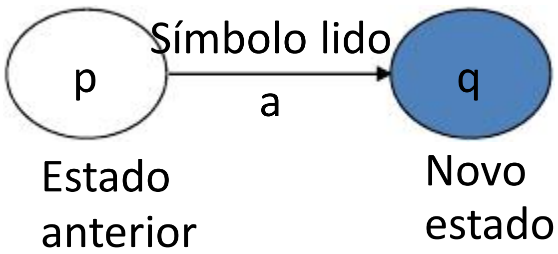
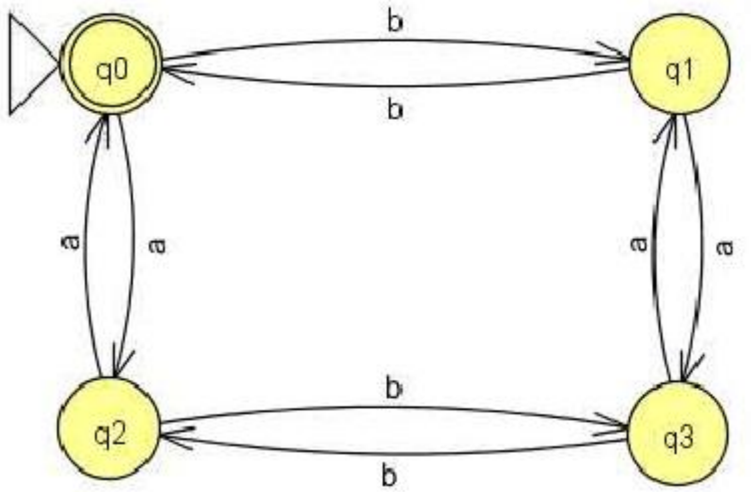
LFA

AFND: com e sem movimentos vazios

Adaptado de C. O. Júnior

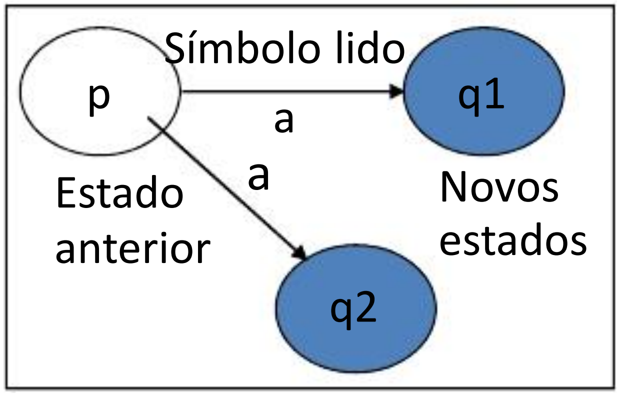
- Autômato finito determinístico
  - Para cada entrada (símbolo) existe **um e somente** um estado ao qual o autômato pode transitar a partir de seu estado atual e do símbolo lido





Na aula de hoje...

- Autômato finito não-determinístico com e sem movimentos vazios
    - O autômato tem o poder de estar em vários estados ao mesmo tempo



No estado **p** ao ler o símbolo **a** assume **q1** e **q2** como novos estados atuais

AFND: aceitação e rejeição

• Diz-se que um autômato finito não-determinístico **aceita** uma cadeia de entrada quando houver alguma sequência de movimentos que o leve da configuração inicial para uma configuração final.

•Diferentemente do autômato finito determinístico, em que essa sequência, se existir, é única para cada cadeia de entrada, no caso do autômato finito não-determinístico é possível que exista mais de uma sequência que satisfaça a essa condição para uma dada cadeia de entrada.

