



# Computabilidade

## Aula 2 – Revisão de Autômatos



# Aulas

- Sextas:
  - Horário: 09h40 até 12h20

# Material das Aulas e Atividades

- Todo o material das aulas, incluindo as atividades propostas, será postado no Google Classroom da disciplina.
- Link:  
<https://classroom.google.com/c/NjAwOTIzMjM1NjA3?cjc=cltfict>
- Ou leia o QR-Code ao lado.







# Autômato

- Máquina abstrata que recebe entradas e produz saídas de acordo com um conjunto de regras predefinidas
- Existem diferentes tipos de autômatos:
  - Autômatos finitos determinísticos
  - Autômatos finitos não-determinísticos
  - Autômatos com pilha
  - Máquinas de Turing

# Autômatos finitos determinísticos

- Pode ser visto como uma máquina formada por três componentes:
  - Fita: Dispositivo de entrada que contém a informação que o autômato deve processar.
  - Unidade de Controle: Armazena o estado corrente da máquina e controla a cabeça de leitura da fita. A unidade de controle se movimenta sobre a fita da esquerda para a direita.
  - Programa: Comanda as leituras e determina a transição de estados da máquina

# Autômatos finitos determinísticos

- Definição formal: Um autômato finito determinístico é formado por um 5-upla:  $M = (\Sigma, Q, \delta, q_0, F)$
- Onde:
  - $\Sigma$  é um alfabeto de símbolos de entrada (símbolos que a máquina reconhece)
  - $Q$  é um conjunto finito de estados possíveis para o autômato
  - $\delta$  é o programa do autômato:  $\delta: Q \times \Sigma \rightarrow Q$ 
    - Isso é: o programa determina um novo estado  $q$  para o autômato baseado no estado atual  $p$  e no símbolo lido  $a$ . Essa combinação pode ser vista como  $\delta(p, a) = q$
  - $q_0$  é o estado inicial do autômato (ao ser iniciado, ele começa no estado  $q_0$ )
  - $F$  é um subconjunto de  $Q$  chamado de estados finais. Se  $M$  terminar em um estado  $q \in F$  então a máquina “aceitou” ou “reconheceu” a entrada.

# Autômatos finitos determinísticos

- Para a máquina ser chamada de determinística:
  - O programa do autômato:  $\delta: Q \times \Sigma \rightarrow Q$  deve indicar claramente todas as transições possíveis
  - Para cada estado possível  $q_i \in Q$  e para cada símbolo (entrada) possível  $a \in \Sigma$  a máquina deve determinar um novo estado  $q_j \in Q$
  - Conhecendo a função de transição  $\delta$  é possível determinar o resultado final.
  - Para uma mesma máquina e entrada, o resultado será sempre o mesmo.

# Autômatos finitos determinísticos

- Exemplo de AFD (autômato finito determinístico):
- $M = (\{q_1, q_2, q_3\}, \{0,1\}, \delta, q_1, \{q_3\})$
- $\delta = \{ (q_1, 0) \rightarrow q_2, (q_1, 1) \rightarrow q_1, (q_2, 0) \rightarrow q_2, (q_2, 1) \rightarrow q_3, (q_3, 0) \rightarrow q_3, (q_3, 1) \rightarrow q_3 \}$
- A função de transição também pode ser vista como um tabela:

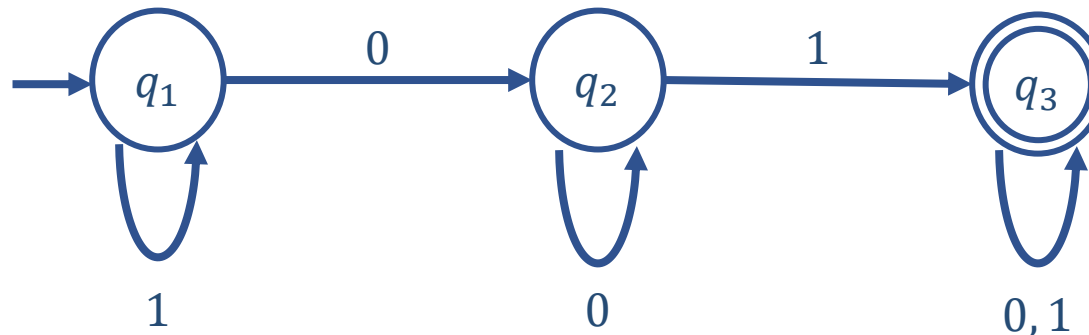
| $\delta$ | 0     | 1     |
|----------|-------|-------|
| $q_1$    | $q_2$ | $q_1$ |
| $q_2$    | $q_2$ | $q_3$ |
| $q_3$    | $q_3$ | $q_3$ |

No estado  $q_2$ , se a entrada for 0, permaneça no estado  $q_2$



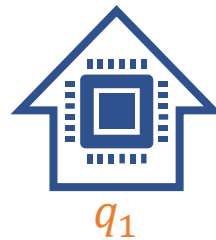
# Autômatos finitos determinísticos

- Os autômatos podem ser representados também graficamente:
  - Cada estado equivale a um círculo
  - Os estados finais são desenhados com borda dupla
  - Cada transição possível equivale a uma seta no diagrama
  - O estado inicial é indicado por uma seta sem origem definida
- O autômato do slide anterior pode ser desenhado como:

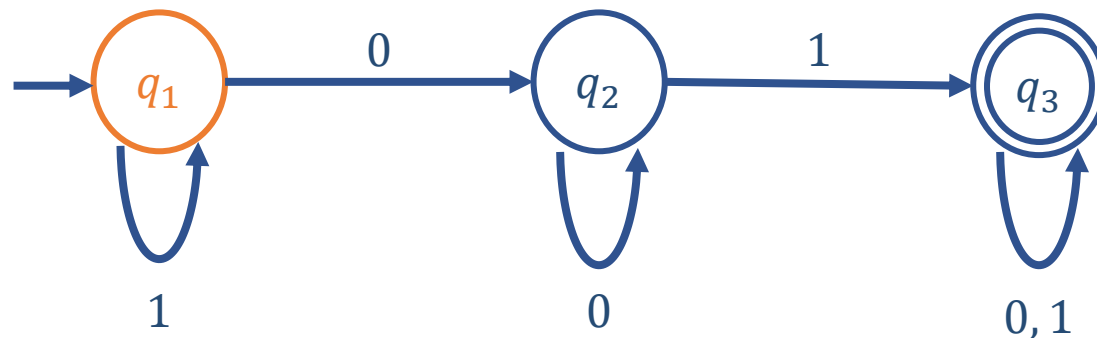


# Autômatos finitos determinísticos

- Simulação de um AFD:

