

# Aspectos Teóricos da Computação

Prof. Rodrigo Martins

[rodrigo.martins@francomontoro.com.br](mailto:rodrigo.martins@francomontoro.com.br)

# Cronograma da Aula

- ◆ Autômatos finitos ou Máquina de estados finito
- ◆ Grafo de estados ou Diagrama de estados
- ◆ Tabela de transição
- ◆ Tabela de estados

# Autômatos Finitos

- ◆ Este material introduz a classe de linguagens conhecidas como “linguagens regulares”.
- ◆ Essas linguagens são exatamente aquelas que podem ser descritas por autômatos finitos.

# O que são Autômatos Finitos?

- ◆ É um modelo matemático de computação usado para projetar tanto softwares quanto hardwares.
- ◆ Consiste em um número finito de estados e é usado para reconhecer padrões dentro de cadeias de texto.

# O que são Autômatos Finitos?

- ◆ Um autômato finito tem um conjunto de estados, e seu controle se desloca de estado para estado em resposta a “entradas” externas.
- ◆ Autômatos finitos podem ser:
  - **Determinístico**, que significa que o autômato não pode estar em mais de um estado em qualquer instante.
  - **Não determinísticos**, significa que o autômato pode estar em vários estados ao mesmo tempo.

# Componentes principais de um AF

## ◆ **Conjunto de Estados:**

- Um AF é composto por um conjunto finito de estados. Cada estado representa uma condição ou situação dentro do autômato.
- Os estados são cruciais para determinar o comportamento do autômato. Eles definem as diferentes situações pelas quais o autômato pode passar durante seu processamento.

# Componentes principais de um AF

## ◆ Alfabeto

- O alfabeto é um conjunto finito de símbolos que o autômato pode processar. Cada símbolo do alfabeto pode ser considerado como uma entrada para o autômato.
- O alfabeto define as entradas possíveis que o autômato pode receber e processar. Cada transição de estado no autômato é desencadeada por um símbolo do alfabeto.

# Componentes principais de um AF

- ◆ **Função de transição**

- A função de transição é uma relação que define como o autômato transita de um estado para outro com base em um símbolo de entrada.



# Componentes principais de um AF

## ◆ Estado Inicial

- Entre os estados do AF, um é designado como o estado inicial. É o estado em que o autômato se encontra no início do processamento.
- O estado inicial é o ponto de partida para qualquer processamento que o autômato realiza. Todas as operações começam a partir deste estado.

# Componentes principais de um AF

## ◆ Estados de Aceitação

- Dentro do conjunto de estados, um ou mais estados são designados como estados de aceitação (ou finais). Estes são os estados que, se alcançados ao final do processamento de uma entrada, indicam que a entrada foi aceita pelo autômato.
- Os estados de aceitação são críticos para determinar se uma sequência de entrada pertence à linguagem que o autômato está projetado para reconhecer. Se o autômato termina em um estado de aceitação após processar uma entrada, a entrada é considerada aceita.

# Reconhecimento de Linguagens por Autômatos Finitos

- ◆ O que são Linguagens Formais?
  - Uma linguagem formal é um conjunto de cadeias de caracteres (strings) formadas a partir de um alfabeto específico.

# Relação Entre Autômatos e Linguagens

- ◆ Função dos Autômatos:
  - Autômatos finitos são usados para reconhecer se uma dada string pertence a uma linguagem formal específica.

# Relação Entre Autômatos e Linguagens

- ◆ Processo de Reconhecimento:
  - Uma string é dada como entrada ao autômato.
  - O autômato processa a string, símbolo por símbolo, seguindo suas regras de transição.
  - Se o autômato chega a um estado de aceitação após processar todos os símbolos da string, a string é considerada parte da linguagem que o autômato reconhece.

# Importância e aplicabilidade dos AF

- ◆ São fundamentais em várias áreas da ciência da computação, engenharia, e outras disciplinas técnicas, devido à sua capacidade de modelar comportamentos e processos complexos de forma simples e eficaz.

# Importância e aplicabilidade das AF

- ◆ Na Modelagem de Comportamento e Processos:
  - Permitem a modelagem clara de comportamentos e processos que têm um número finito de estados distintos.
  - Útil em sistemas onde o comportamento depende do histórico de eventos ou do estado atual do sistema.

# Importância e aplicabilidade das AF

- ◆ Desenvolvimento de Software e Hardware:
  - São usadas para desenhar a lógica de controle de aplicativos, jogos, sistemas operacionais e interfaces de usuário.
  - Em hardware, são fundamentais no design de circuitos digitais, como contadores e registradores.