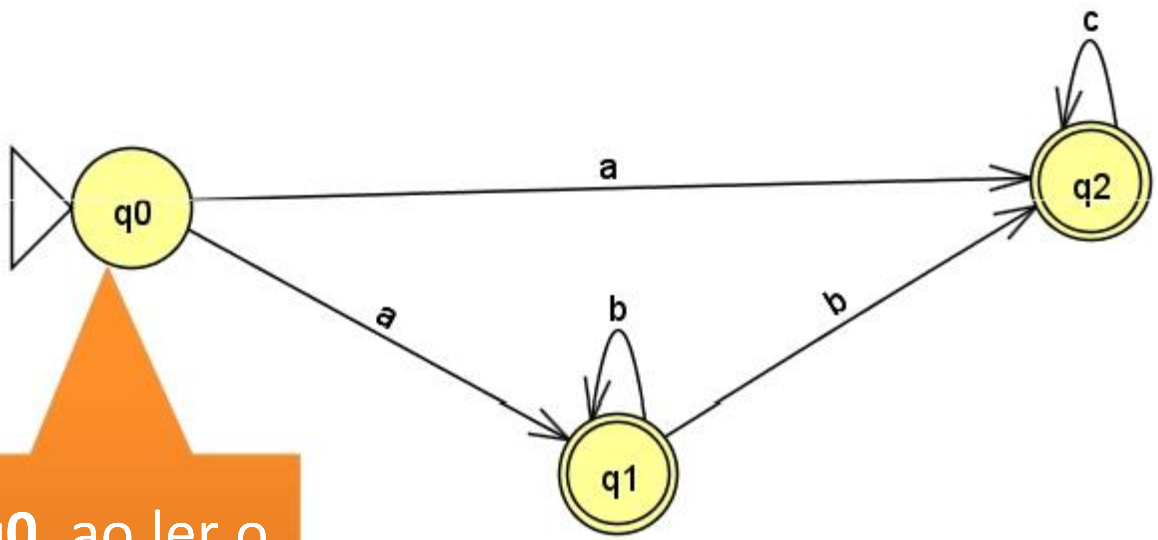
•Sempre que o autômato finito não-determinístico se deparar com mais de uma possibilidade de movimentação, é feita a escolha (arbitrária) de uma das alternativas; em caso de insucesso no reconhecimento, deve-se considerar sucessivamente cada uma das demais alternativas ainda não consideradas, até o seu esgotamento; persistindo o insucesso, e esgotadas as alternativas, diz-se que o autômato **rejeita** a cadeia.

Aceitação e rejeição de cadeias em AF's

	Dada uma cadeia de entrada, ele:	Aceita a cadeia de entrada se:	Rejeita a cadeia de entrada se:
AFD	Executa uma única sequência de movimentos.	Pára em uma configuração final.	Pára em uma configuração não-final.
AFND	Pode executar várias sequências distintas de movimentos.	Pára em uma configuração final.	Pára sem conseguir atingir nenhuma configuração final.