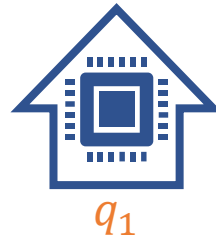


Unidade de controle:  
Responsável por ler a fita  
e determinar os estados

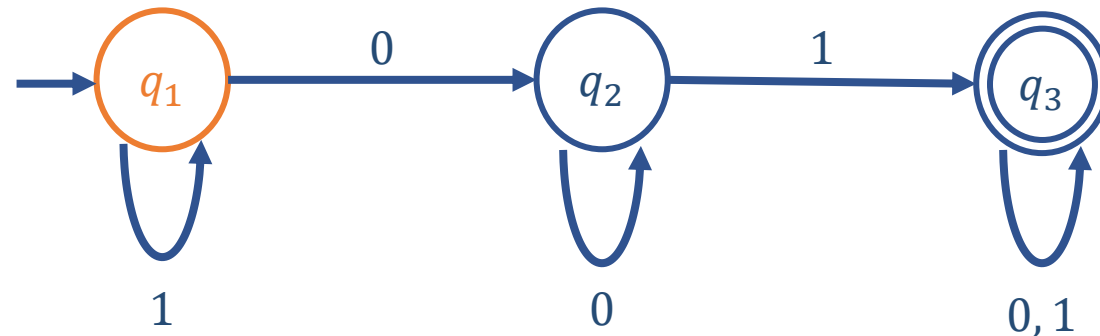


# Autômatos finitos determinísticos

- Simulação de um AFD:
- Estado inicial:  $q_1$

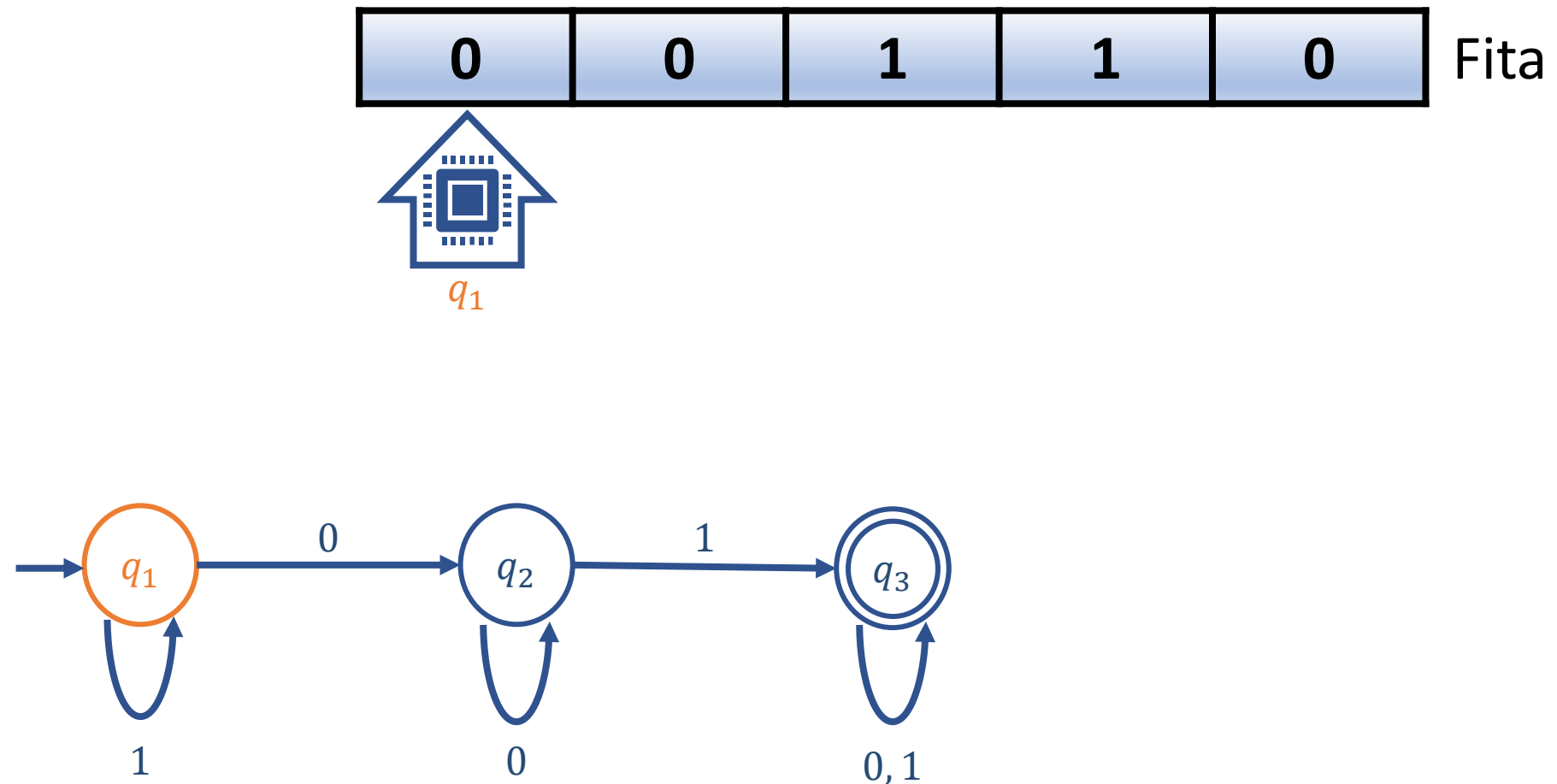


Unidade de controle:  
Responsável por ler a fita  
e determinar os estados



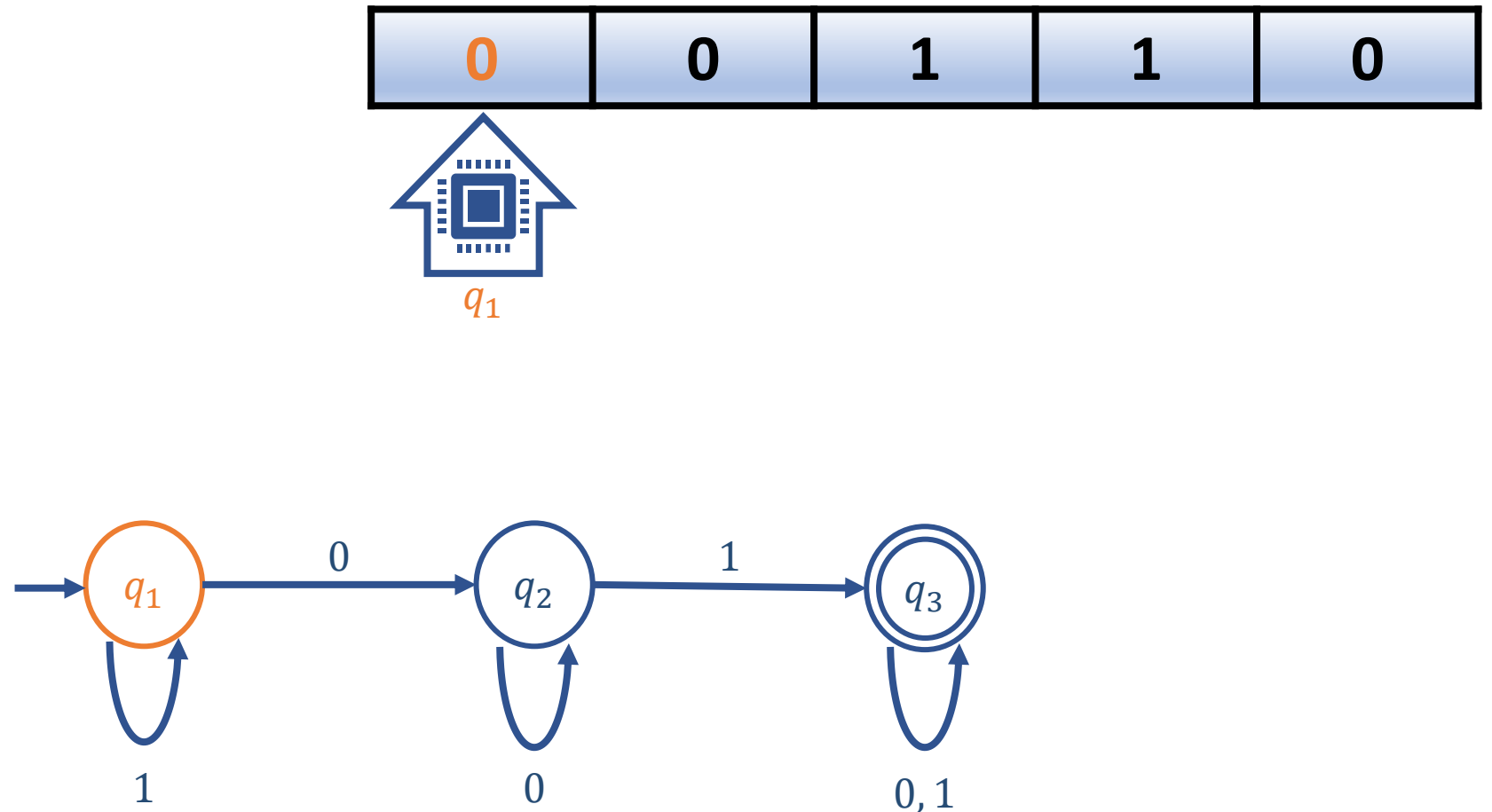
# Autômatos finitos determinísticos

- Simulação de um AFD:



# Autômatos finitos determinísticos

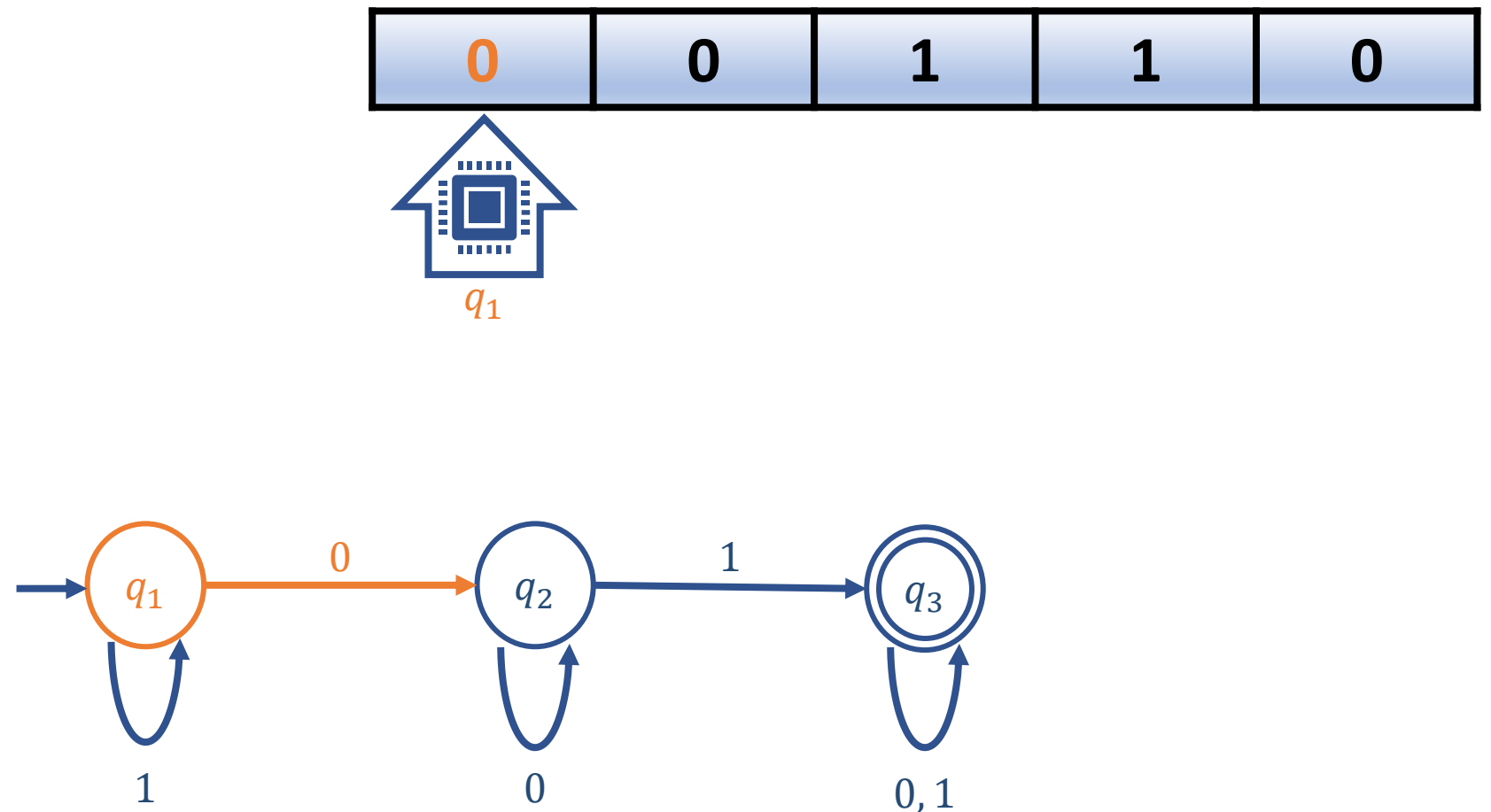
- Simulação de um AFD:





