SBVLIFA: Linguagens Formais e Autômatos

Aula 04: Autômatos Finitos Não-Determinísticos com ε-transições

Bacharelado em Ciência da Computação
Prof. Dr. David Buzatto

INSTITUTO FEDERAL DE EDUCAÇÃO, CIÊNCIA E TECNOLOGIA
SÃO PAULO
Campus São João da Boa Vista

Linguagens Regulares

- Linguagens Regulares

Tipo	Classe de Linguagens	Modelo de Gramática	Modelo de Reconhecedor
0	Recursivamente enumeráveis	Irrestrita	Máquina de Turing
1	Sensíveis ao contexto	Sensível ao contexto	Máquina de Turing com fita limitada
2	Livres de contexto	Livre de contexto	Autômato de pilha
3	**Regulares**	**Linear (direita ou esquerda)**	**Autômato finito**

INSTITUTO FEDERAL DE EDUCAÇÃO, CIÊNCIA E TECNOLOGIA
SÃO PAULO
Campus São João da Boa Vista

Linguagens Regulares

- **Relembrando...**
  - **Autômato Finito Determinístico (DFA):** o controle é determinístico, ou seja, **sempre está em um único estado** em qualquer instante;
  - **Autômato Finito Não-Determinístico (NFA):** o controle **pode estar em mais de um estado** em qualquer instante.
    - **Autômato Finito Não-Determinístico com ε-transições (ε-NFA):** É uma outra extensão dos autômatos finitos, que permite que os NFAs façam transições sobre a string vazia. Alguns autores fazem essa divisão entre NFAs e ε-NFAs, entretanto, formalmente, um NFA já tem permissão para realizar transições espontâneas, sendo assim, faremos essa complementação nessa aula.

Linguagens Regulares

Autômatos Finitos Não-Determinísticos com ε-transições

- Os ε-NFAs permitem que haja uma transição espontânea de um estado para outro;
- Essa característica, assim como a diferença entre DFAs e NFAs, não permite que esse autômato reconheça linguagens que não sejam regulares, entretanto, auxilia na capacidade de modelagem da máquina abstrata;
- A transição vazia é representada da mesma forma que as transições que já vimos, entretanto é rotulada pelo símbolo ε;
- Os ε-NFAs são intimamente ligados às expressões regulares, assunto da próxima aula.

Autômatos Finitos Não-Determinísticos com ε-transições

- **Definição Formal:**

$$A = (Q, \Sigma, \delta, q_0, F)$$

- A: autômato finito não-determinístico com ε-transições, uma 5-upla, onde:
  - Q: conjunto finito de estados;
  - Σ: conjunto finito de símbolos de entrada (alfabeto);
  - δ: função de transição, na forma $\delta(q, a) \rightarrow \{p_1, p_2, \dots, p_n\}$, ou seja, $\delta: Q \times \Sigma_\varepsilon \rightarrow \mathcal{P}(Q)$
  - q_0: estado inicial, tal que $q_0 \in Q$
  - F: conjunto de estados finais ou de aceitação, tal que $F \subseteq Q$
- **Obs:** $\Sigma_\varepsilon = \Sigma \cup \{\varepsilon\}$, ou seja, a string vazia não faz parte do alfabeto!

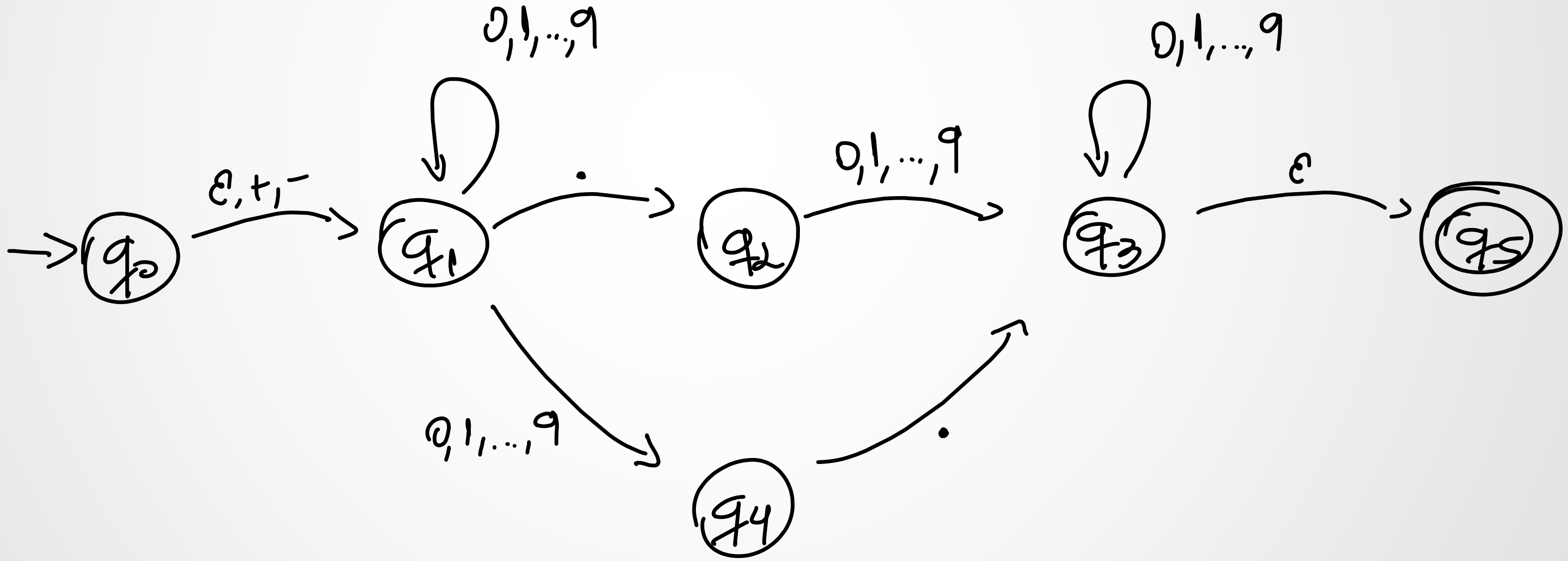
INSTITUTO FEDERAL DE EDUCAÇÃO, CIÊNCIA E TECNOLOGIA
SÃO PAULO
Campus São João da Boa Vista

Autômatos Finitos Não-Determinísticos com ε-transições
Exemplo

- Um ε-NFA que aceita números decimais, consistindo em:
  - Um sinal + ou -, opcional;
  - Uma string de dígitos;
  - Um ponto decimal;
  - Uma outra string de dígitos. Essa string ou a primeira podem ser vazias, mas não ambas.