# Importância e aplicabilidade das AF

- ◆ Sistemas de controle:
  - Em engenharia, são aplicadas para controlar o funcionamento de máquinas e processos industriais.
  - Por exemplo, sistemas de controle de semáforos, elevadores, e máquinas automáticas.

# Importância e aplicabilidade das AF

- ◆ Telecomunicações e Redes:
  - São usadas na análise e design de protocolos de comunicação, onde diferentes estados representam diferentes modos de operação ou fases de comunicação.

# Importância e aplicabilidade das AF

- ◆ Linguagens de Programação e Compiladores:
  - No design de compiladores, ajudam a analisar a sintaxe de linguagens de programação e a implementar analisadores léxicos e sintáticos.

# Importância e aplicabilidade das AF

- ◆ Segurança Cibernética:
  - Na análise de segurança e criptografia, são utilizadas para modelar protocolos de segurança e verificar a sua integridade.

# Importância e aplicabilidade das AF

- ◆ Interface Homem-Máquina:
  - Em sistemas interativos e jogos, são usadas para gerenciar os estados de interação do usuário, como menus, configurações de jogo, ou estados de personagens.

# Importância e aplicabilidade das AF

- ◆ Automação de Testes:
  - São empregadas na automação de testes de software, onde diferentes caminhos de teste podem ser representados com diferentes estados.

# Representação de Autômatos Finitos

## ◆ Diagramas de Estados

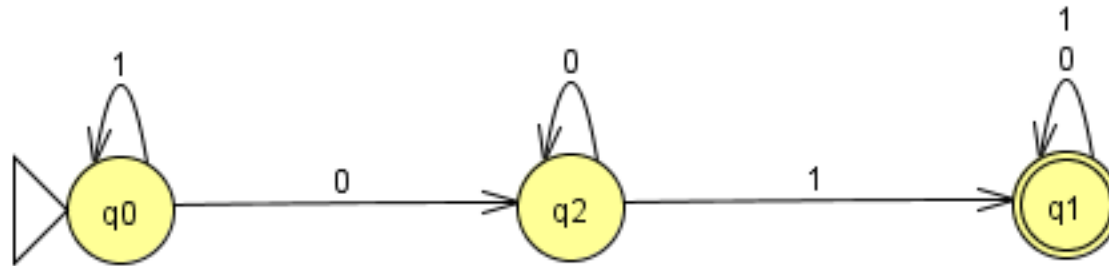
- É uma representação visual dos estados de um autômato e das transições entre esses estados.

## ◆ Componentes:

- Círculos: Representam os estados.
- Setas: Indicam as transições de um estado para outro.
- Símbolos nas Setas: Representam os símbolos de entrada que causam a transição.
- Estado Inicial: Indicado por uma seta apontando para ele sem origem.
- Estados de Aceitação: Geralmente marcados com um círculo duplo.