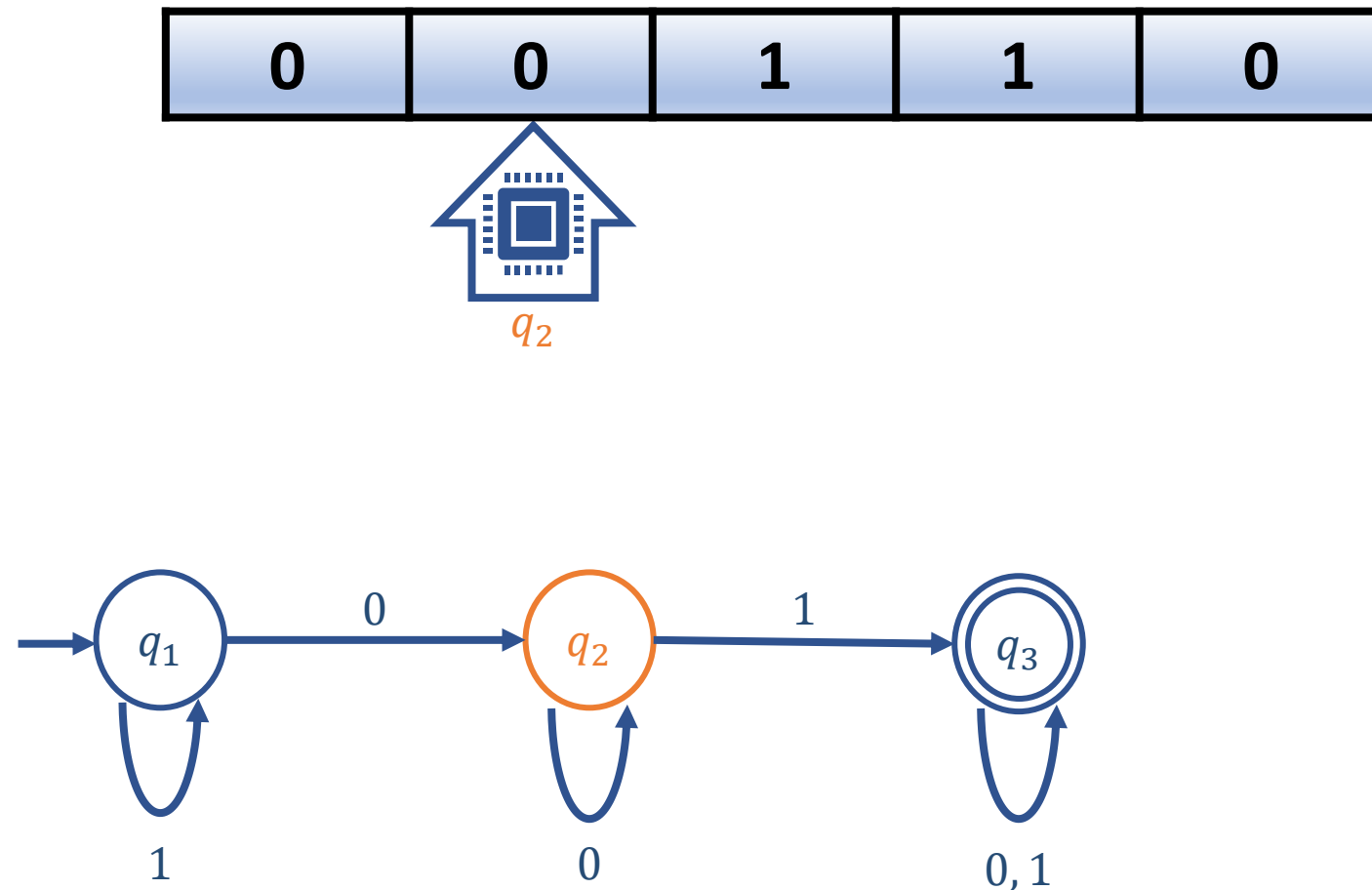
# Autômatos finitos determinísticos

- Simulação de um AFD:



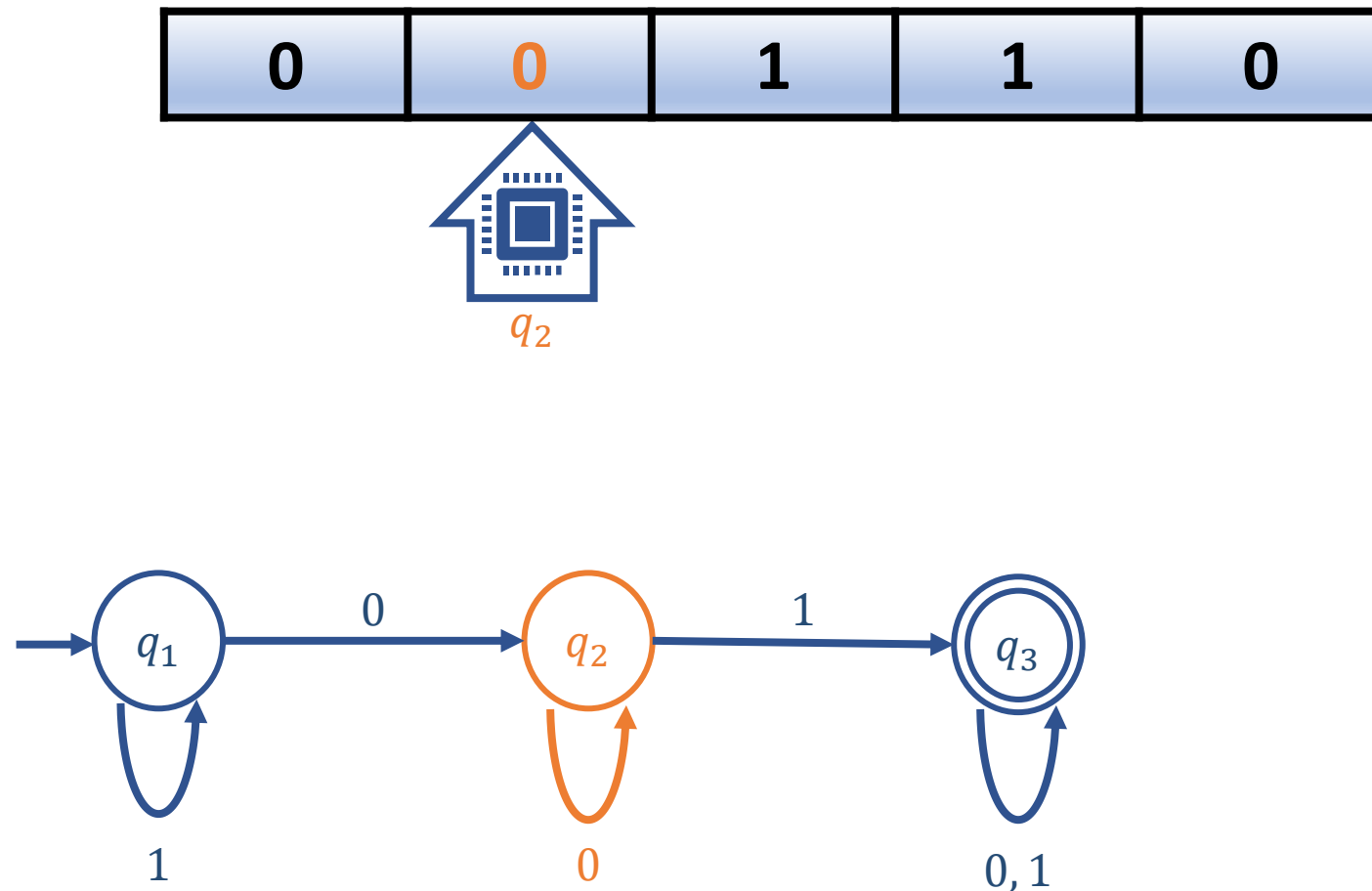
# Autômatos finitos determinísticos

- Simulação de um AFD:



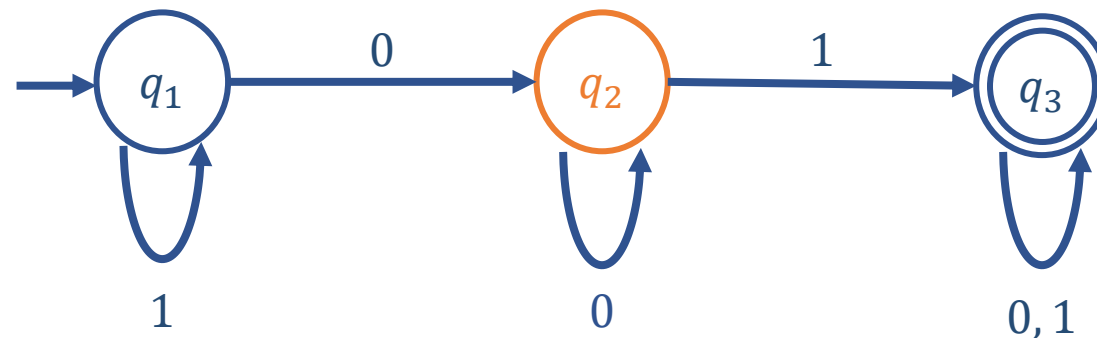
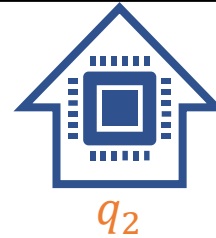
# Autômatos finitos determinísticos