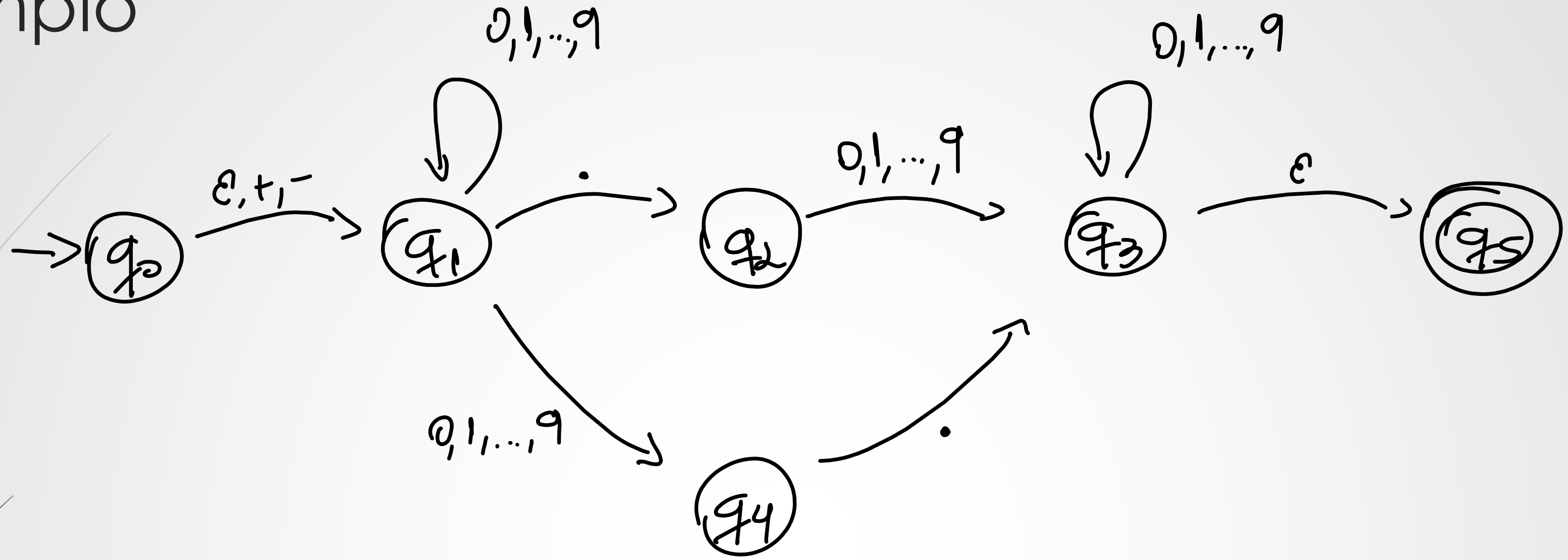
Autômatos Finitos Não-Determinísticos com ε-transições
Exemplo

- **Diagrama de transições:**



Autômatos Finitos Não-Determinísticos com ε-transições

Exemplo



- $A = (Q, \Sigma, \delta, q_0, F)$, onde:
- $Q = \{ q_0, q_1, q_2, q_3, q_4, q_5 \}$
- $\Sigma = \{ ., +, -, 0,1,2,3,4,5,6,7,8,9 \}$
- $F = \{q_5\}$
- δ:

	ε	$+, -$	$.$	$0, 1, 2, \dots, 9$
$\rightarrow q_0$	$\{q_1\}$	$\{q_1\}$	$\emptyset$	$\emptyset$
q_1	$\emptyset$	$\emptyset$	$\{q_2\}$	$\{q_1, q_4\}$
q_2	$\emptyset$	$\emptyset$	$\emptyset$	$\{q_3\}$
q_3	$\{q_5\}$	$\emptyset$	$\emptyset$	$\{q_3\}$
q_4	$\emptyset$	$\emptyset$	$\{q_3\}$	$\emptyset$
$* q_5$	$\emptyset$	$\emptyset$	$\emptyset$	$\emptyset$

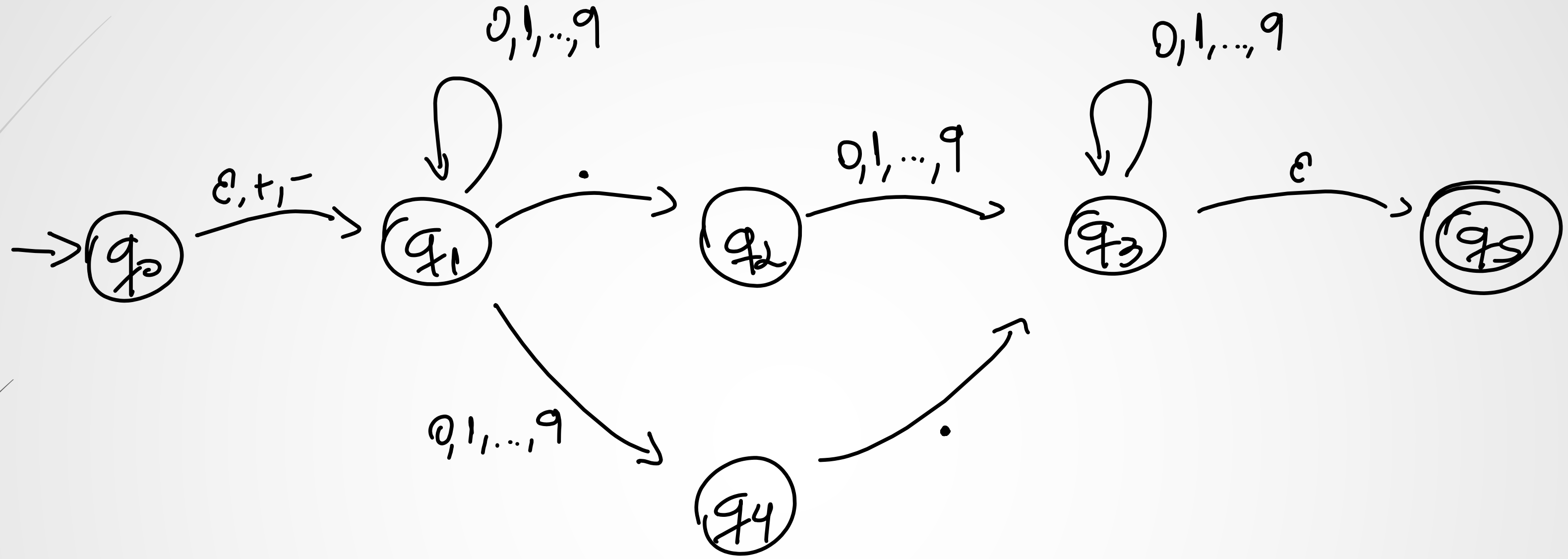
Autômatos Finitos Não-Determinísticos com ε-transições
ε-fechamentos

- Definição central necessária para a extensão da função de transição para strings;
- Informalmente, o **ε-fechamento** de um estado q consiste no conjunto de estados acessíveis a partir de q, seguindo todas as transições rotuladas por ε.
- Formalmente, o ε-fechamento pode ser definido recursivamente pela função $ECLOSE(q)$ da seguinte forma:
  - **Base:** O estado q está em $ECLOSE(q)$.
  - **Indução:** Se o estado p está em $ECLOSE(q)$ e existe uma transição do estado p para o estado r rotulada por ε, então r está em $ECLOSE(q)$, ou seja, se δ é a função de transição do ε-NFA envolvido, e p está em $ECLOSE(q)$, então $ECLOSE(q)$ também contém todos os estados em $\delta(p, \varepsilon)$.

