um estado de  $A$  e  $w$  um string do alfabeto de  $A$ .

Uma  **$(A, q, w)$ -árvore** é uma árvore enraizada em  $q$  definida a seguir:

**Base:**  $w = \epsilon$

A árvore contém apenas o nó raiz  $q$

**Indução:**  $w$  é da forma  $ax$ , tal que  $a \in \Sigma$  e  $x \in \Sigma^*$

Seja  $\delta$  a função de transição do AFN. Se  $\delta(q, a) = \{p_1, p_2, \dots, p_k\}$ , então a árvore contém o nó  $q$  e os filhos de  $q$  são  $p_1, p_2, \dots, p_k$ , estes sendo raízes de  $(A, p_i, x)$ -árvores.

**Exercício:** Considere o autômato  $N$  do slide anterior.

Desenhe as árvores  $(N, q_0, 00)$  e  $(N, q_2, 11)$  correspondentes.

# Árvore de computações possíveis

# Árvore de computações possíveis

## Def.: $(A, q, w)$ -árvores

Seja  $(A, q, w)$  uma tripla tal que  $A$  é um AFN,  $q$  um é estado de  $A$  e  $w$  um string do alfabeto de  $A$ .

Uma  **$(A, q, w)$ -árvore** é uma árvore enraizada em  $q$  definida a seguir:

**Base:**  $w = \epsilon$

A árvore contém apenas o nó raiz  $q$

**Indução:**  $w$  é da forma  $ax$ , tal que  $a \in \Sigma$  e  $x \in \Sigma^*$

Seja  $\delta$  a função de transição do AFN. Se  $\delta(q, a) = \{p_1, p_2, \dots, p_k\}$ , então então a árvore contém o nó  $q$  e  $q$  possui os filhos  $p_1, p_2, \dots, p_k$  que são raízes de uma  $(A, p_i, x)$ -árvore.

# Árvore de computações possíveis

## Def.: $(A, q, w)$ -árvores

Seja  $(A, q, w)$  uma tripla tal que  $A$  é um AFN,  $q$  um é estado de  $A$  e  $w$  um string do alfabeto de  $A$ .

Uma  $(A, q, w)$ -árvore é uma árvore enraizada em  $q$  definida a seguir:

**Base:**  $w = \epsilon$

A árvore contém apenas o nó raiz  $q$

**Indução:**  $w$  é da forma  $ax$ , tal que  $a \in \Sigma$  e  $x \in \Sigma^*$

Seja  $\delta$  a função de transição do AFN. Se  $\delta(q, a) = \{p_1, p_2, \dots, p_k\}$ , então então a árvore contém o nó  $q$  e  $q$  possui os filhos  $p_1, p_2, \dots, p_k$  que são raízes de uma  $(A, p_i, x)$ -árvore.

## Def.: Árvore de computações possíveis

# Árvore de computações possíveis

## Def.: $(A, q, w)$ -árvores

Seja  $(A, q, w)$  uma tripla tal que  $A$  é um AFN,  $q$  um é estado de  $A$  e  $w$  um string do alfabeto de  $A$ .

Uma  **$(A, q, w)$ -árvore** é uma árvore enraizada em  $q$  definida a seguir:

**Base:**  $w = \epsilon$

A árvore contém apenas o nó raiz  $q$

**Indução:**  $w$  é da forma  $ax$ , tal que  $a \in \Sigma$  e  $x \in \Sigma^*$

Seja  $\delta$  a função de transição do AFN. Se  $\delta(q, a) = \{p_1, p_2, \dots, p_k\}$ , então então a árvore contém o nó  $q$  e  $q$  possui os filhos  $p_1, p_2, \dots, p_k$  que são raízes de uma  $(A, p_i, x)$ -árvore.

## Def.: Árvore de computações possíveis

Seja  $A = (Q, \Sigma, \delta, q_0, F)$  um AFN,  $w \in \Sigma^*$ .

# Árvore de computações possíveis

## Def.: $(A, q, w)$ -árvores

Seja  $(A, q, w)$  uma tripla tal que  $A$  é um AFN,  $q$  um é estado de  $A$  e  $w$  um string do alfabeto de  $A$ .

Uma  **$(A, q, w)$ -árvore** é uma árvore enraizada em  $q$  definida a seguir:

**Base:**  $w = \epsilon$

A árvore contém apenas o nó raiz  $q$

**Indução:**  $w$  é da forma  $ax$ , tal que  $a \in \Sigma$  e  $x \in \Sigma^*$

Seja  $\delta$  a função de transição do AFN. Se  $\delta(q, a) = \{p_1, p_2, \dots, p_k\}$ , então então a árvore contém o nó  $q$  e  $q$  possui os filhos  $p_1, p_2, \dots, p_k$  que são raízes de uma  $(A, p_i, x)$ -árvore.

## Def.: Árvore de computações possíveis

Seja  $A = (Q, \Sigma, \delta, q_0, F)$  um AFN,  $w \in \Sigma^*$ . A **árvore de computações possíveis** de  $A$  com  $w$  é uma  $(A, q_0, w)$ -árvore.