INSTITUTO FEDERAL
Paraná
Campus Pinhais

Linguagens formais e autômatos

Autômatos finitos não determinísticos

Gabriel V C Candido
gabriel.candido@ifpr.edu.br

Instituto Federal do Paraná - Pinhais

Sumário

Sumário

Autômatos finitos determinísticos (AFD)

Autômatos finitos não determinísticos (AFN)

Autômatos finitos não determinísticos com transições λ

Autômatos finitos determinísticos (AFD)

Definição

Um *autômato finito determinístico (AFD)* é uma quíntupla $M = (Q, \Sigma, \delta, q_0, F)$, onde:

- Q: conjunto finito de estados;
- Σ: alfabeto, conjunto finito de símbolos;
- $q_0 \in Q$: é o estado inicial;
- $F \subseteq Q$: é conjunto de estados finais;
- $\delta : Q \times \Sigma \to Q$: é a função total de transição.

Exemplo 1

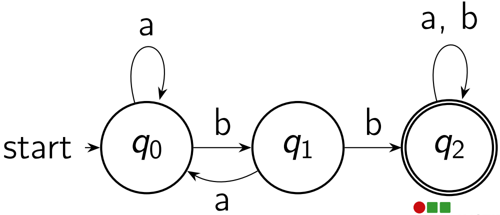
De novo ...

Um AFD que aceita o conjunto de palavras sobre $\{a, b\}$ que contêm a subpalavra bb.

$L(M) = (a+b)^* bb (a+b)^*$.

$$Q = \{q_0, q_1, q_2\}; \Sigma = \{a, b\}; F = \{q_2\};$$
$$M(Q, \Sigma, \delta, q_0, F)$$

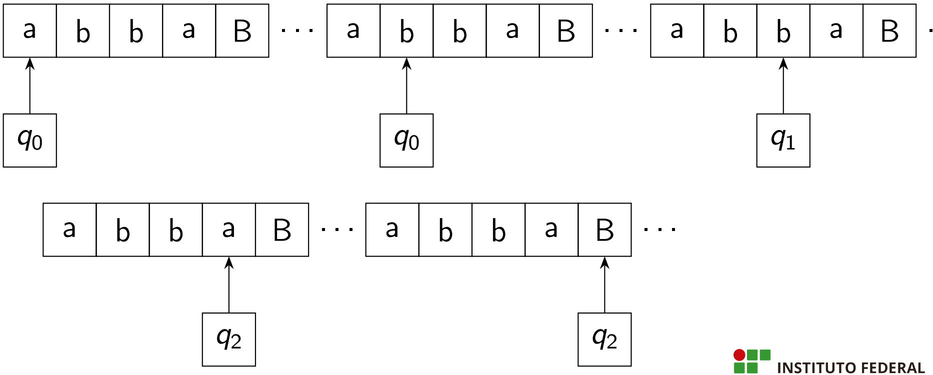
δ	a	b
q_0	q_0	q_1
q_1	q_0	q_2
q_2	q_2	q_2



Exemplo 1

Fita, de novo . . .

$$M : Q = \{q_0, q_1, q_2\}; \Sigma = \{a, b\}; F = \{q_2\};$$
$$\delta(q_0, a) = q_0; \delta(q_0, b) = q_1; \delta(q_1, a) = q_0; \delta(q_1, b) = q_2; \delta(q_2, a) = q_2; \delta(q_2, b) = q_2;$$



Configuração instantânea e ciclo de instrução

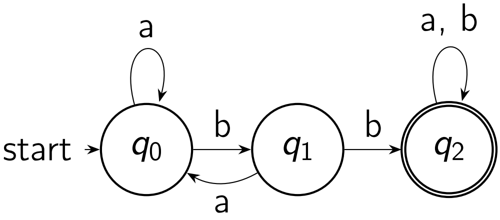
Formalizando a ideia da fita.

Uma configuração instantânea é o par $[q_i, w]$: o estado atual e a subpalavra não processada.

$[q_i, aw] \vdash [q_j, w]$, desde que $\delta(q_i, a) = q_j$.