Exemplo de AFND

- Reconhece a linguagem ab*c*



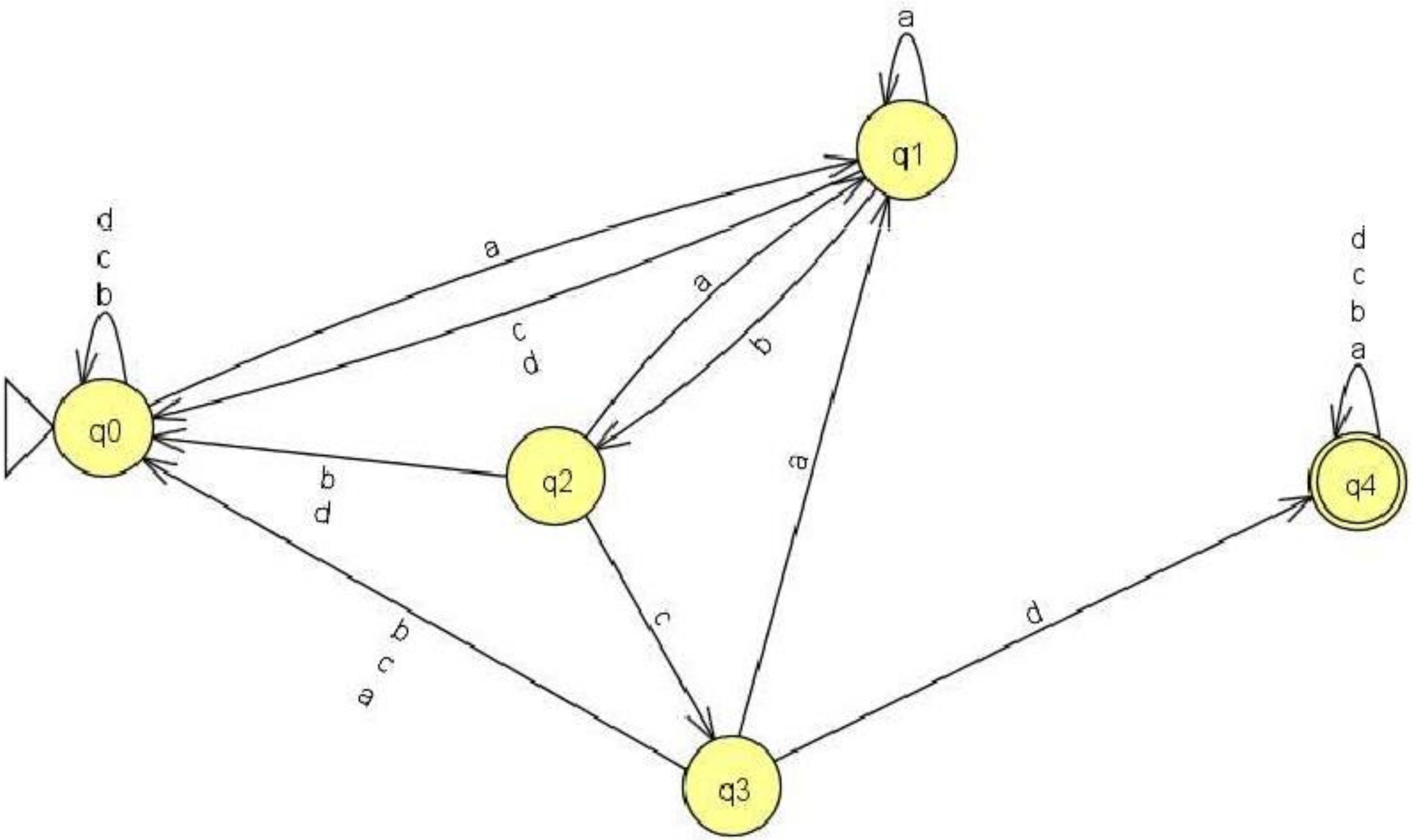
No estado **q0**, ao ler o símbolo **a**, o autômato vai para os estados **q1** e **q2**

AFD versus AFND

- os autômatos finitos não-determinísticos, em certos casos, podem mostrar-se mais simples de serem analisados do que as correspondentes versões determinísticas