# Representação de Autômatos Finitos

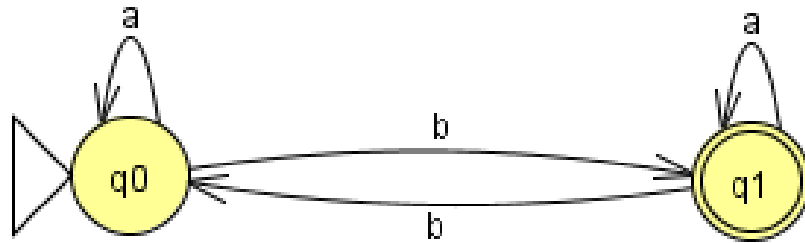
- ◆ Exemplo 1: diagrama de estados





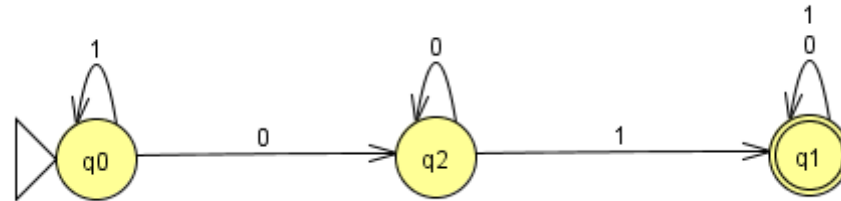
# Representação de Autômatos Finitos

- ◆ Exemplo 2: diagrama de estados



# Representação de Autômatos Finitos

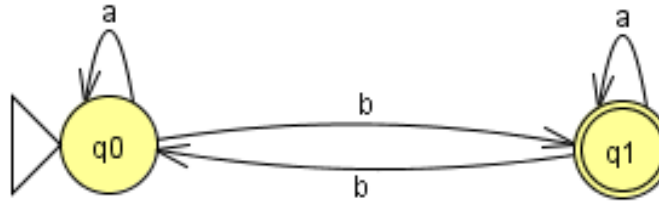
- ◆ Exemplo 1: tabela de transição



Estado Atual	Próximo Estado	
	Entrada Atual	
	0	1
Q0	Q2	Q0
Q1	Q1	Q1
Q2	Q2	Q1

# Representação de Autômatos Finitos

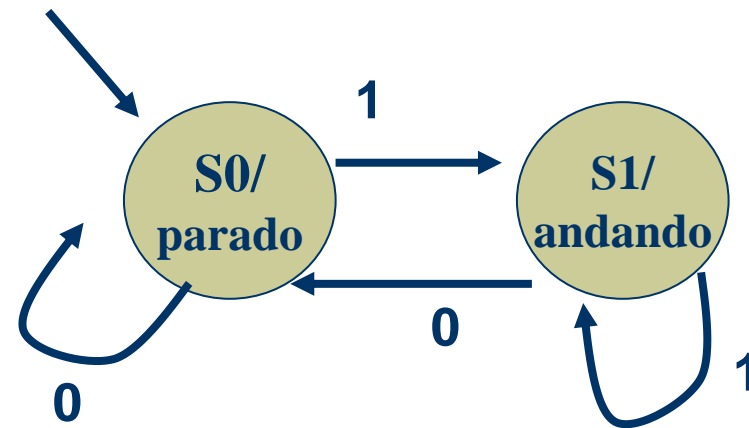
- ◆ Exemplo 2: tabela de transição



Estado Atual	Próximo Estado	
	Entrada Atual	
	a	b
Q0	Q0	Q1
Q1	Q1	Q0

# Autômato Finitos

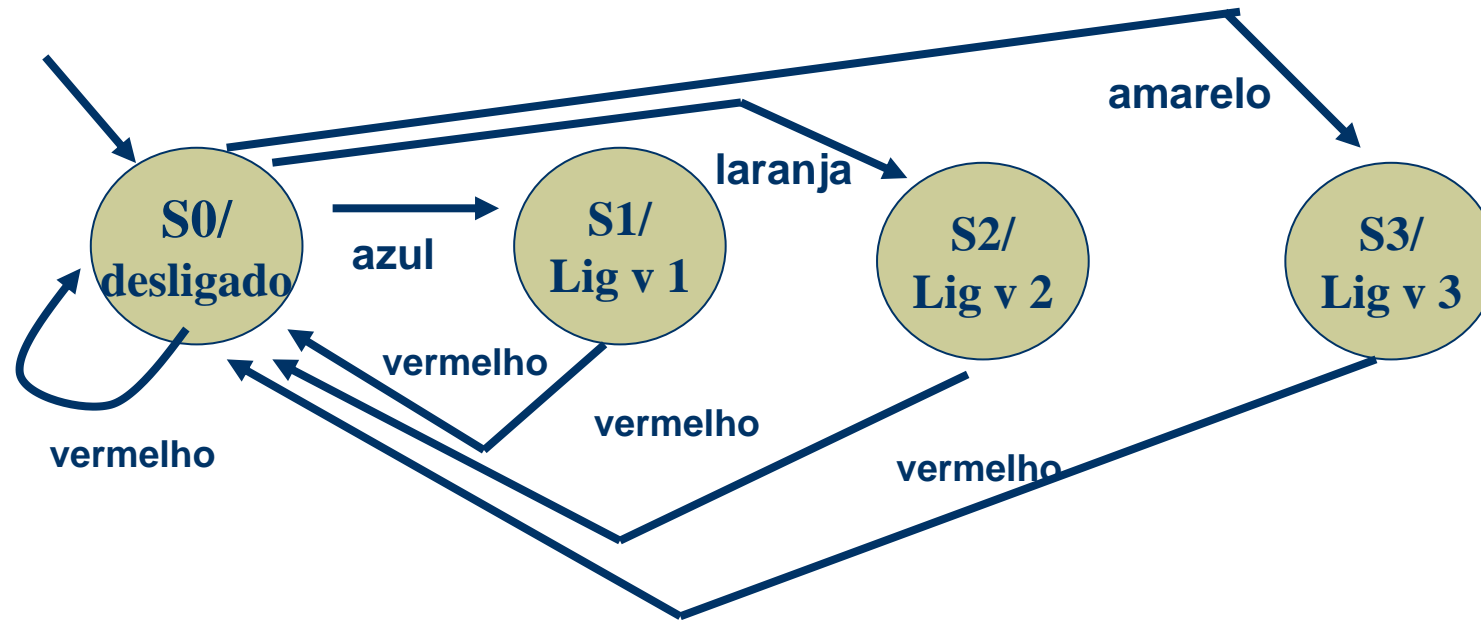
- ◆ Estados  $\{S0, S1\}$
- ◆ Entrada = (0, 1)
- ◆ Saída = (parado, andando)
- ◆ Entrada 0  $\rightarrow$  para
- ◆ Entrada 1  $\rightarrow$  anda