Exemplo 1

Configuração instantânea e ciclo de instrução



$$[q_0, abba] \vdash [q_0, bba] \vdash [q_1, ba] \vdash [q_2, a] \vdash [q_2, \lambda]$$

$$[q_0, abba] \vdash^* [q_2, \lambda]$$

Aplicação prática de AFD

Árvores trie

$K = \{$`a, alma, asa, barco, brasa, broa, ca, calma, casa, disco`$\}$

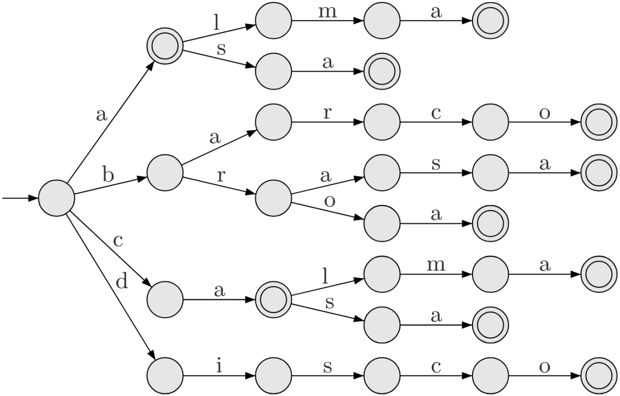


Figura: Fonte: NV 2.2

Sumário

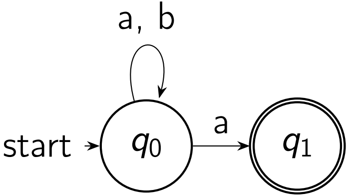
Autômatos finitos determinísticos (AFD)

Autômatos finitos não determinísticos (AFN)

Autômatos finitos não determinísticos com transições λ

INSTITUTO FEDERAL
Paraná
Campus Pinhais

Autômatos finitos não determinísticos (AFN)



Note que:

- Temos uma "indecisão" no estado q_0 (várias computações possíveis);
- Faltam transições para alguns símbolos de Σ em alguns estados.

Autômatos finitos não determinísticos (AFN)

Definição

Um *autômato finito não determinístico (AFN)* é uma quíntupla $M = (Q, \Sigma, \delta, q_0, F)$, onde:

- Q: conjunto finito de estados;
- Σ: alfabeto, conjunto finito de símbolos;
- $q_0 \in Q$: é o estado inicial;
- $F \subseteq Q$: é conjunto de estados finais;
- $\delta : Q \times \Sigma \rightarrow \mathcal{P}(Q)$: é a função de transição.

Exemplo 1

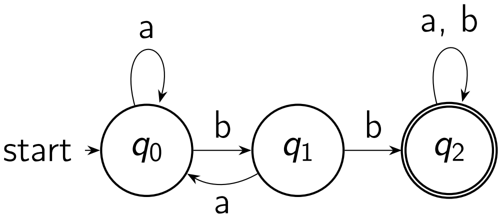
De novo, de novo, em AFD!

Um AFD que aceita o conjunto de palavras sobre $\{a, b\}$ que contêm a subpalavra bb.

$L(M) = (a+b)^* bb(a+b)^*$.

$$M : Q = \{q_0, q_1, q_2\}; \Sigma = \{a, b\}; F = \{q_2\};$$

δ	a	b
q_0	q_0	q_1
q_1	q_0	q_2
q_2	q_2	q_2



Exemplo 1

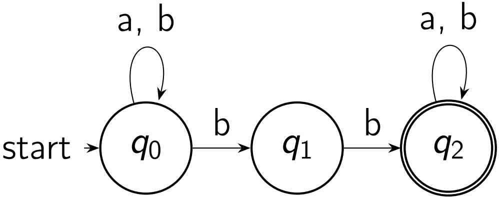
De novo, de novo, mas novo: em AFN!

Um AFN que aceita o conjunto de palavras sobre $\{a, b\}$ que contêm a subpalavra bb.