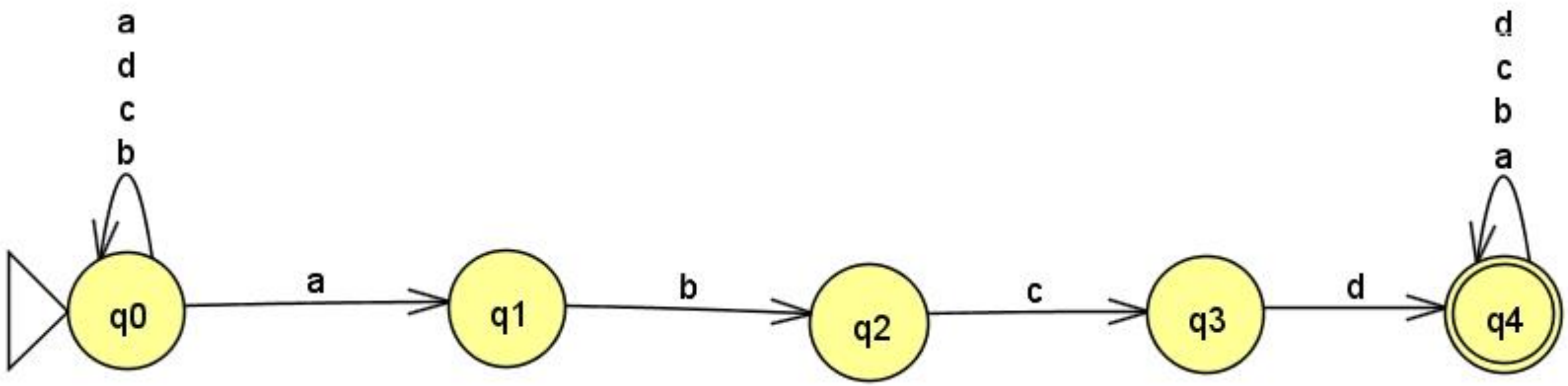
AFD versus AFND

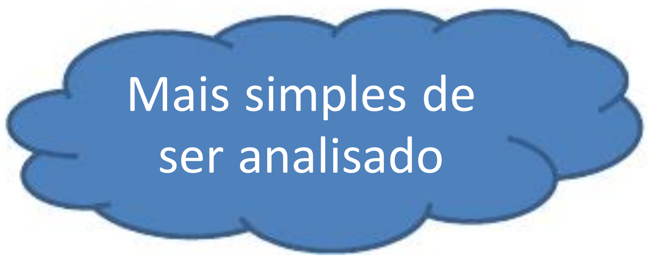
- AFD: Reconhece a linguagem **(a|b|c|d)*abcd (a|b|c|d)***



AFD versus AFND

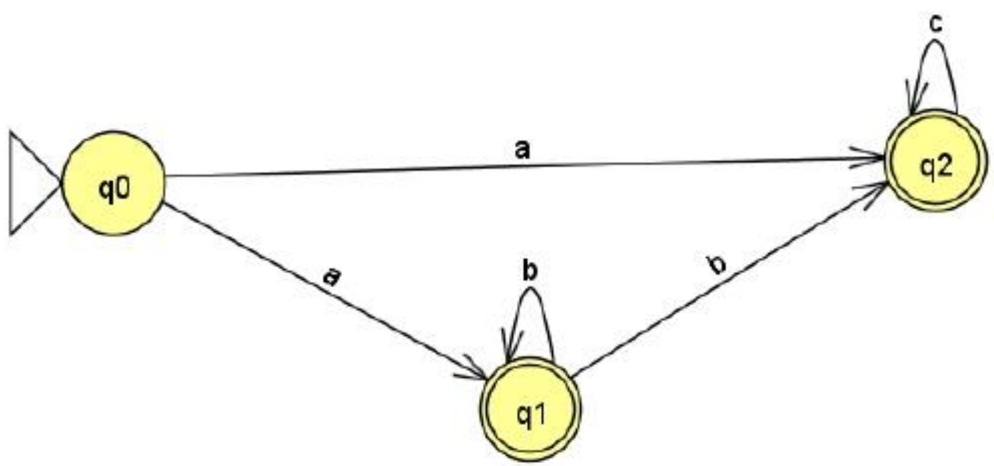
- AFND: Reconhece a mesma linguagem **(a|b|c|d)*abcd (a|b|c|d)***





AFND: notação tabular

- Cada linha da tabela representa um estado distinto q do autômato, e cada coluna é associada a um elemento distinto de seu alfabeto © de entrada. As células correspondentes à intersecção de cada linha com cada coluna são preenchidas com o elemento (conjunto) determinado pela função de transição



notação tabular

δ	a	b	c
q0	{q1,q2}		
*q1		{q1,q2}	
*q2			{q2}

- Construa AFNs para as seguintes linguagens:
  1. $\{0,1\}^*\{1010\}$, $\Sigma = \{0,1\}$
  2. $\{0,00\}\{11\}^*$, $\Sigma = \{0,1\}$
  3. $\{a,b,c\}^*\{abc\}\{a,b,c\}^*$, $\Sigma = \{a,b,c\}$
  4. $\{a,b,c\}^* \{abc,bca\}$, $\Sigma = \{a,b,c\}$