INSTITUTO FEDERAL DE EDUCAÇÃO, CIÊNCIA E TECNOLOGIA
SÃO PAULO
Campus São João da Boa Vista

Autômatos Finitos Não-Determinísticos com ε-transições

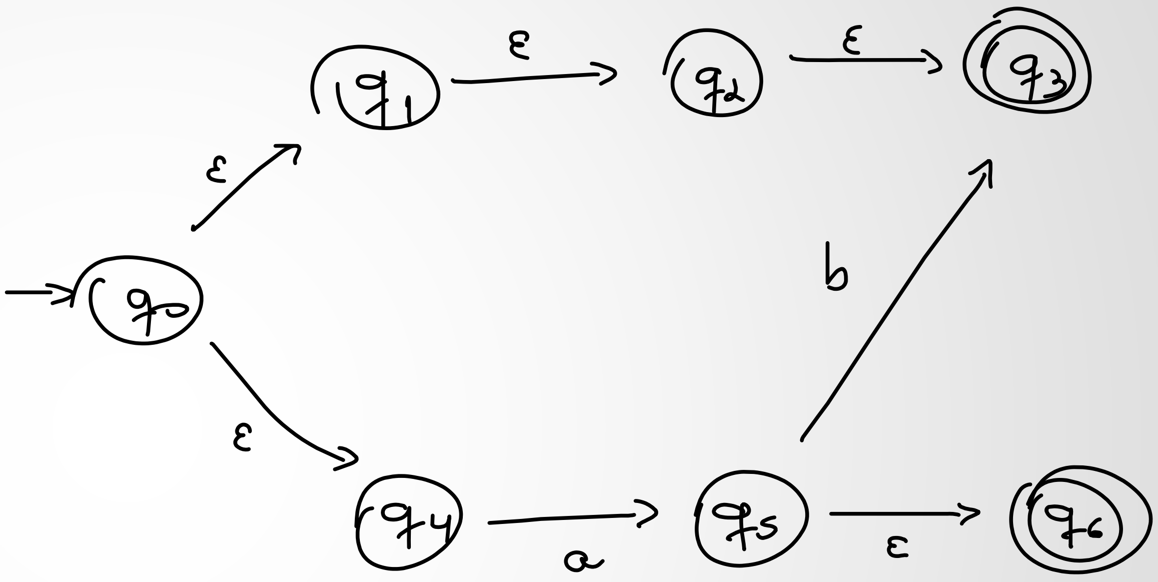
Exemplos de ε-fechamentos



- $ECLOSE(q_0)$: $\{q_0, q_1\}$
- $ECLOSE(q_1)$: $\{q_1\}$
- $ECLOSE(q_2)$: $\{q_2\}$
- $ECLOSE(q_3)$: $\{q_3, q_5\}$
- $ECLOSE(q_4)$: $\{q_4\}$
- $ECLOSE(q_5)$: $\{q_5\}$

Autômatos Finitos Não-Determinísticos com ε-transições

Exemplos de ε-fechamentos



- $ECLOSE(q_0)$: $\{q_0, q_1, q_2, q_3, q_4\}$
- $ECLOSE(q_1)$: $\{q_1, q_2, q_3\}$
- $ECLOSE(q_2)$: $\{q_2, q_3\}$
- $ECLOSE(q_3)$: $\{q_3\}$
- $ECLOSE(q_4)$: $\{q_4\}$
- $ECLOSE(q_5)$: $\{q_5, q_6\}$
- $ECLOSE(q_6)$: $\{q_6\}$

Autômatos Finitos Não-Determinísticos com ε-transições

Extensão da Função de Transição às Strings

- Necessária para tornar exata a noção da linguagem de um ε-NFA;
- Se δ é a função de transição, $\hat{\delta}$ (delta chapéu) é a função de transição estendida;
- Definição:
  - **Base:** $\hat{\delta}(q, \varepsilon) = ECLOSE(q)$

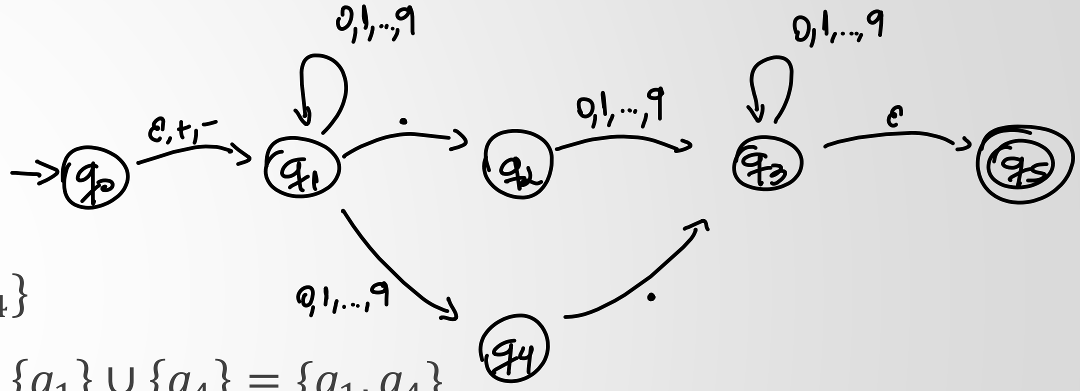
  Se estamos em q e não lemos nada, podemos estar em qualquer estado de $ECLOSE(q)$

  - **Indução:**
    - $w = xa$, onde a é o último símbolo de w, x é o restante de w e $a \in \Sigma$;
    - $\hat{\delta}(q, x) = \{p_1, p_2, \dots, p_k\}$
    - $\bigcup_{i=1}^{k} \delta(p_i, a) = \{r_1, r_2, \dots, r_m\}$
    - $\hat{\delta}(q, w) = \bigcup_{j=1}^{m} ECLOSE(r_j)$

INSTITUTO FEDERAL DE EDUCAÇÃO, CIÊNCIA E TECNOLOGIA
SÃO PAULO
Campus São João da Boa Vista

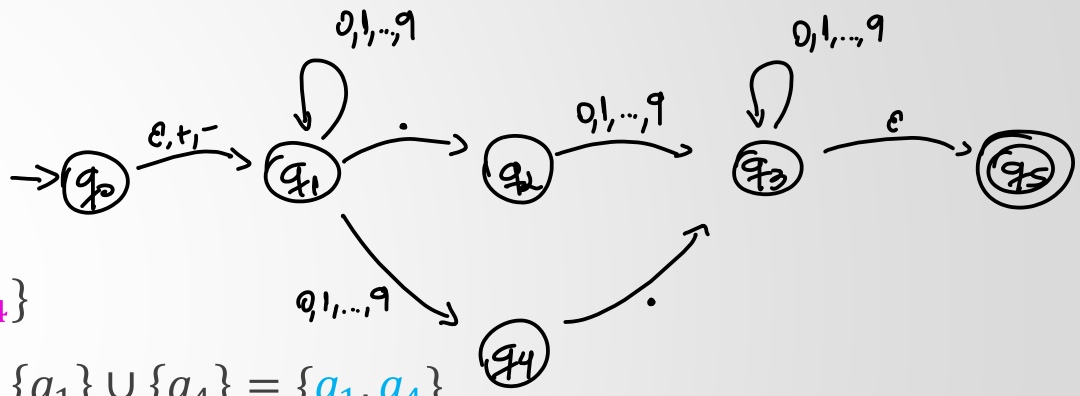
Autômatos Finitos Não-Determinísticos com ε-transições
Extensão da Função de Transição às Strings