# Autômato Finitos

## Exemplo Ventilador com 3 velocidades

- ◆ Estados {S0,S1,S2,S3}
- ◆ Entrada=(botão vermelho, azul, laranja amarelo)
- ◆ Saída=(desligado, ligado velocidade 1, ligado velocidade 2 e ligado velocidade 3)



# Autômatos Finitos

- ◆ Para descrever um autômato finito ou uma máquina de estado finito em particular, precisamos definir os três conjuntos e as duas funções que a definem.
- ◆ Por exemplo: Seja uma máquina de estado finito  $M$ , descrita da seguinte maneira:
- ◆  $M = [S, I, O, f_i \text{ e } f_o]$
- ◆  $S = \{ s_0, s_1, s_2 \}$
- ◆  $I = \{ 0, 1 \}$
- ◆  $O = \{ 0, 1 \}$
- ◆  $f_i$  e  $f_o$  funções de input e output (transições).

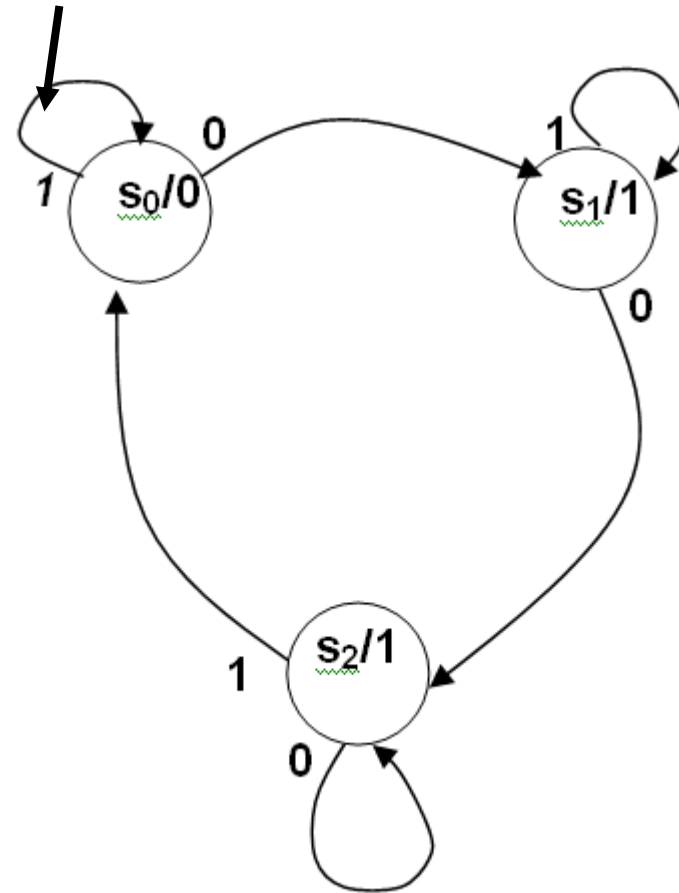
# Autômatos Finitos

Estado Atual	Próximo Estado		Saída
	Entrada Atual		
	0	1	
$s_0$	$s_1$	$s_0$	0
$s_1$	$s_2$	$s_1$	1
$s_2$	$s_2$	$s_0$	1

- ♦  $M = [S, I, O, f_i \text{ e } f_o]$
- ♦ A máquina  $M$  começa no estado  $s_0$  que tem saída 0. Se o primeiro símbolo de entrada for um 0, o próximo estado da máquina é  $s_1$ , que tem saída 1
- ♦ Se o próximo símbolo de entrada for 1, a máquina permanece no estado  $s_1$ , com saída 1.