AFND com **movimentos vazios** (transições em vazio) AFNDε

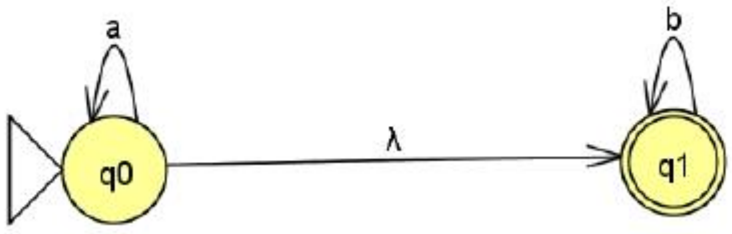
AFND com movimentos vazios (transições em vazio) **AFND ε**

•AFND que apresentam transições em vazio são aqueles que admitem transições de um estado para outro com **ε** **ou** λ, além das transições normais, que utilizam os símbolos do alfabeto de entrada.

•Transições em vazio podem ser executadas sem que seja necessário consultar o símbolo corrente da fita de entrada, e sua execução nem sequer causa o deslocamento do cursor de leitura.

AFNDε

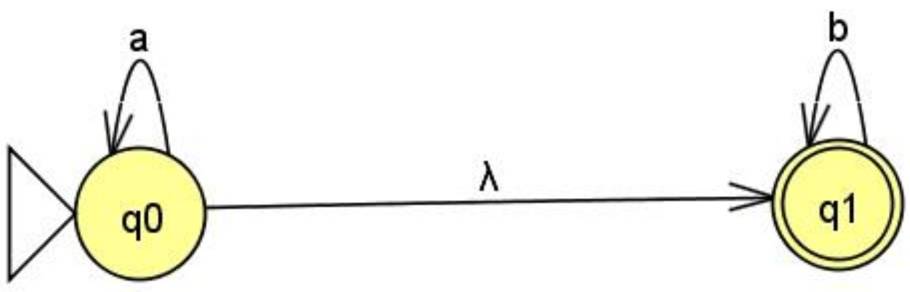
• Quando um autômato transita em vazio, isso significa que ele muda de estado sem consultar a cadeia de entrada.



•Sempre que ocorrer a simultaneidade entre alguma transição em vazio e outras transições (vazias ou não) com origem em um mesmo estado, isso acarreta a necessidade de uma escolha arbitrária da transição a ser aplicada na respectiva configuração, e isso, por sua vez, caracteriza a manifestação de um não-determinismo.