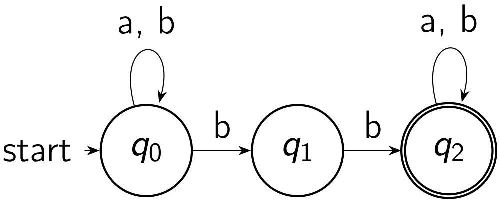
$L(M) = (a + b)^*bb(a + b)^*$.

$$M : Q = \{q_0, q_1, q_2\}; \Sigma = \{a, b\}; F = \{q_2\};$$

δ	a	b
q_0	$\{q_0\}$	$\{q_0, q_1\}$
q_1	$\emptyset$	$\{q_2\}$
q_2	$\{q_2\}$	$\{q_2\}$



Computações possíveis



$$
\begin{aligned}
&[q_0, ababb] \\
&\vdash [q_0, babb] \\
&\vdash [q_0, abb] \\
&\vdash [q_0, bb] \\
&\vdash [q_0, b] \\
&\vdash [q_0, \lambda] \\
&q_0 \notin F
\end{aligned}
$$

$$
\begin{aligned}
&[q_0, ababb] \\
&\vdash [q_0, babb] \\
&\vdash [q_1, abb] \\
&q_1 \notin F
\end{aligned}
$$

$$
\begin{aligned}
&[q_0, ababb] \\
&\vdash [q_0, babb] \\
&\vdash [q_0, abb] \\
&\vdash [q_0, bb] \\
&\vdash [q_1, b] \\
&\vdash [q_2, \lambda] \\
&q_2 \in F
\end{aligned}
$$

Computações possíveis

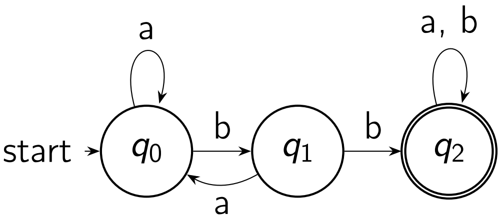
A aceitação de um AFN acontece se **existe pelo menos uma** computação que aceite a palavra.

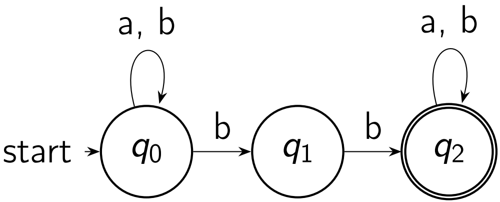
$L(M) = \{w \in \Sigma^* |$ existe uma computação $[q_0, w] \vdash^* [q_j, \lambda], q_j \in F\}$

Exemplo 1

Ninguém aguenta mais . . .