# Diagrama de Estados ou Grafo de Estados

- ◆ Outra forma de definir as funções  $f_i$  e  $f_o$  é através de um grafo chamado grafo de estados.





# Autômatos Finitos

## Tabela de Transição

- O zero inicial da cadeia de saída é supérfluo - ele apenas reflete o estado inicial, e não o resultado de qualquer entrada.

Ciclo	$t_0$	$t_1$	$t_2$	$t_3$	$t_4$	$t_5$
Entrada	<i><b>0</b></i>	<i><b>1</b></i>	<i><b>1</b></i>	<i><b>0</b></i>	<i><b>1</b></i>	-
Estado	$s_0$	$s_1$	$s_1$	$s_1$	$s_2$	$S_0$
Saída	<u><i><b>0</b></i></u>	<i><b>1</b></i>	<i><b>1</b></i>	<i><b>1</b></i>	<i><b>1</b></i>	<i><b>0</b></i>

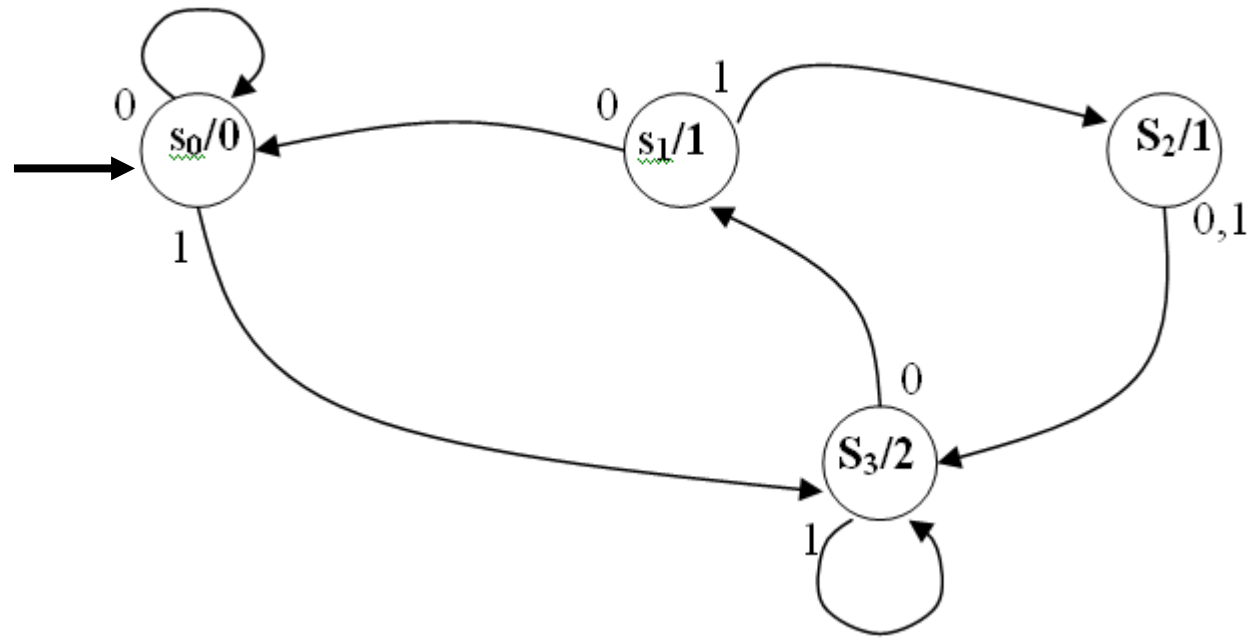
# Exemplo 1

- ◆ Para a máquina definida, reproduza a saída para a sequência de entrada 1010.
- ◆ A saída é 00111

<b>Ciclo</b>	$t_0$	$t_1$	$t_2$	$t_3$	$t_4$
<b>Entrada</b>	<i>1</i>	<i>0</i>	<i>1</i>	<i>0</i>	-
<b>Estado</b>	$s_0$	$s_0$	$s_1$	$s_1$	$s_2$
<b>Saída</b>	<i>0</i>	<i>0</i>	<i>1</i>	<i>1</i>	<i>1</i>

## Exemplo 2

- ◆ Uma máquina  $M$  é dada pelo grafo de estados.



- ◆ Forneça a tabela de estados de  $M$ .