AFNDε

- Reconhece a cadeia a*b*



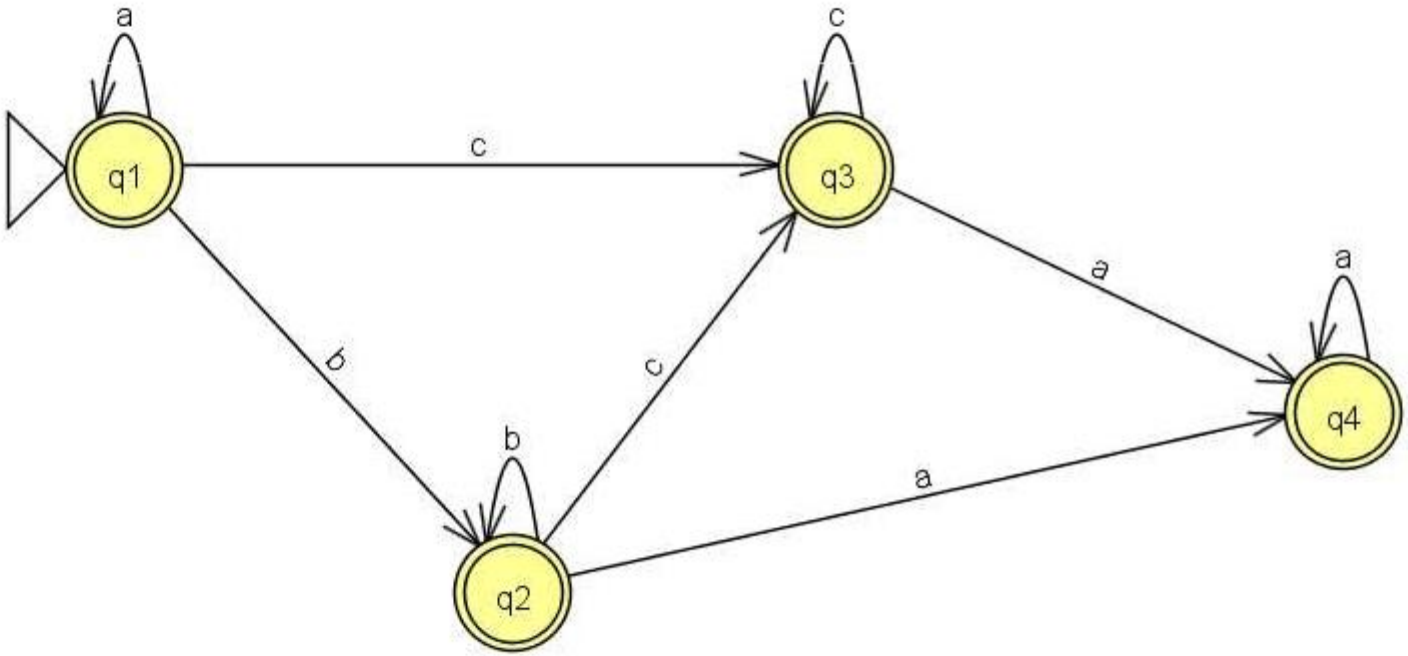
- **Simule para a entrada ab**

Uso de transições em vazio

- Assim como ocorre no caso dos AFD's e AFND's, alguns AFNDε se mostram mais simples de serem analisados do que as correspondentes versões isentas de transições em vazio.

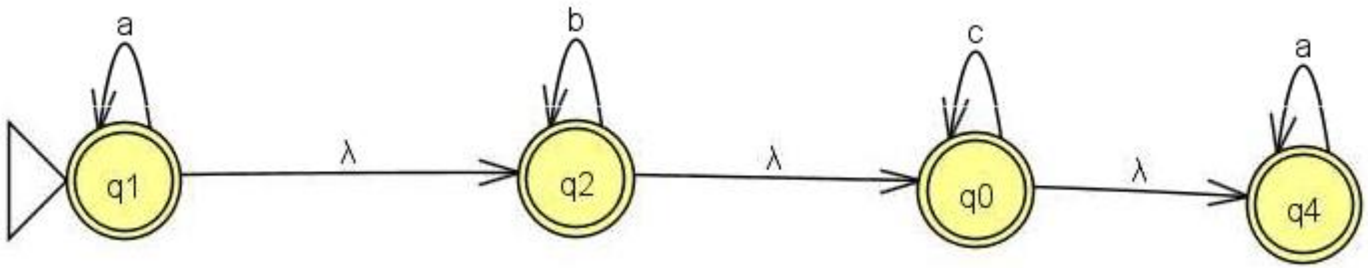
AF que reconhece a linguagem a*b*c*a*.

- sem movimento vazio



AF que reconhece a linguagem a*b*c*a*.

- com movimento vazio



AF com e sem movimento vazio – Linguagem

- Todo autômato com transições em vazio gera uma linguagem que é aceita por algum autômato finito que não contém transições em vazio

Eliminação de transições em vazio

Eliminação de transições em vazio

- Algoritmo: “Obtenção de um autômato finito N, sem transições em vazio, a partir de um autômato finito M, com transições em vazio.”