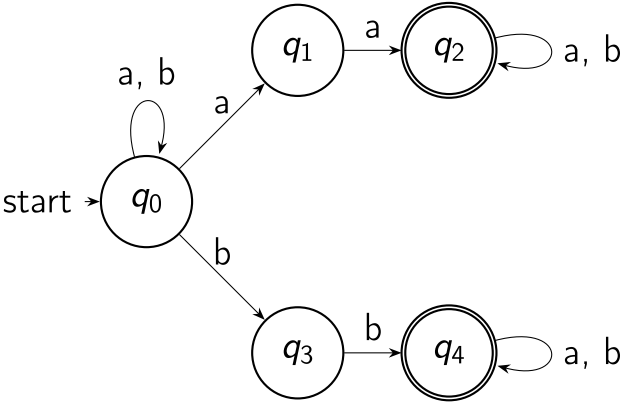
$$L(M) = (a+b)^* bb(a+b)^*.$$





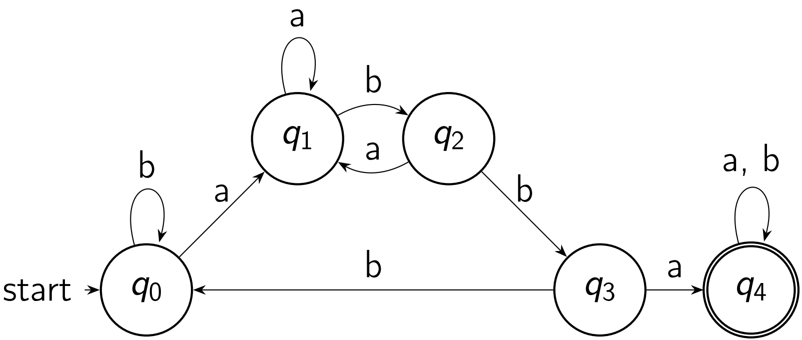
Exemplo 2

AFN que aceita
$(a+b)^*bb(a+b)^* + (a+b)^*aa(a+b)^*$



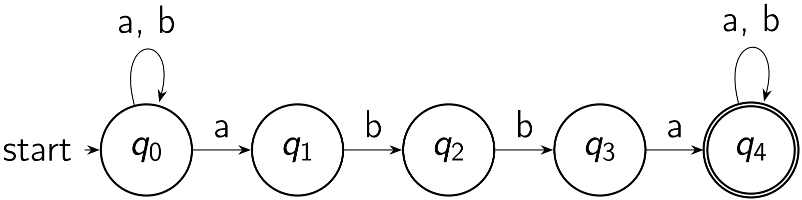
Exemplo 3

AFD que aceita palavras que contêm *abba*.



Exemplo 3

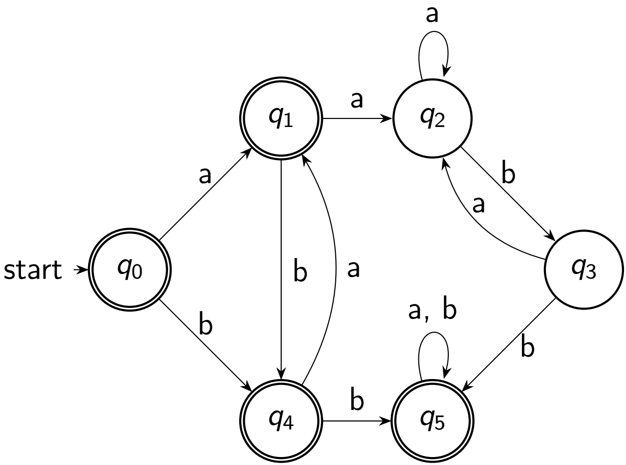
AFN que aceita palavras que contêm *abba*.



Exercício

AFN que aceita $(a+b)^*bb(a+b)^* + (b+ab)^*(a+\lambda)$, palavras que contêm bb ou não contêm aa.

O AFD é:



Sumário

Autômatos finitos determinísticos (AFD)

Autômatos finitos não determinísticos (AFN)

Autômatos finitos não determinísticos com transições λ

INSTITUTO FEDERAL
Paraná
Campus Pinhais

Precisamos consumir um símbolo

- Nos AFD: $\delta : Q \times \Sigma \to Q$
- Nos AFN: $\delta : Q \times \Sigma \to \mathcal{P}(Q)$

Transições λ

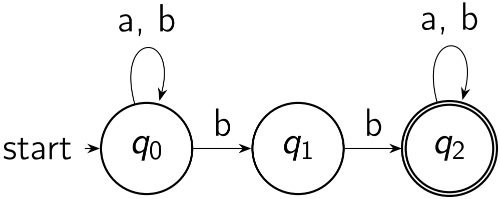
Definição

Um *autômato finito não determinístico com transições* λ *(AFN-*λ*)* é uma quíntupla $M = (Q, \Sigma, \delta, q_0, F)$, onde Q, Σ, q_0, F são como nos AFN. A função de transição é:

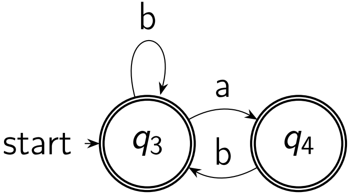
$$\delta : Q \times (\Sigma \cup \lambda) \to \mathcal{P}(Q)$$