- **Exemplo:** Considerando o exemplo do ε-NFA que reconhece números decimais, realizar a computação de $\hat{\delta}(q_0, 5.6)$:
  - $\hat{\delta}(q_0, \varepsilon) = ECLOSE(q_0) = \{q_0, q_1\}$
  - $\hat{\delta}(q_0, 5)$:
    - 1: $\delta(q_0, 5) \cup \delta(q_1, 5) = \emptyset \cup \{q_1, q_4\} = \{q_1, q_4\}$
    - 2: $\hat{\delta}(q_0, 5) = ECLOSE(q_1) \cup ECLOSE(q_4) = \{q_1\} \cup \{q_4\} = \{q_1, q_4\}$
  - $\hat{\delta}(q_0, 5.)$:
    - 1: $\delta(q_1, .) \cup \delta(q_4, .) = \{q_2\} \cup \{q_3\} = \{q_2, q_3\}$
    - 2: $\hat{\delta}(q_0, 5.) = ECLOSE(q_2) \cup ECLOSE(q_3) = \{q_2\} \cup \{q_3, q_5\} = \{q_2, q_3, q_5\}$
  - $\hat{\delta}(q_0, 5.6)$:
    - 1: $\delta(q_2, 6) \cup \delta(q_3, 6) \cup \delta(q_5, 6) = \{q_3\} \cup \{q_3\} \cup \emptyset = \{q_3\}$
    - 2: $\hat{\delta}(q_0, 5.6) = ECLOSE(q_3) = \{q_3, q_5\}$



Autômatos Finitos Não-Determinísticos com ε-transições
Extensão da Função de Transição às Strings

- **Exemplo:** Considerando o exemplo do ε-NFA que reconhece números decimais, realizar a computação de $\hat{\delta}(q_0, 5.6)$:



  - $\hat{\delta}(q_0, \varepsilon) = ECLOSE(q_0) = \{q_0, q_1\}$
  - $\hat{\delta}(q_0, 5)$:
    - 1: $\delta(q_0, 5) \cup \delta(q_1, 5) = \emptyset \cup \{q_1, q_4\} = \{q_1, q_4\}$
    - 2: $\hat{\delta}(q_0, 5) = ECLOSE(q_1) \cup ECLOSE(q_4) = \{q_1\} \cup \{q_4\} = \{q_1, q_4\}$
  - $\hat{\delta}(q_0, 5.)$:
    - 1: $\delta(q_1, .) \cup \delta(q_4, .) = \{q_2\} \cup \{q_3\} = \{q_2, q_3\}$
    - 2: $\hat{\delta}(q_0, 5.) = ECLOSE(q_2) \cup ECLOSE(q_3) = \{q_2\} \cup \{q_3, q_5\} = \{q_2, q_3, q_5\}$
  - $\hat{\delta}(q_0, 5.6)$:
    - 1: $\delta(q_2, 6) \cup \delta(q_3, 6) \cup \delta(q_5, 6) = \{q_3\} \cup \{q_3\} \cup \emptyset = \{q_3\}$
    - 2: $\hat{\delta}(q_0, 5.6) = ECLOSE(q_3) = \{q_3, q_5\}$

Autômatos Finitos Não-Determinísticos com ε-transições
Definição de Linguagem de um ε-NFA

- Dado um ε-NFA $A = (Q, \Sigma, \delta, q_0, F)$, sua linguagem $L(A)$ é definida por:

$$L(A) = \{\, w \mid \hat{\delta}(q_0, w) \cap F \neq \emptyset \,\}$$

- Isto é, $\mathrm{L}(A)$ é o conjunto de strings w em Σ^* tais que $\hat{\delta}(q_0, w)$ contém pelo menos um estado de aceitação. **Se L é $L(A)$ para algum ε-NFA A, dizemos que L é uma linguagem regular, pois um ε-NFA possui um DFA equivalente.**

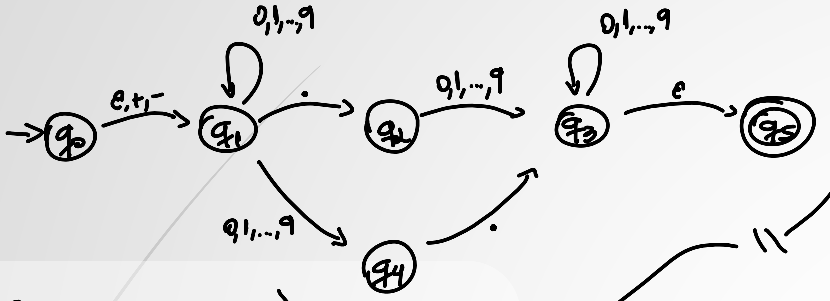
Autômatos Finitos Não-Determinísticos com ε-transições
Equivalência entre DFAs e ε-NFAs (remoção de ε-transições)

- Dado um ε-NFA $E = (Q_E, \Sigma, \delta_E, q_0, F_E)$, construir o DFA $D = (Q_D, \Sigma, \delta_D, q_D, F_D)$ tal que $L(E) = L(D)$
  - Os alfabetos são os mesmos.
- Construção dos outros componentes:
  - Q_D é o conjunto de subconjuntos de Q_E em que todos os estados acessíveis de D são subconjuntos com ε-fechamento de Q_E, isto é, conjuntos $S \subseteq Q_E$ tais que $S = \mathrm{ECLOSE}(S)$
  - $q_D = \mathrm{ECLOSE}(q_0)$
  - F_D é o conjunto de subconjuntos S de Q_E tais que $S \cap F_E \neq \emptyset$, isto é, F_D representa todos os conjuntos de estados de E que incluem pelo menos um estado de aceitação de E;
  - Para cada conjunto $S \subseteq Q_E$ e para cada símbolo de entrada a em Σ:
    a) Seja $S = \{p_1, p_2, \dots, p_k\}$
    b) Calcule $\bigcup_{i=1}^{k} \delta_E\,(p_1, a) = \{r_1, r_2, \dots, r_m\}$
    c) Então $\delta_D(S, a) = \bigcup_{j=1}^{m} \mathrm{ECLOSE}(r_j)$

Autômatos Finitos Não-Determinísticos com ε-transições

Equivalência entre DFAs e ε-NFAs (remoção de ε-transições)

ε-NFA



$E = (Q_E, \Sigma, \delta_E, q_0, F_E)$

$Q_E = \{q_0, q_1, q_2, q_3, q_4, q_5\}$

$F_E = \{q_5\}$

δ_E:

	ε	+,-	.	0,1,...,9
$\rightarrow q_0$	$\{q_1\}$	$\{q_1\}$	$\emptyset$	$\emptyset$
q_1	$\emptyset$	$\emptyset$	$\{q_2\}$	$\{q_1, q_4\}$
q_2	$\emptyset$	$\emptyset$	$\emptyset$	$\{q_3\}$
q_3	$\{q_5\}$	$\emptyset$	$\emptyset$	$\{q_3\}$
q_4	$\emptyset$	$\emptyset$	$\{q_3\}$	$\emptyset$
$*q_5$	$\emptyset$	$\emptyset$	$\emptyset$	$\emptyset$