  - Entrada: um autômato finito com transições em vazio M;
  - Saída: um autômato finito sem transições em vazio N, tal que L(N) = L(M);
  - Método:

1. Eliminação das transições em vazio

• Considere-se um estado qualquer q_i Q. Se houver uma transição em vazio de q_i para q_j, deve-se eliminá-la, copiando-se para a linha que representa o estado q_i todas as transições que partem dos estados q_j para os quais é feita a transição em vazio.

• Esse procedimento corresponde, em notação tabular, à realização de uma fusão ("merge") entre a linha do estado q_i que contém a transição em vazio para o estado-destino q_j e a própria linha do estado q_j, armazenando-se o resultado novamente na linha correspondente ao estado q_i.

• Havendo mais de uma transição em vazio indicadas, deve-se repetir cumulativamente o procedimento para todas elas.

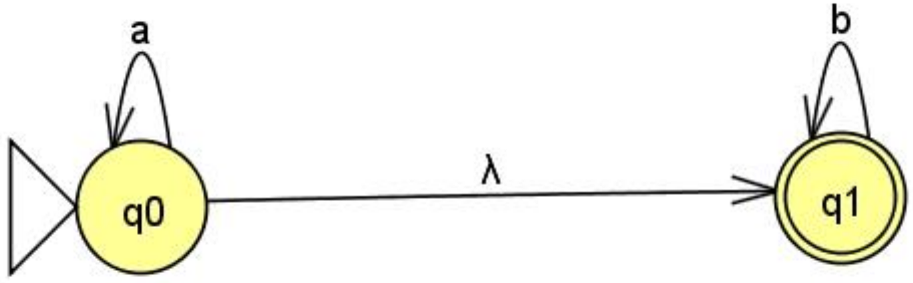
• Se $\delta(q_i, ⎿)$ F, então $F' \leftarrow F' * \{q_i\}$, sendo que inicialmente $F' \leftarrow$ F.

2. Iteração

- Repetir o passo anterior para os demais estados do autômato, até que todos eles tenham sido considerados (ou seja, até que a última linha tenha sido atingida).

- Nos casos em que houver transições em vazio para estados que por sua vez também transitam em vazio para outros estados, será necessário iterar o procedimento várias vezes sobre a tabela, até que todas as transições em vazio tenham sido eliminadas.