- Simulação de um AFD:



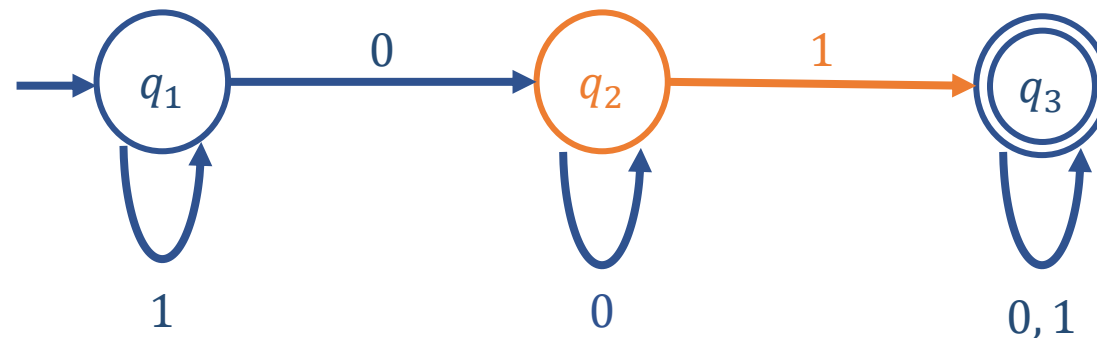
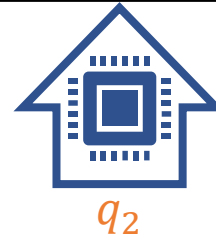
# Autômatos finitos determinísticos

- Simulação de um AFD:



# Autômatos finitos determinísticos

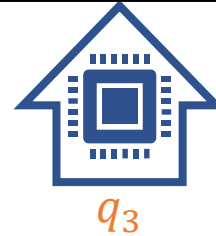
- Simulação de um AFD:



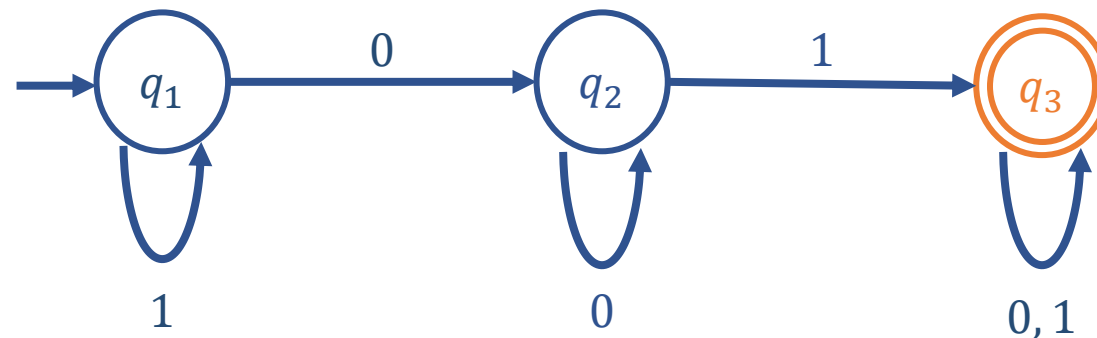


# Autômatos finitos determinísticos

- Simulação de um AFD:

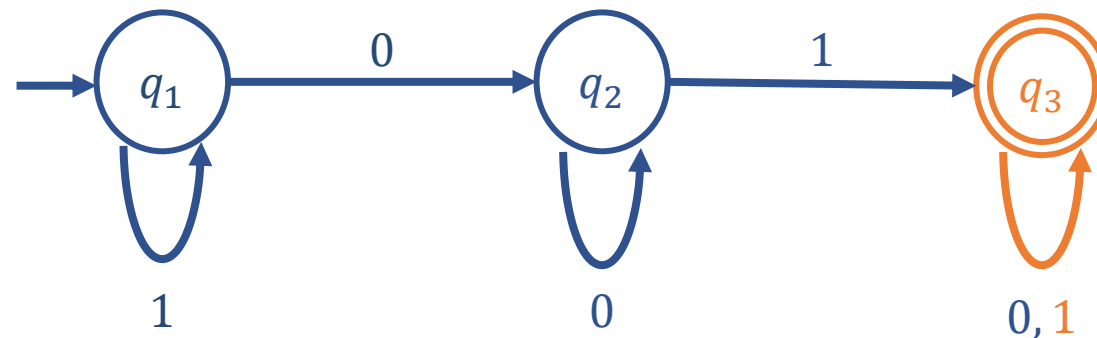
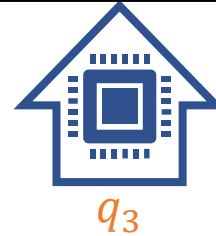


$q_3$



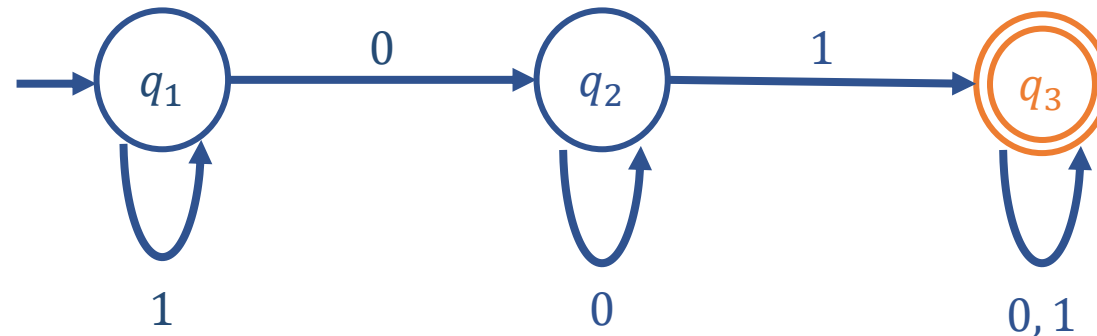
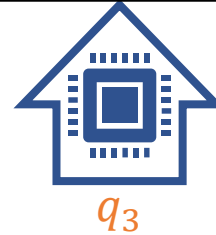
# Autômatos finitos determinísticos

- Simulação de um AFD:



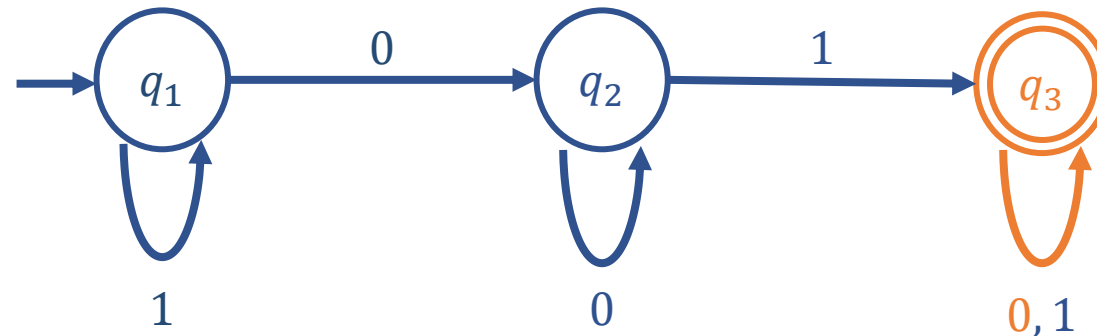
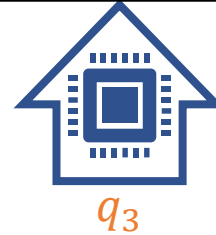
# Autômatos finitos determinísticos

- Simulação de um AFD:



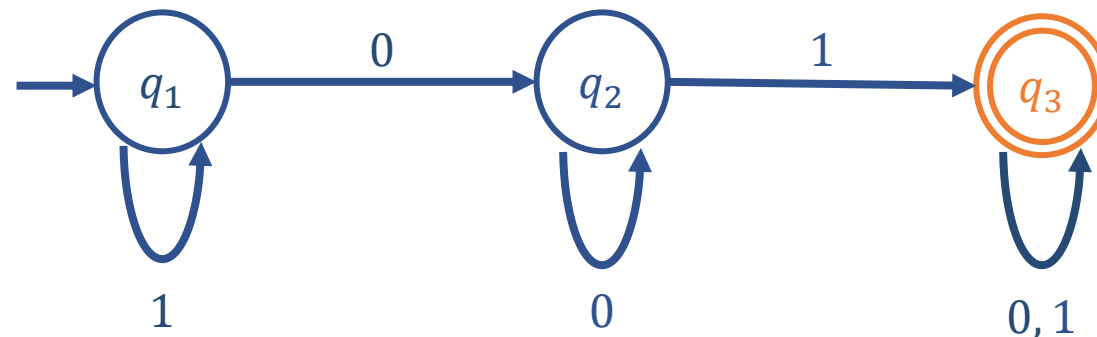
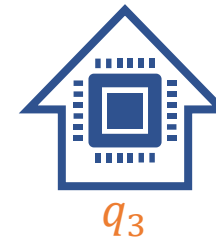
# Autômatos finitos determinísticos

- Simulação de um AFD:



# Autômatos finitos determinísticos

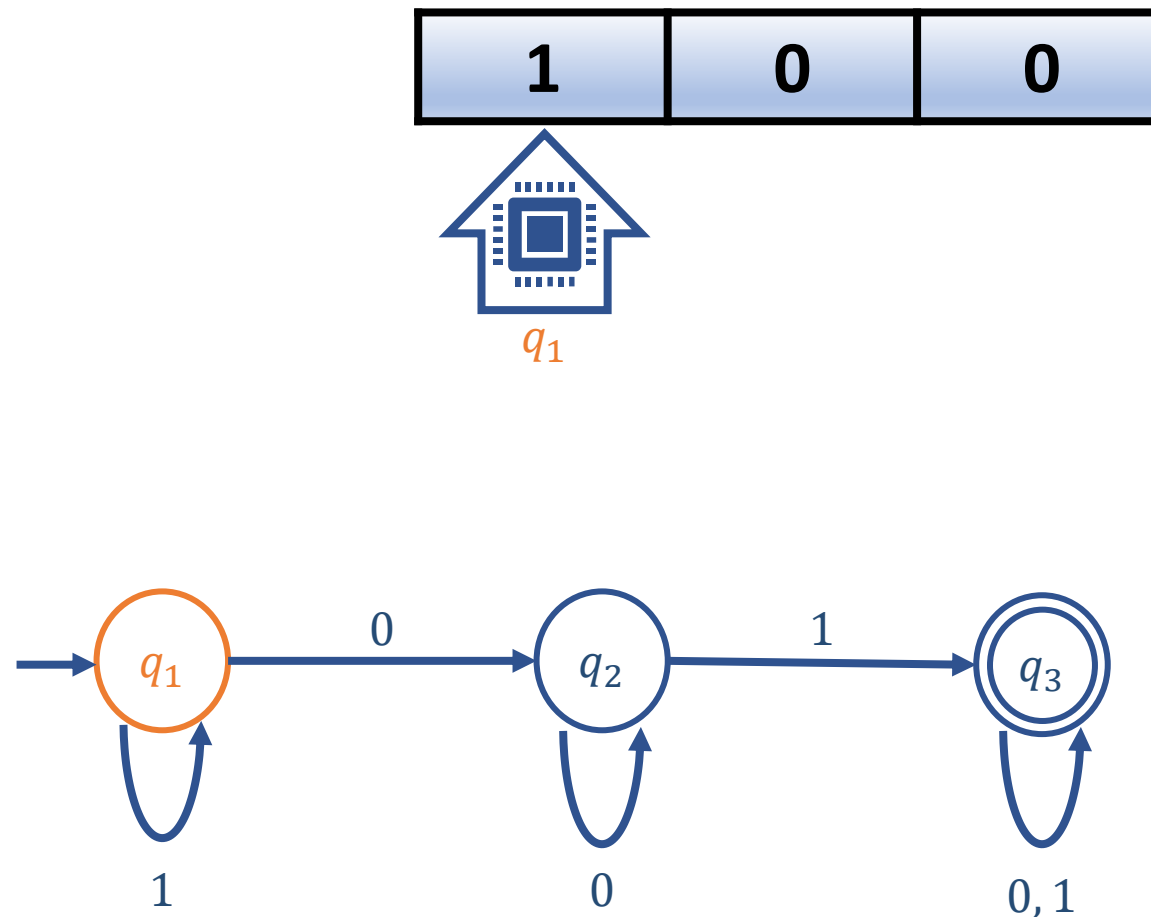
- Simulação de um AFD:
  - Autômato terminou a leitura da fita
  - Terminou em um estado final? **Sim**
  - **O autômato aceitou a entrada**
  - **Reconheceu a entrada como válida**





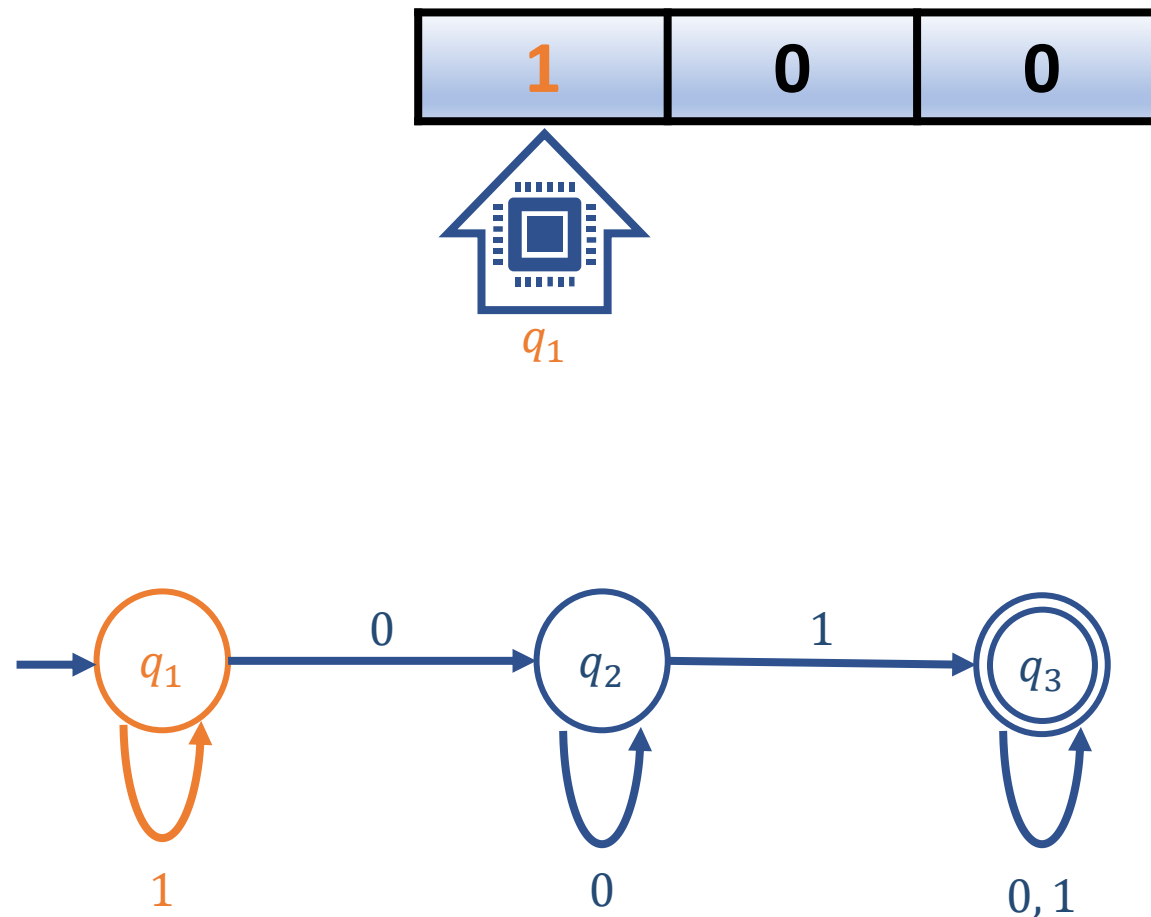
# Autômatos finitos determinísticos

- Simulação de um AFD:



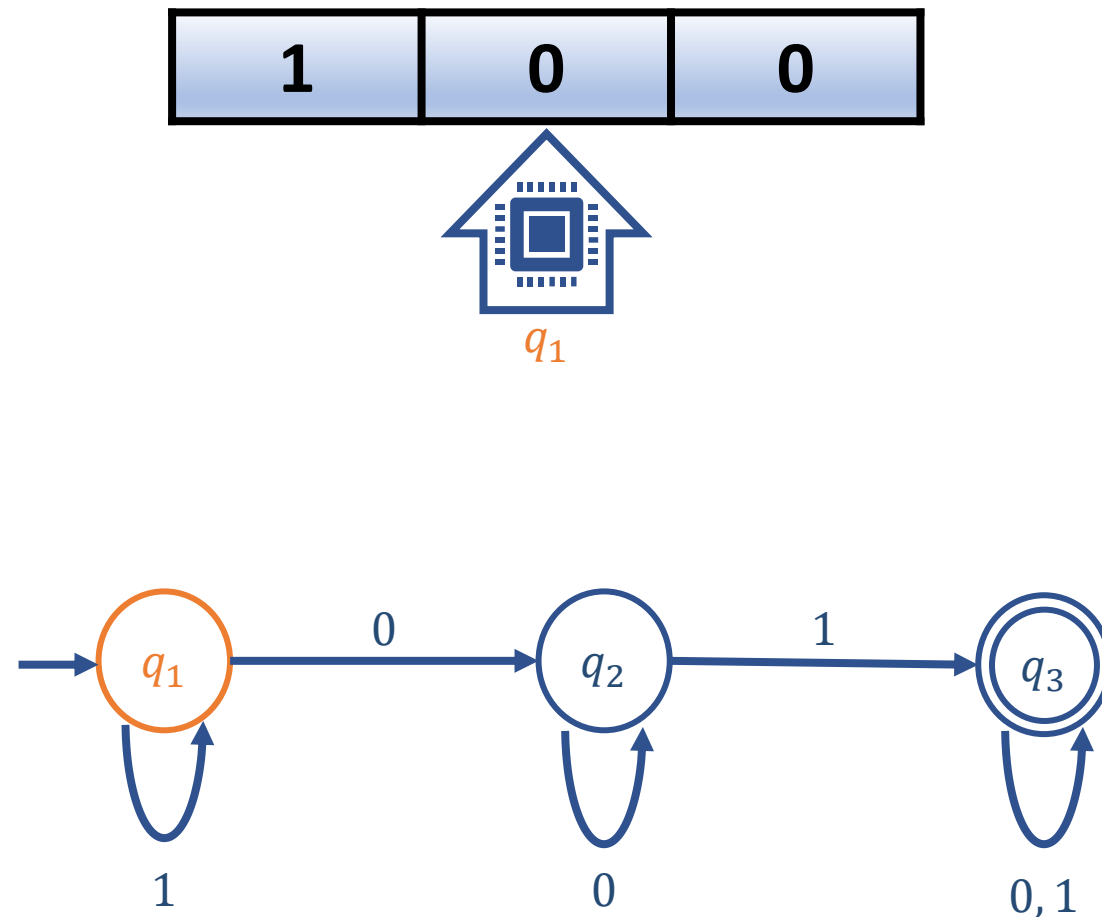
# Autômatos finitos determinísticos

- Simulação de um AFD:



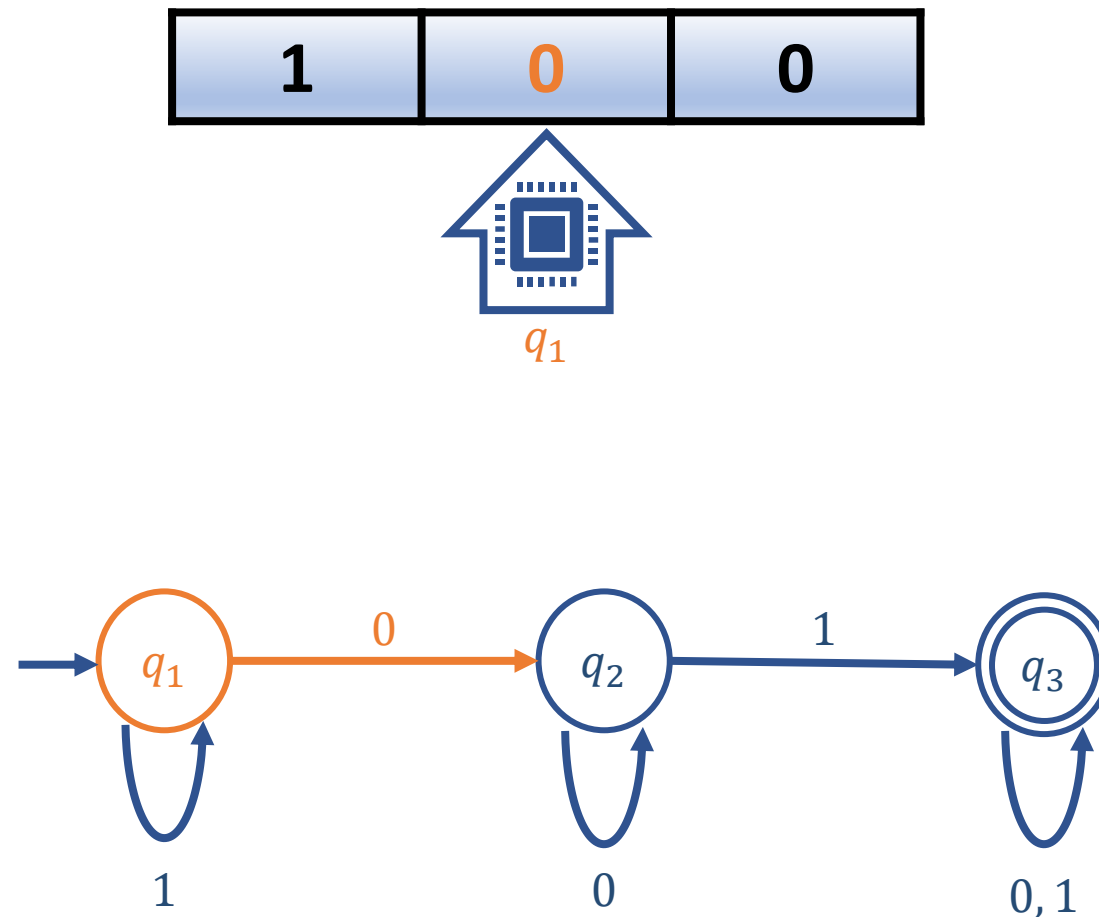
# Autômatos finitos determinísticos

- Simulação de um AFD:



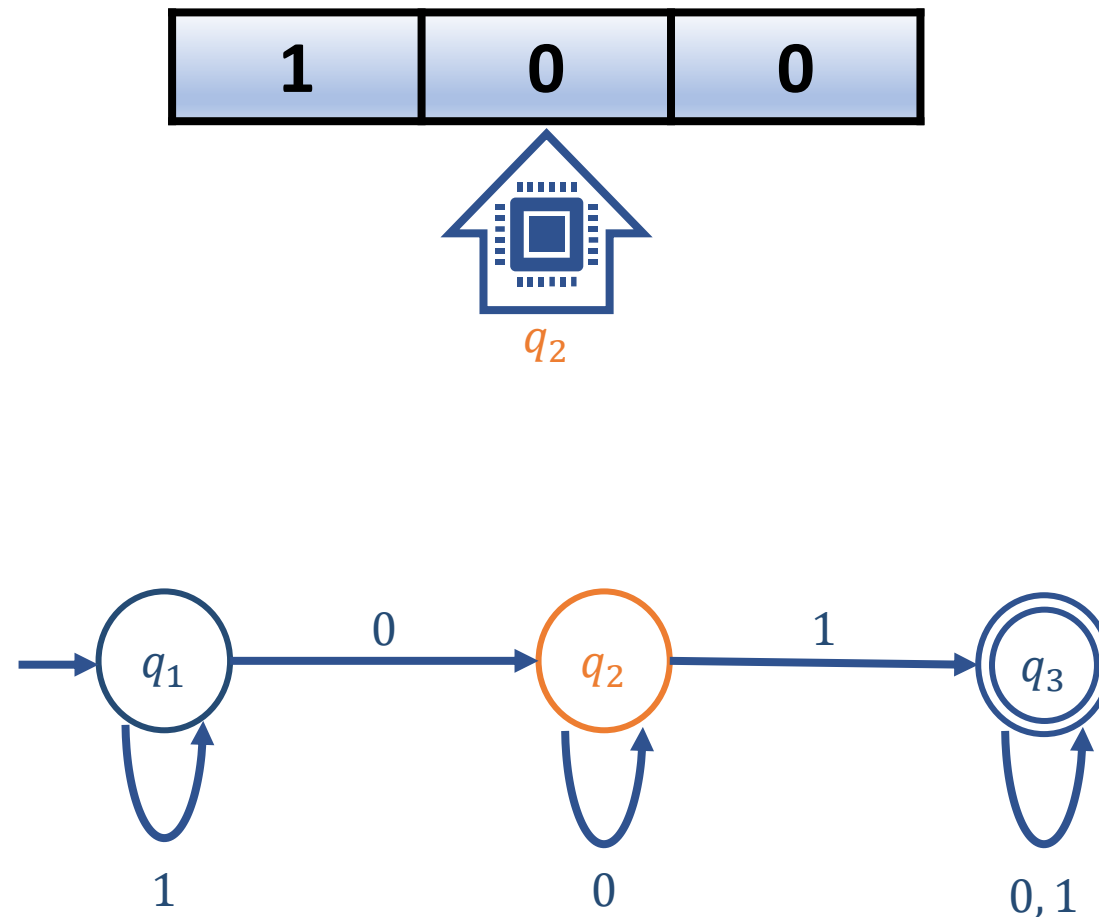
# Autômatos finitos determinísticos

- Simulação de um AFD:



# Autômatos finitos determinísticos

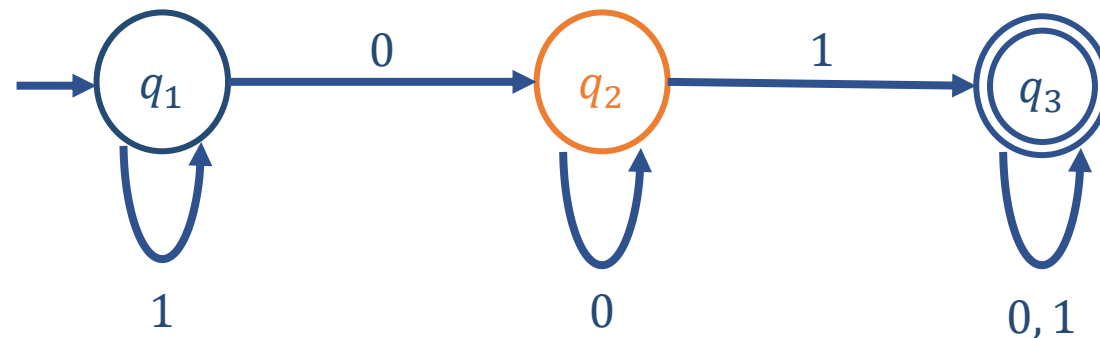
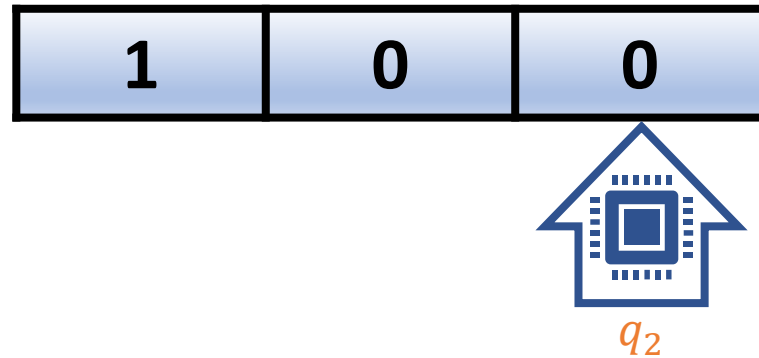
- Simulação de um AFD:





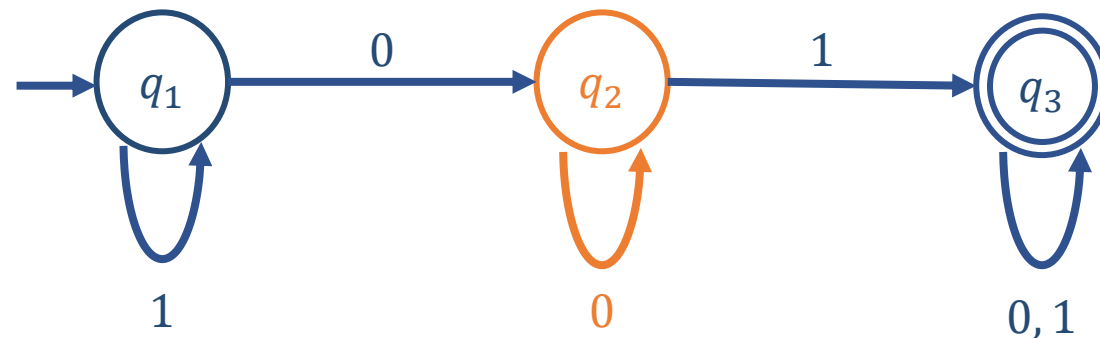
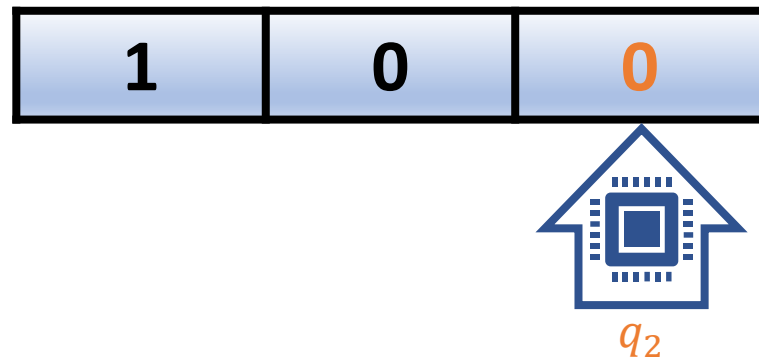
# Autômatos finitos determinísticos

- Simulação de um AFD:



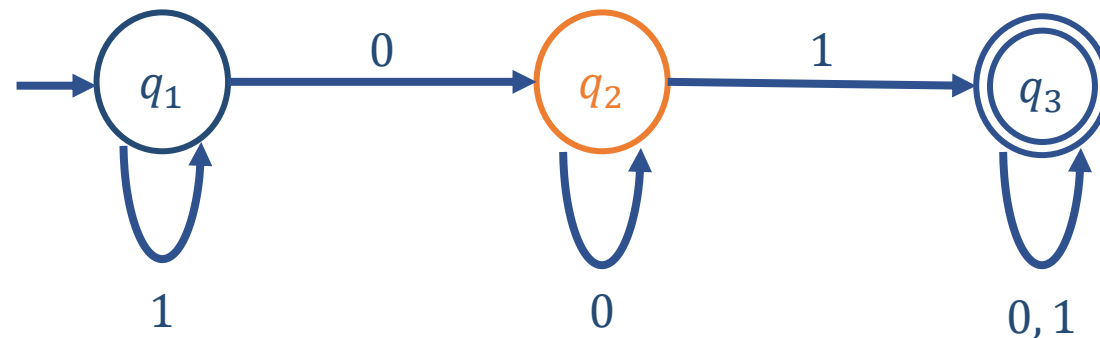
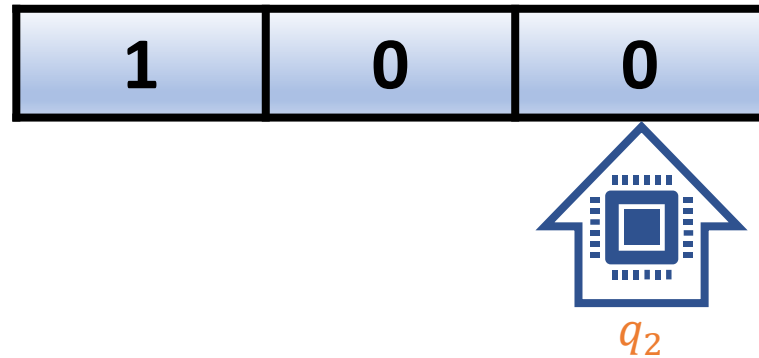
# Autômatos finitos determinísticos

- Simulação de um AFD:



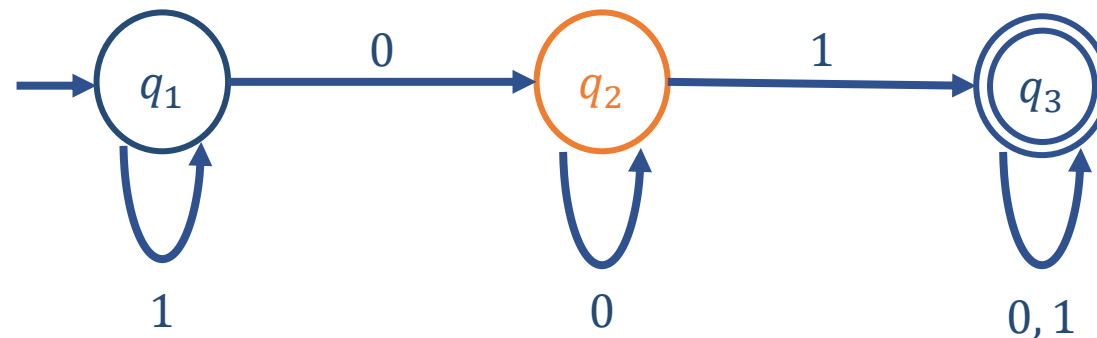
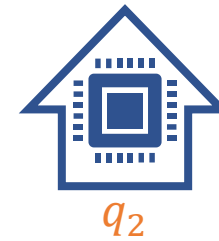
# Autômatos finitos determinísticos

- Simulação de um AFD:



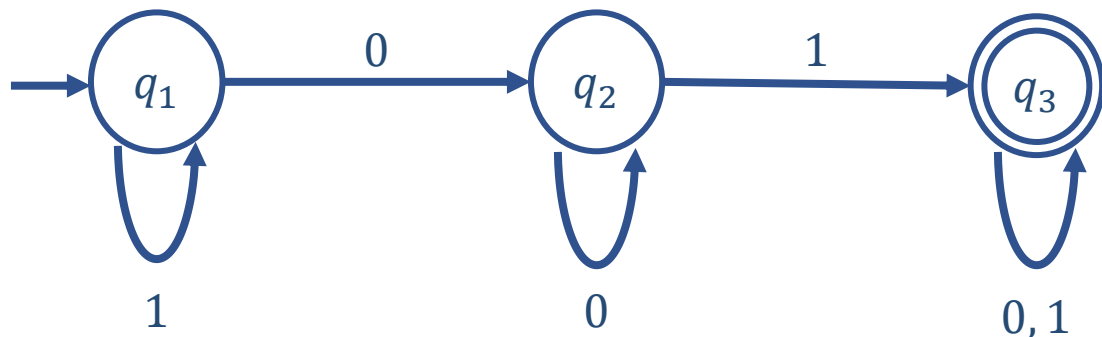
# Autômatos finitos determinísticos

- Simulação de um AFD:
  - Autômato terminou a leitura da fita
  - Terminou em um estado final? **Não**
  - **O autômato não aceitou a entrada**
  - **Reconheceu a entrada como inválida**



# Autômatos finitos determinísticos

- Os AFDs são muito utilizados para reconhecer se uma determinada entrada satisfaz ou não uma linguagem.
- Por exemplo:
  - Qual linguagem esse autômato reconhece?
  - Isso é: quais características uma entrada precisa ter para ser aceita pelo autômato?
  - A entrada precisa conter pelo menos um 0 e, em seguida, pelo menos um 1.



01 ✓

0001 ✓

0111 ✓

00100 ✓

10 ✗

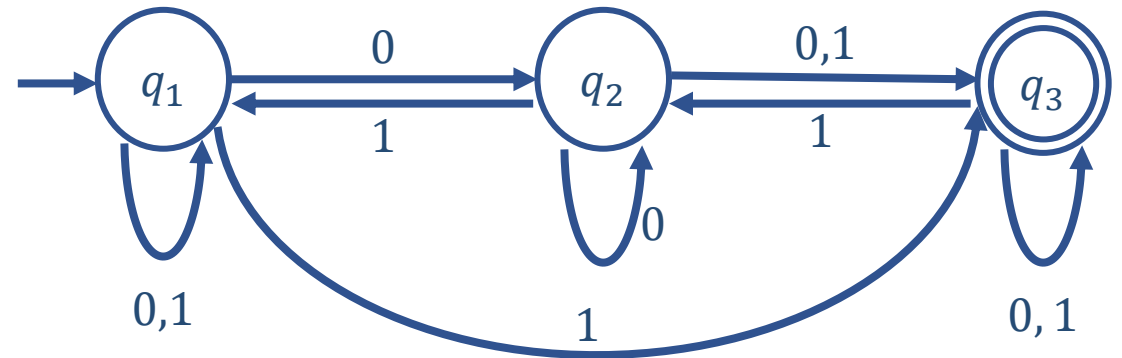
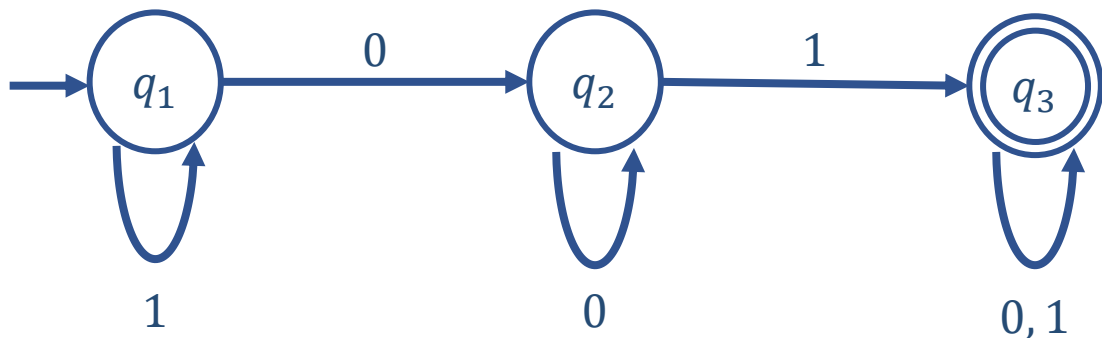
1000 ✗

11111 ✗

00000 ✗

# Autômato Finito Não Determinístico

- Diferentemente dos AFDs, os autômatos finitos não determinísticos permitem múltiplos caminhos possíveis saindo de um estado para um mesmo símbolo (entrada).