Construção do DFA $D = (Q_D, \Sigma, \delta_D, q_D, F_D)$

$q_D = \text{ECLOSE}(q_0) = \{q_0, q_1\}$

$Q_D = \{\{q_0, q_1\}, \{q_1\}, \{q_2\}, \{q_1, q_4\}, \{q_3, q_5\}, \{q_2, q_3, q_5\}\}$

$F_D = \{\{q_3, q_5\}, \{q_2, q_3, q_5\}\}$

δ_D:

	+,-	.	0,1,...,9
$\rightarrow \{q_0, q_1\}$	$\{q_1\}$	$\{q_2\}$	$\{q_1, q_4\}$
$\{q_1\}$	$\emptyset$	$\{q_2\}$	$\{q_1, q_4\}$
$\{q_2\}$	$\emptyset$	$\emptyset$	$\{q_3, q_5\}$
$\{q_1, q_4\}$	$\emptyset$	$\{q_2, q_3, q_5\}$	$\{q_1, q_4\}$
$*\{q_3, q_5\}$	$\emptyset$	$\emptyset$	$\{q_3, q_5\}$
$*\{q_2, q_3, q_5\}$	$\emptyset$	$\emptyset$	$\{q_3, q_5\}$

INSTITUTO FEDERAL DE
EDUCAÇÃO, CIÊNCIA E TECNOLOGIA
SÃO PAULO
Campus São João da Boa Vista

Autômatos Finitos Não-Determinísticos com ε-transições

Equivalência entre DFAs e ε-NFAs (remoção de ε-transições)

$D = (Q_D, \Sigma, \delta_D, q_D, F_D)$

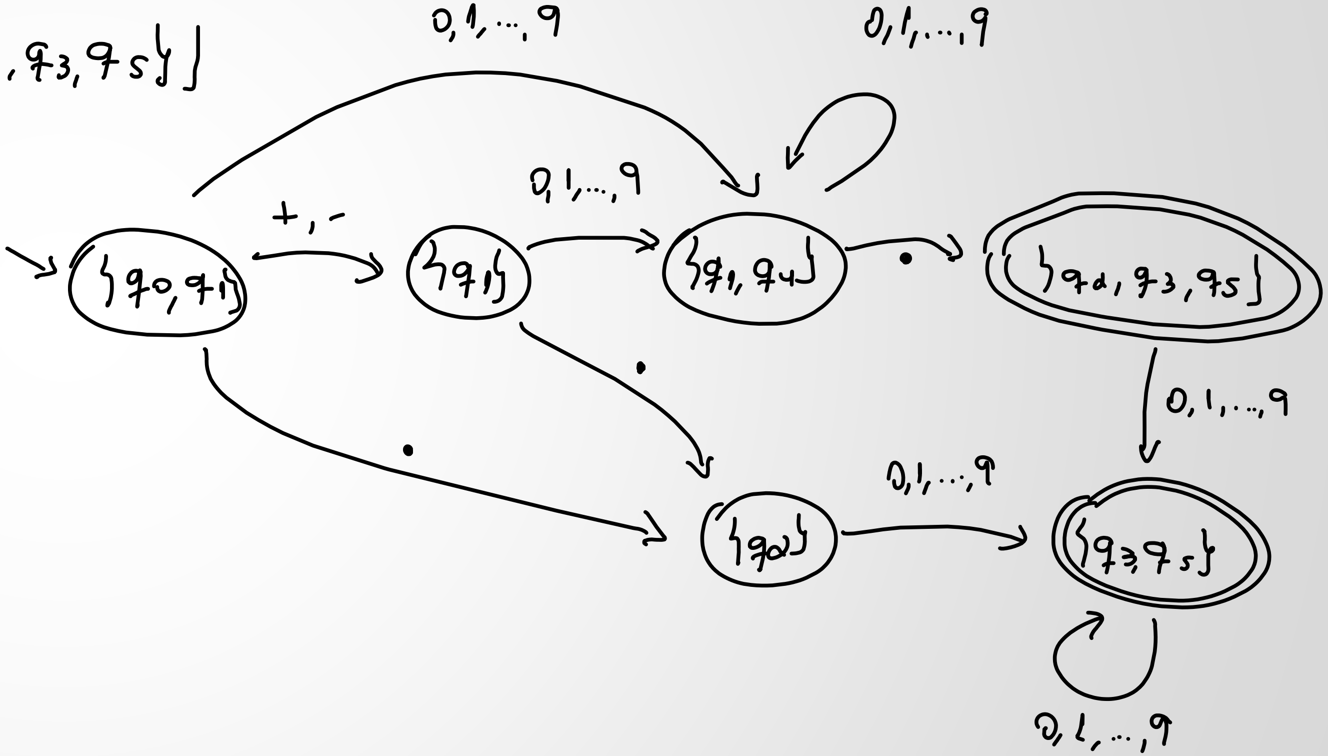
$q_D = ECLOSE(q_0) = \{q_0, q_1\}$

$Q_D = \{\{q_0, q_1\}, \{q_1\}, \{q_2\}, \{q_1, q_4\}, \{q_3, q_5\}, \{q_2, q_3, q_5\}\}$

$F_D = \{\{q_3, q_5\}, \{q_2, q_3, q_5\}\}$

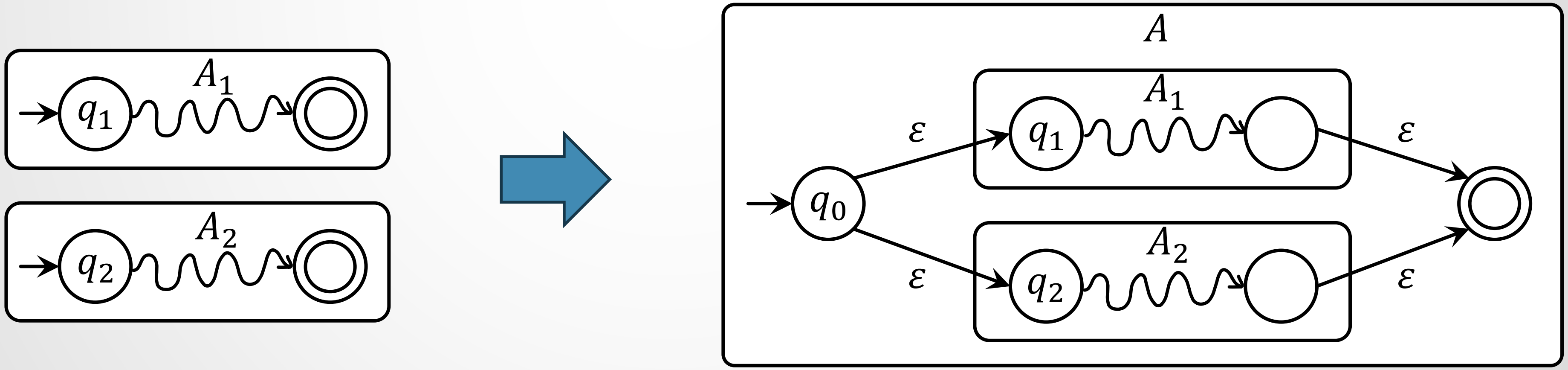
δ_D:

	+, -	.	0, 1, ..., 9
$\rightarrow \{q_0, q_1\}$	$\{q_1\}$	$\{q_2\}$	$\{q_1, q_4\}$
$\{q_1\}$	$\emptyset$	$\{q_2\}$	$\{q_1, q_4\}$
$\{q_2\}$	$\emptyset$	$\emptyset$	$\{q_3, q_5\}$
$\{q_1, q_4\}$	$\emptyset$	$\{q_2, q_3, q_5\}$	$\{q_1, q_4\}$
$* \{q_3, q_5\}$	$\emptyset$	$\emptyset$	$\{q_3, q_5\}$
$* \{q_2, q_3, q_5\}$	$\emptyset$	$\emptyset$	$\{q_3, q_5\}$



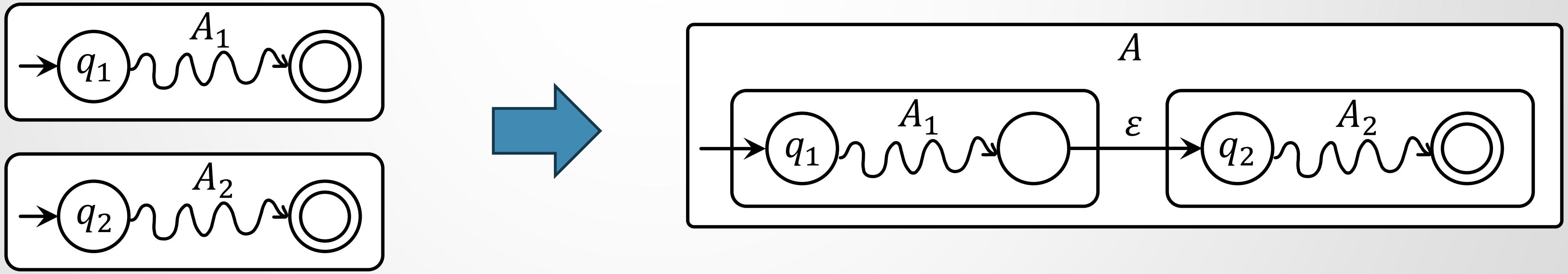
Teoremas de Fechamento Sobre as Operações Regulares

- **Teorema 1:** Se L_1 e L_2 são linguagens regulares, então uma linguagem $L = L_1 \cup L_2$ também é regular.
- **Prova:** dados os autômatos finitos $A_1 = (Q_1, \Sigma, \delta_1, q_1, F_1)$ que reconhece L_1 e $A_2 = (Q_2, \Sigma, \delta_2, q_2, F_2)$ que reconhece L_2, construir $A = (Q, \Sigma, \delta, q_0, F)$ que reconhece L. $\square$



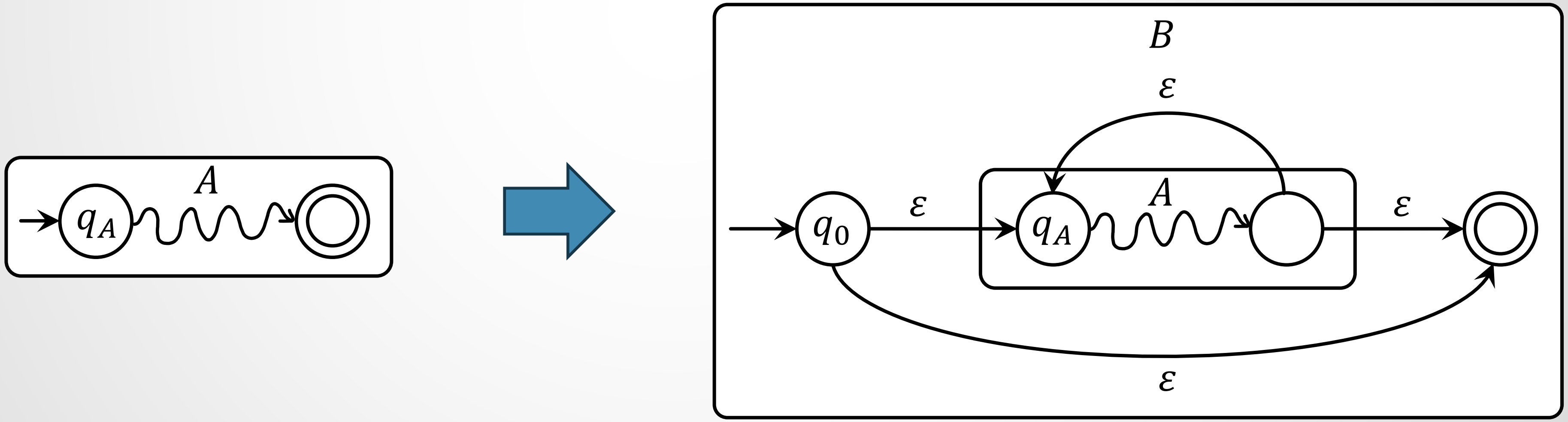
Teoremas de Fechamento Sobre as Operações Regulares

- **Teorema 2:** Se L_1 e L_2 são linguagens regulares, então uma linguagem $L = L_1L_2$ também é regular.
- **Prova:** dados os autômatos finitos $A_1 = (Q_1, \Sigma, \delta_1, q_1, F_1)$ que reconhece L_1 e $A_2 = (Q_2, \Sigma, \delta_2, q_2, F_2)$ que reconhece L_2, construir $A = (Q, \Sigma, \delta, q_1, F_2)$ que reconhece L. $\square$



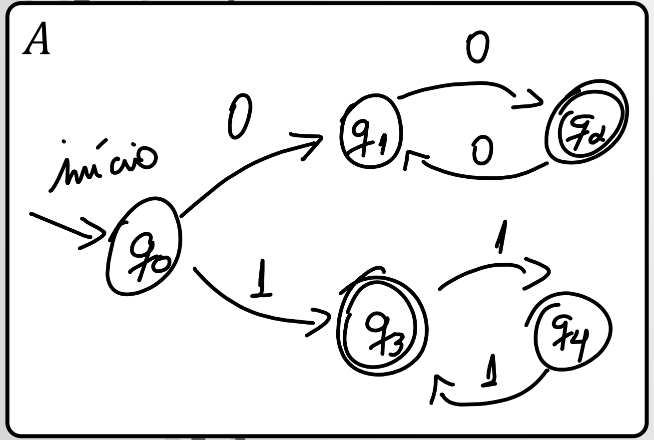
Teoremas de Fechamento Sobre as Operações Regulares