Exemplo: eliminação da transição em vazio



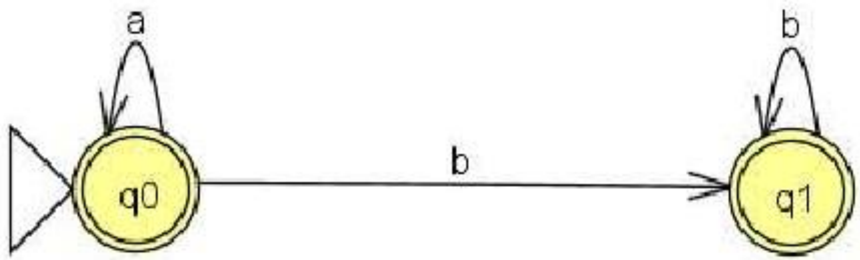
notação tabular

δ	a	b	⌊
q0	{q0}		{q1}
*q1		{q1}	

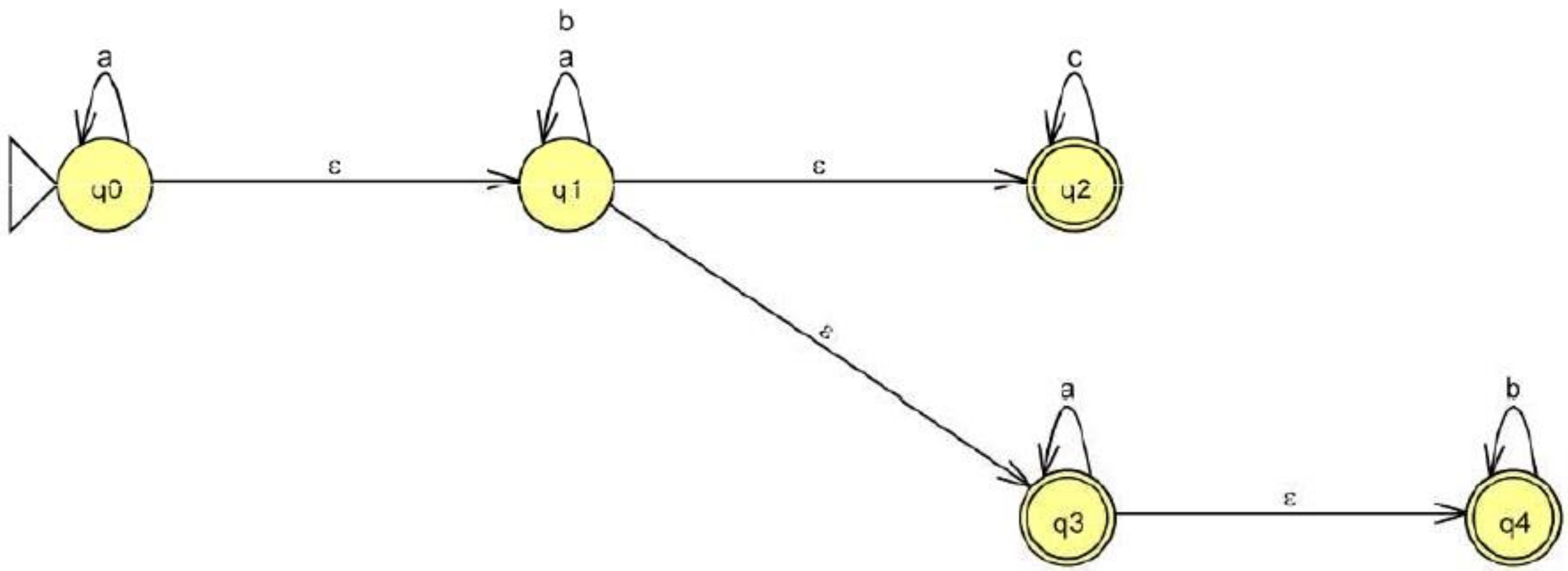
Como há uma transição em vazio de q0 para q1, deve-se copiar as transições de q1 para q0 (δ(q1,b) apenas, neste caso) e, além disso, considerar q0 como estado final, uma vez que q1 é estado final. Abaixo, a tabela e o AF sem movimento vazio.

notação tabular

δ	a	b
*q0	{q0}	{q1}
*q1		{q1}



Exercício: Obter um AF sem movimento vazio



Exercícios: Construa AFND's com ou sem movimentos vazios para as linguagens abaixo. Para os que apresentarem movimentos vazios, aplique o algoritmo de eliminação de movimentos vazios:

L1 = Aceita cadeias $\in \{1,2\}^*$ tal que o último símbolo na cadeia tenha aparecido anteriormente. Em seguida, construa a Tabela de Transição de Estados e a Função de Transição de Estados

L2 = Aceita cadeias $\in \{1,2,3\}^*$ tal que o último símbolo na cadeia tenha aparecido anteriormente. Por exemplo, 121 é aceita; 31312 não é aceita. Em seguida, construa a Tabela de Transição de Estados e a Função de Transição de Estados

L3 = $\{ w|w \in \{a,b,c\}^*$, aa ou bb é subpalavra e cccc é sufixo de w$\}$

L4 = $\{ w \mid w \in \{a,b\}^*$ e o quarto símbolo da direita para a esquerda de w é a $\}$

L5 = $\{ w1w2w1 \mid w1, w2 \in \{0,1\}^*$ e $|w1|=2 \}$