Exemplo 4

$M_1 : (a + b)^* bb(a + b)^*$



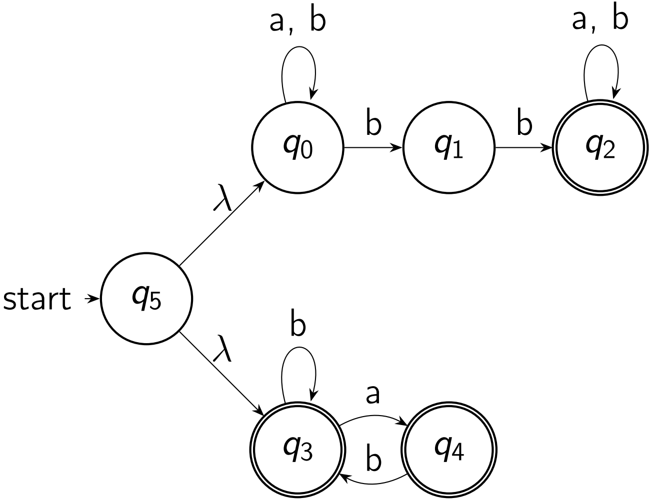
$M_2 : (b + ab)^*(a + \lambda)$:



Exemplo 4

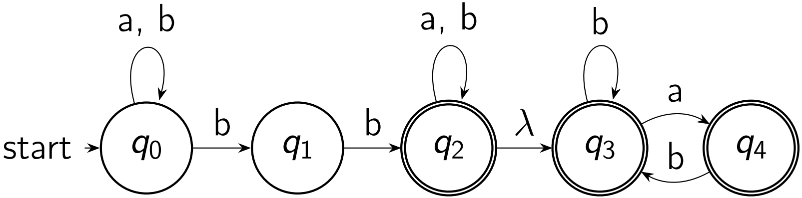
É o exercício anterior

Então o seguinte autômato aceita $L(M_1) \cup L(M_2)$:



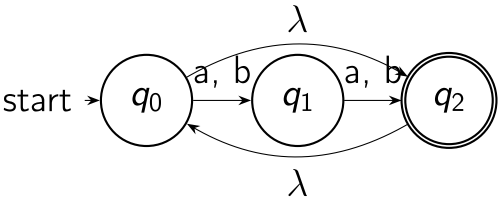
Exemplo 4

Então o seguinte autômato aceita $L(M_1)L(M_2)$:



Exemplo 5

AFN-λ para palavras sobre $\{a, b\}$ de tamanho par:



AFN-λ

Seja $M = (Q, \Sigma, \delta, q_0, F)$ um AFN-λ. Então podemos construir um AFN-λ equivalente M' com as seguintes características:

- O grau de entrada do nó inicial é zero
- M' possui apenas um estado final
- O grau de saída desse estado final é zero

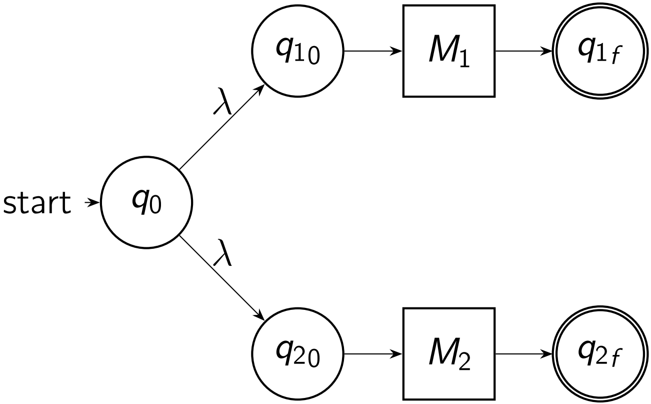
AFN-λ

Podemos construir, um AFN-λ que aceita:

- $L(M_1) \cup L(M_2)$
- $L(M_1)L(M_2)$
- $L(M_1)^*$

AFN-λ

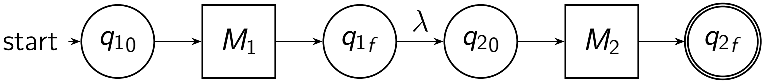
$L(M_1) \cup L(M_2)$:



INSTITUTO FEDERAL
Paraná
Campus Pinhais

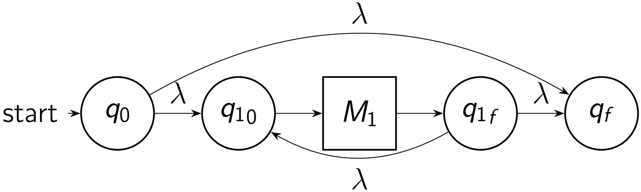
AFN-λ

$L(M_1)L(M_2)$:



AFN-λ

$L(M_1)^*$:



AFN-λ

O que ganhamos com não determinismo e as transições λ?

INSTITUTO FEDERAL
Paraná
Campus Pinhais

Removendo o não determinismo

Definição

O fecho λ de um estado q_i é definido assim:

- **BASE:** $q_i \in$ fecho-$\lambda(q_i)$;
- **PASSO RECURSIVO:** Sejam $q_j \in$ fecho-$\lambda(q_i)$. Se $q_k \in \delta(q_j, \lambda)$ então $q_k \in$ fecho-$\lambda(q_i)$.
- **FECHO:** $q_j \in$ fecho-$\lambda(q_i)$ apenas se pode ser obtido de q_i por um número finito de aplicações do passo recursivo.

A ideia do passo é: quais estados eu consigo chegar usando λ?

Removendo o não determinismo

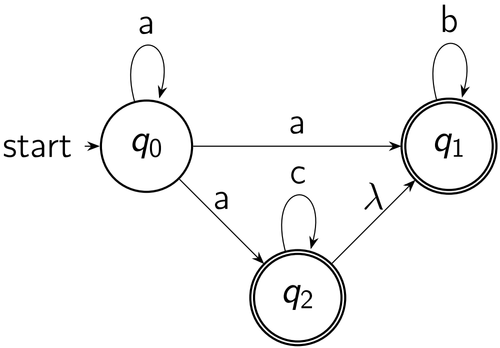
Agora queremos remover as transições λ. Vamos criar uma função t que mapeia todos os estados que podemos chegar, conforme o fecho-λ:

Definição

$$t(q_i, a) = \bigcup_{q_j \in \text{fecho-}\lambda(q_i)} \text{fecho-}\lambda(\delta(q_j, a))$$

Exemplo 6

$a^+ c^* b^*$:



δ	a	b	c	λ
q_0	$\{q_0, q_1, q_2\}$	$\emptyset$	$\emptyset$	$\emptyset$
q_1	$\emptyset$	$\{q_1\}$	$\emptyset$	$\emptyset$
q_2	$\emptyset$	$\emptyset$	$\{q_2\}$	$\{q_1\}$

Exemplo 6

δ	a	b	c	λ
q_0	$\{q_0, q_1, q_2\}$	$\emptyset$	$\emptyset$	$\emptyset$
q_1	$\emptyset$	$\{q_1\}$	$\emptyset$	$\emptyset$
q_2	$\emptyset$	$\emptyset$	$\{q_2\}$	$\{q_1\}$

t	a	b	c
q_0	$\{q_0, q_1, q_2\}$	$\emptyset$	$\emptyset$
q_1	$\emptyset$	$\{q_1\}$	$\emptyset$
q_2	$\emptyset$	$\{q_1\}$	$\{q_1, q_2\}$

Removendo o não determinismo

Convertendo o AFN-λ para AFD:

1. Seu estado inicial é o fecho-$\lambda(q_0)$;
2. Para cada símbolo de Σ, criar os estados necessários e transições conforme a função t