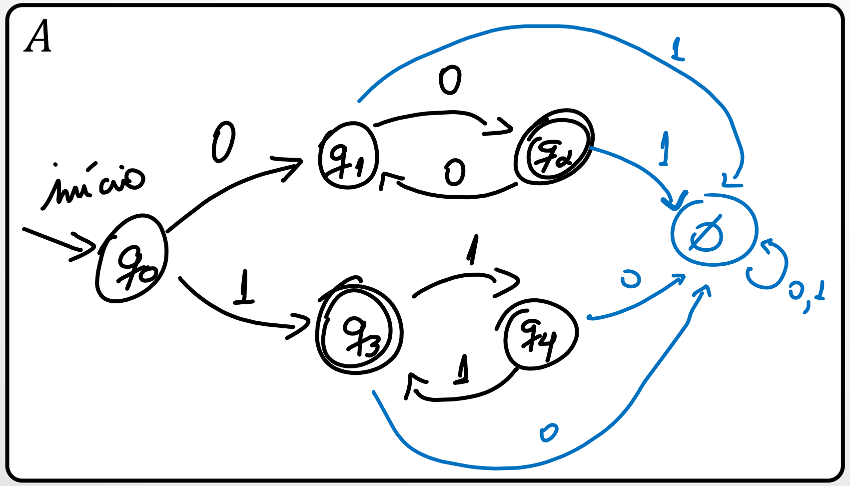
- **Teorema 3:** Se L é uma linguagem regular, L^* também é regular.
- **Prova:** dado o autômato finito $A = (Q_A, \Sigma, \delta_A, q_A, F_A)$ que reconhece L, construir $B = (Q, \Sigma, \delta, q_0, F)$ que reconhece L^*. $\square$

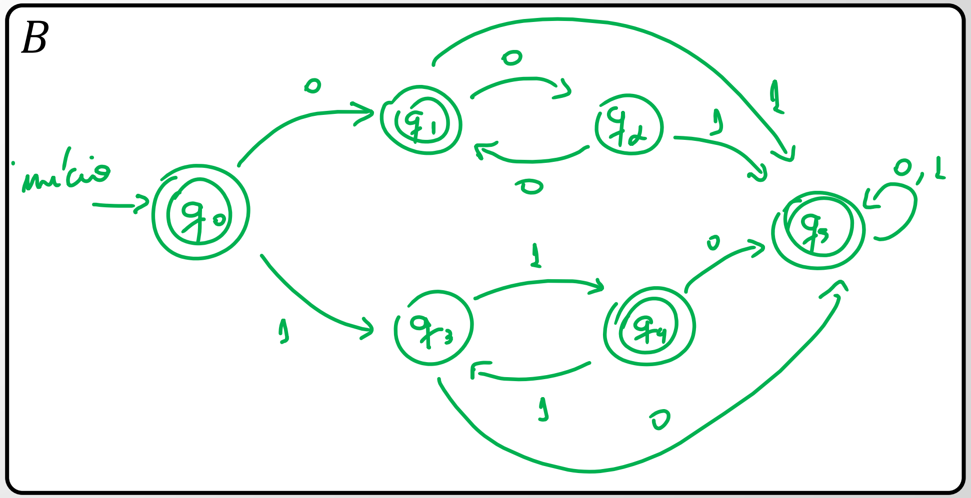


Teoremas de Fechamento Sobre as Operações Regulares

- **Teorema 4:** Se L é uma linguagem regular, seu complemento $\overline{L}$ também é regular.
- **Prova:** dado o autômato finito $A = (Q, \Sigma, \delta, q_0, F_A)$ que reconhece L, construir $B = (Q, \Sigma, \delta, q_0, F_B)$ que reconhece $\overline{L}$. Primeiramente completamos a especificação de A caso haja necessidade, ou seja, criamos um novo estado, chamado de estado nulo (ou sumidouro) e, a partir de cada estado existente, criamos novas transições sobre os símbolos do alfabeto faltantes para o estado nulo. Posteriormente, todos os estados de aceitação (finais) de A deverão tornar-se estados de não aceitação e, todos os estados de não aceitação de A deverão tornar-se estados de aceitação. Sendo assim, toda string w aceita por A, ou seja, $\hat{\delta}(q_0, w) \cap F_A \neq \emptyset$, não pode ser aceita por B e, toda string x aceita por B, ou seja, $\hat{\delta}(q_0, x) \cap F_B \neq \emptyset$, não pode ser aceita por A. $\square$







Teoremas de Fechamento Sobre as Operações Regulares

- **Teorema 5:** Se L_1 e L_2 são linguagens regulares, então uma linguagem $L = L_1 \cap L_2$ também é regular.
- **Prova:** pela lei de De Morgan, $\overline{L} = \overline{L_1 \cap L_2} = \overline{L_1} \cup \overline{L_2}$. Pelo **Teorema 1**, sabemos que a união de duas linguagens regulares é também regular e, pelo **Teorema 4**, também sabemos que o complemento de uma linguagem regular é regular. Sendo assim, como L_1 é regular, $\overline{L_1}$ também o é, assim como L_2, sendo regular, $\overline{L_2}$ também o é. Dado que pela lei de De Morgan $\overline{L} = \overline{L_1} \cup \overline{L_2}$, L também é regular, pois $L = \overline{\overline{L}}$. $\square$

Autômatos Finitos Não-Determinísticos com ε-transições
Exercícios Escritos

Exercício e4.1: Considerando o ε-NFA a seguir:

a) Calcule o ε-fechamento de cada estado;

b) Forneça todas as strings de comprimento menor ou igual a três aceitas pelo autômato;

c) Converta o ε-NFA em DFA.

	ε	a	b	c
$\rightarrow p$	$\emptyset$	$\{p\}$	$\{q\}$	$\{r\}$
q	$\{p\}$	$\{q\}$	$\{r\}$	$\emptyset$
$* r$	$\{q\}$	$\{r\}$	$\emptyset$	$\{p\}$

Autômatos Finitos Não-Determinísticos com ε-transições
Exercícios Escritos

Exercício e4.2: Considerando o ε-NFA a seguir:

a) Calcule o ε-fechamento de cada estado;

b) Forneça todas as strings de comprimento menor ou igual a três aceitas pelo autômato;

c) Converta o ε-NFA em DFA.

	ε	a	b	c
$\rightarrow p$	$\{q, r\}$	$\emptyset$	$\{q\}$	$\{r\}$
q	$\emptyset$	$\{p\}$	$\{r\}$	$\{p, q\}$
$* r$	$\emptyset$	$\emptyset$	$\emptyset$	$\emptyset$

Autômatos Finitos Não-Determinísticos com ε-transições
Exercícios Escritos

Exercício e4.3: Para cada linguagem abaixo, defina formalmente o seu respectivo ε-NFA e apresente o diagrama de transições:

a) $L = \{ a^i b^j c^k \mid i \geq 0, j \geq 0, k \geq 0 \}$

b) O conjunto de strings que consistem em 01 repetido uma ou mais vezes ou 010 repetido uma ou mais vezes.