# Autômato Finito Não Determinístico

- Modelo teórico em que a transição depende de fatores aleatórios
- O resultado do processamento não pode ser determinado com base na entrada e na função de transição.
- Ele depende de fatores aleatórios.
- Usado também para reconhecer algumas linguagens e gramáticas.

# Autômato com Pilha

- Uma evolução da ideia dos autômatos finitos.
- Possui uma pilha como componente adicional.
- A pilha é usada como uma memória para o autômato.
- Pilha:
  - Estrutura de dados que permite armazenar e recuperar informações de forma LIFO (last-in, first-out)
  - LIFO = O último elemento inserido na pilha é o primeiro a ser removido
  - As inserções e remoções são executadas exclusivamente no topo da pilha



# Autômato com Pilha

- A pilha armazena informações sobre o histórico das transições realizadas pela máquina.
- O autômato pode ler um símbolo de entrada, fazer uma transição de estado e, opcionalmente, modificar a pilha adicionando ou removendo elementos.
- A transição de estado é determinada por três fatores:
  - o estado atual
  - o símbolo lido da fita de entrada
  - o topo da pilha

# Autômatos com Pilha

- Assim como em um autômato finito, um AP pode ser determinístico ou não-determinístico.
- Em autômatos determinísticos com pilha:
- Para cada combinação de estado, símbolo de entrada e topo da pilha existe apenas uma transição possível.

# Autômatos com Pilha

- Um autômato com pilha é formado por um 6-upla:
- $M = (\Sigma, \Gamma, Q, \delta, q_0, F)$
- Onde:
  - $\Sigma$  é o alfabeto de símbolos de entrada (símbolos que a máquina reconhece)
  - $\Gamma$  é o alfabeto de símbolos que podem ser escritos na pilha
  - $Q$  é um conjunto finito de estados possíveis para o autômato
  - $\delta$  é o programa do autômato:  $\delta: Q \times \Sigma \times \Gamma \rightarrow Q \times \Gamma$
  - $q_0$  é o estado inicial do autômato (ao ser iniciado, ele começa no estado  $q_0$ )
  - $F$  é um subconjunto de  $Q$  chamado de estados finais. Se  $M$  terminar em um estado  $q \in F$  então a máquina “aceitou” ou “reconheceu” a entrada.
- Vale observar que  $\varepsilon \in \Sigma$  e  $\varepsilon \in \Gamma$ . Onde  $\varepsilon$  significa uma entrada vazia (seja para a fila de entrada ou para a pilha)

# Autômatos com Pilha

- Observe que cada transição é formada por:
  - $(q_i, a, b) \rightarrow (q_j, c)$
- Onde:
  - $q_i \in Q$  é o estado atual da máquina
  - $a \in \Sigma$  é a informação lida da fita de entrada
  - $b \in \Gamma$  é a informação que consta no topo da pilha
  - $q_j \in Q$  é o estado para o qual a máquina deve transacionar
  - $c \in \Gamma$  é a informação que deve ser escrita no topo da pilha
- Se estou no estado  $q_i$  E li  $a$  na fila E  $b$  está no topo da pilha então vá para o estado  $q_j$  E escreva  $c$  na pilha

# Atividade:

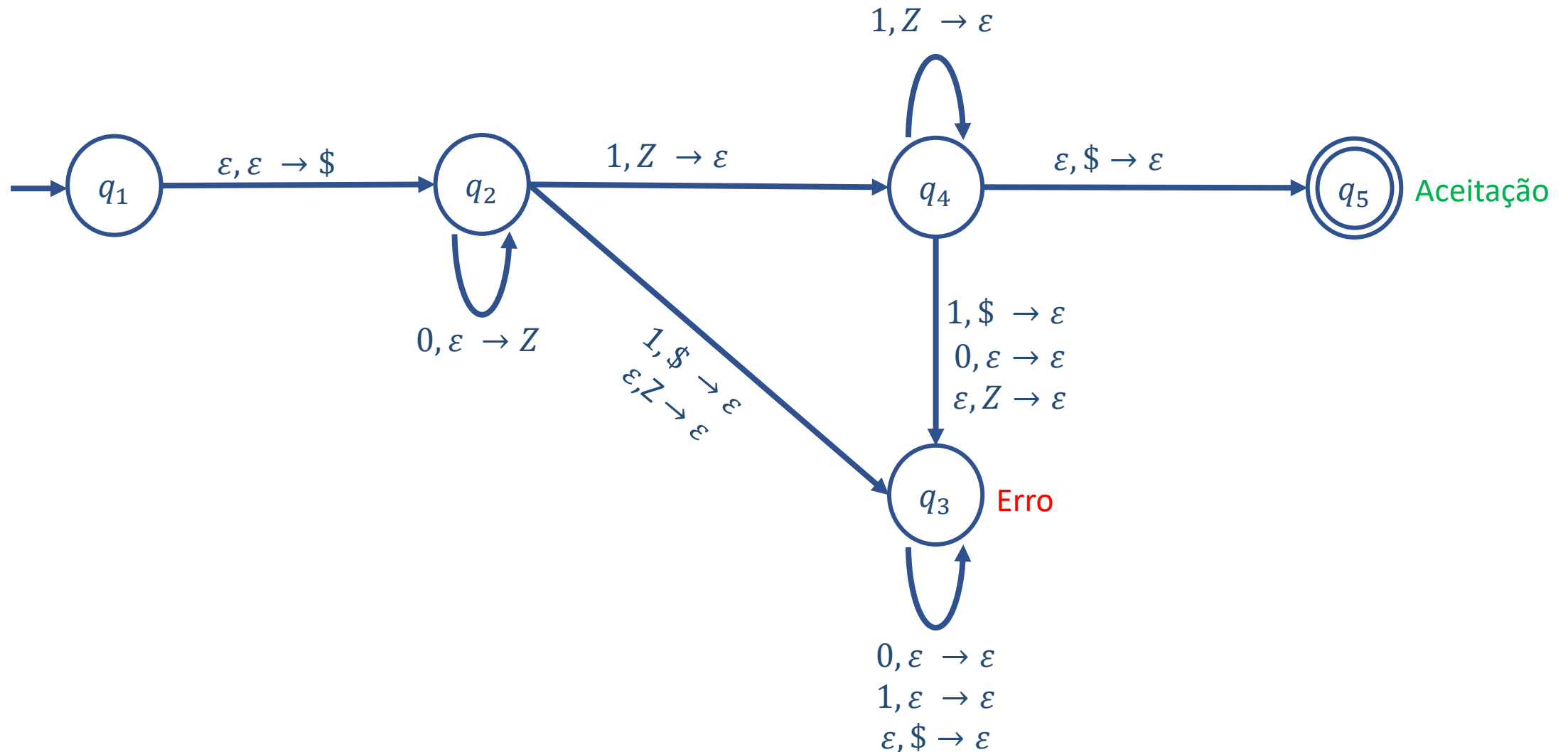
- Pense em um AFD com Pilha para reconhecer uma linguagem formada por  $0^n 1^n$ .
- Por exemplo:
  - 000111 e 01 são aceitos
  - Enquanto 001100 e 00110011 não são aceitos
- Dica: Como verificar se a pilha está vazia?
  - Escrever um símbolo logo no início do autômato.

# Resposta

- $M = (\{0,1,\varepsilon\}, \{\$,Z,\varepsilon\}, \{q_1,q_2,q_3,q_4,q_5\}, \delta, q_1, \{q_5\})$
- Função de Transição  $\delta$ :

| Estado Atual | Símbolo lido da fila | Valor lido da pilha | → | Novo Estado | Valor escrito na pilha |
|--------------|----------------------|---------------------|---|-------------|------------------------|
| $q_1$        | $\varepsilon$        | $\varepsilon$       |   | $q_2$       | $\$$                   |
| $q_2$        | 0                    | $\varepsilon$       |   | $q_2$       | Z                      |
| $q_2$        | 1                    | $\$$                |   | $q_3$       | $\varepsilon$          |
| $q_2$        | $\varepsilon$        | Z                   |   | $q_3$       | $\varepsilon$          |
| $q_2$        | 1                    | Z                   |   | $q_4$       | $\varepsilon$          |
| $q_3$        | 0,1                  | $\varepsilon$       |   | $q_3$       | $\varepsilon$          |
| $q_3$        | $\varepsilon$        | $\$$                |   | $q_3$       | $\varepsilon$          |
| $q_4$        | 1                    | Z                   |   | $q_4$       | $\varepsilon$          |
| $q_4$        | 1                    | $\$$                |   | $q_3$       | $\varepsilon$          |
| $q_4$        | 0                    | $\varepsilon$       |   | $q_3$       | $\varepsilon$          |
| $q_4$        | $\varepsilon$        | Z                   |   | $q_3$       | $\varepsilon$          |
| $q_4$        | $\varepsilon$        | $\$$                |   | $q_5$       | $\varepsilon$          |

# Resposta





# Próxima Aula

- Máquina de Turing
- Tese de Church-Turing