Removendo o não determinismo

Algoritmo de AFN-λ para AFD

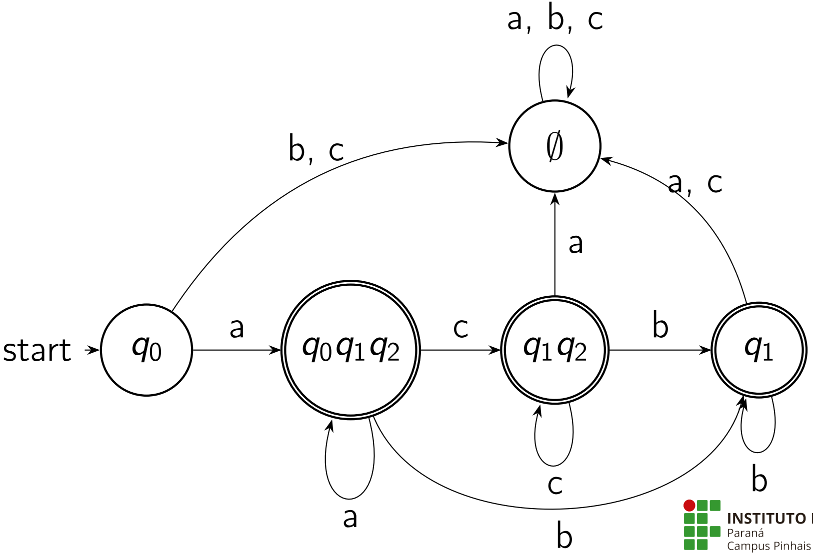
Entrada: Um AFN-λ $M = (Q, \Sigma, \delta, q_0, F)$ e a função de transição de entrada de M, t. A saída é DM, o AFD equivalente à M.

Inicialize $Q' := \{fecho\text{-}\lambda(q_0)\}$
Repita
    SE existe um nodo $X \in Q'$ e um símbolo $a \in \Sigma$
    tal que não há um arco saindo de X rotulado a, ENTÃO
        Seja $Y = \bigcup_{q_i \in X} t(q_i, a)$
        SE $Y \notin Q'$ ENTÃO $Q' := Q' \cup \{Y\}$
        Adicione um arco de X para Y rotulado a
    SENÃO feito:= true
ATÉ feito;
$F' := \{x \in Q' \mid x$ contém um elemento $q_i \in F\}$

Figura: Fonte: Prof. Castilho

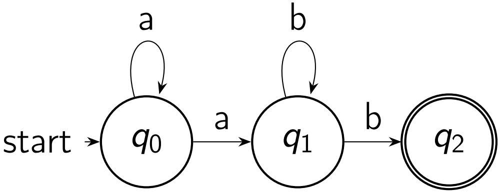
Exemplo 6

AFD para $a^+c^*b^*$:



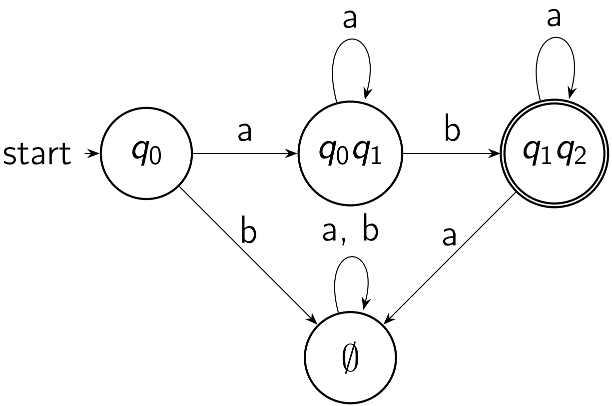
Exemplo 7

Converter este AFN que reconhece a^+b^+:



Exemplo 7

AFD que reconhece a^+b^+:



Concluindo . . .

Podemos converter um AFN-λ em um AFD.

Portanto, o não determinismo e as transições λ **não aumentam o poder computacional**.

Podemos provar isso matematicamente, mostrando que se uma computação chega em um determinado estado em um AFN-λ, ela vai chegar no estado correspondente após a conversão para AFD. (podemos fazer isso somente com o que já definimos até aqui)

Concluindo . . .

Podemos construir e desenhar um AFN-λ (mais fácil).

E usar o algoritmo (um computador) para converter para AFD (implementável).

Concluindo ...

- AFD's são um tipo especial de AFN's
- AFD's são um tipo especial de AFN-λ's
- AFN-λ's podem ser reduzidos para AFD's