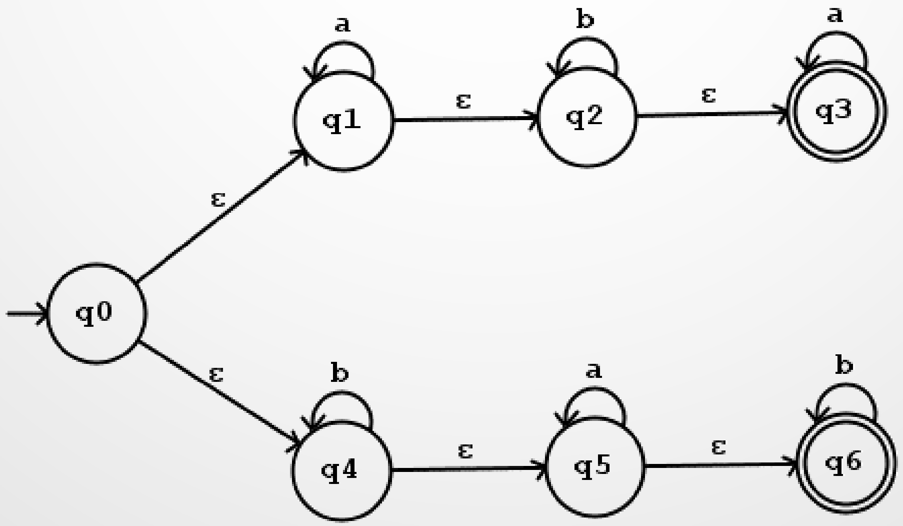
Exercício e4.4: Para cada item do exercício anterior, construa um DFA equivalente aplicando a remoção de ε-transições.

Autômatos Finitos Não-Determinísticos com ε-transições
Exercícios Escritos

Exercício e4.5: Considerando o ε-NFA a seguir, converta-o em um DFA equivalente.

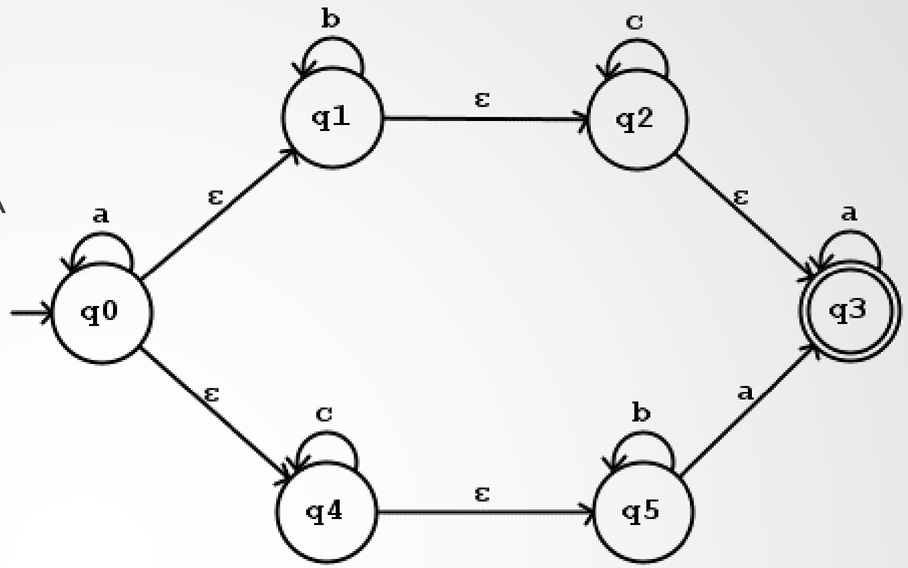
	ε	a	b
$\rightarrow q_0$	$\{q_1\}$	$\{q_0\}$	$\emptyset$
q_1	$\{q_2\}$	$\emptyset$	$\{q_1\}$
q_2	$\{q_3\}$	$\{q_2\}$	$\emptyset$
$* q_3$	$\emptyset$	$\emptyset$	$\{q_3\}$

Exercício e4.6: Considerando o ε-NFA a seguir, converta-o em um DFA equivalente.

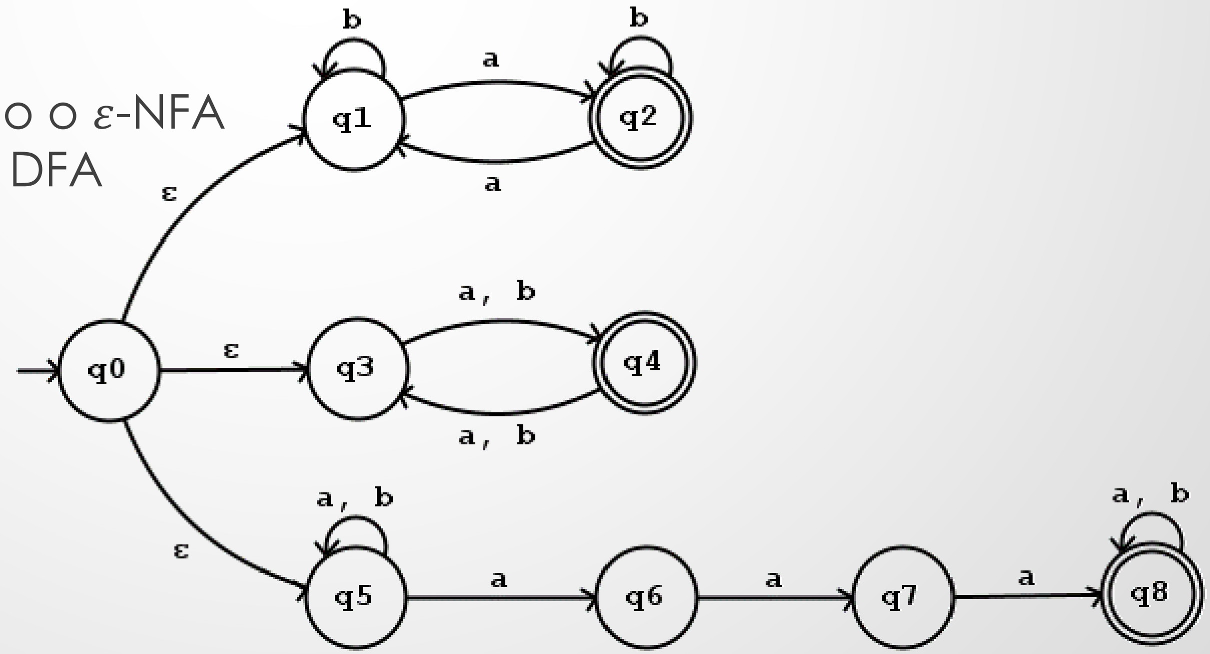


Autômatos Finitos Não-Determinísticos com ε-transições
Exercícios Escritos

Exercício e4.7: Considerando o ε-NFA a seguir, converta-o em um DFA equivalente.



Exercício e4.8: Considerando o ε-NFA a seguir, converta-o em um DFA equivalente.



Bibliografia

HOPCROFT, J. E.; ULLMAN, J. D.; MOTWANI, R. **Introdução à Teoria de Autômatos, Linguagens e Computação**. 2. ed. Rio de Janeiro: Elsevier, 2002. 560 p.

RAMOS, M. V. M.; JOSÉ NETO, J.; VEGA, I. S. **Linguagens Formais: Teoria, Modelagem e Implementação**. Porto Alegre: Bookman, 2009. 656 p.

SIPSER, M. **Introdução à Teoria da Computação**. 2. ed. São Paulo: Cengage Learning, 2017. 459 p.