# A Tabela de Estados

Estado Atual	Próximo Estado		Saída
	Entrada $0$	Atual $1$	
$s_0$	$s_0$	$s_3$	$0$
$s_1$	$s_0$	$s_2$	$1$
$s_2$	$s_3$	$s_3$	$1$
$s_3$	$s_1$	$s_3$	$2$

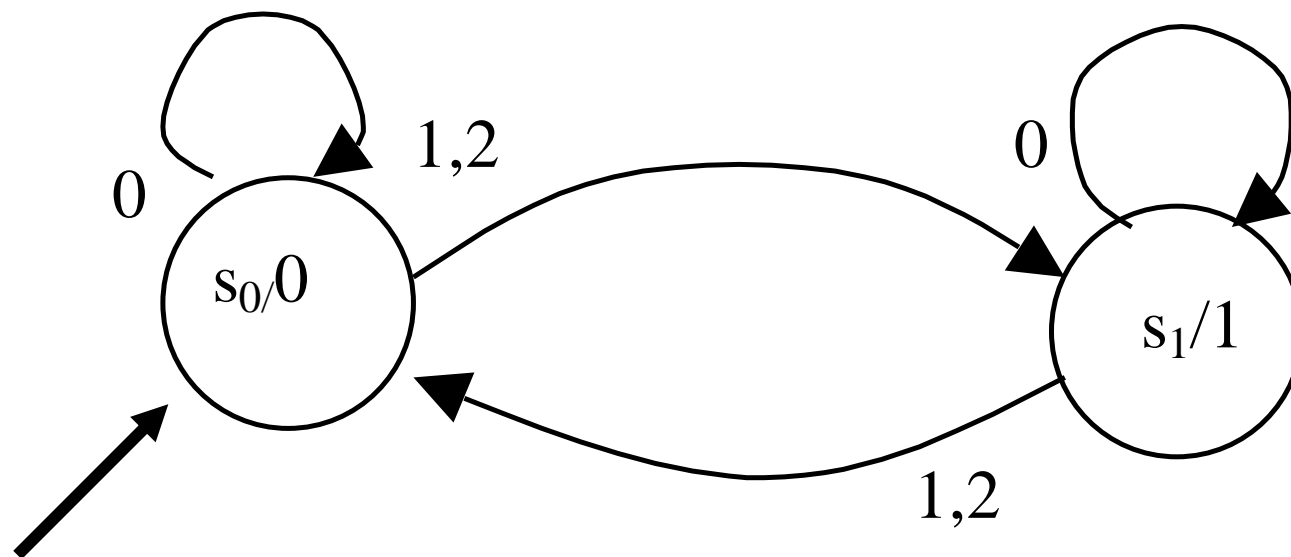
# Exemplo 3

- Uma máquina de estados é descrita pela tabela de estados:

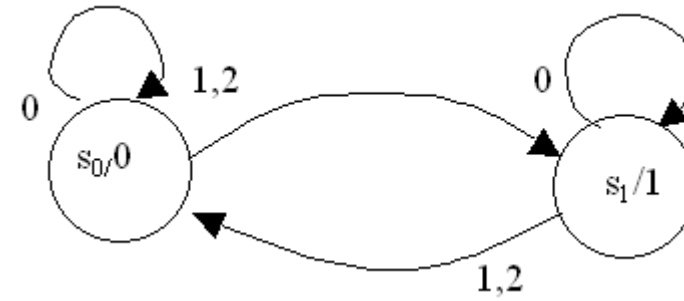
Estado Normal	Próximo Estado			Saída
	Entrada Atual			
	<i>0</i>	<i>1</i>	<i>2</i>	
<i>s<sub>0</sub></i>	<i>s<sub>0</sub></i>	<i>s<sub>1</sub></i>	<i>s<sub>1</sub></i>	<i>0</i>
<i>S<sub>1</sub></i>	<i>s<sub>1</sub></i>	<i>s<sub>0</sub></i>	<i>s<sub>0</sub></i>	<i>1</i>

- Desenhar o grafo de estados para de M

# Grafo

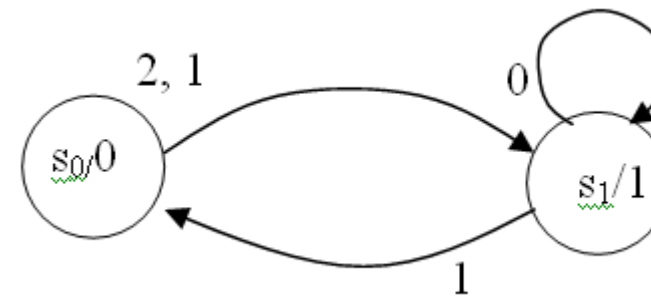


## Exemplo 4



- Qual a saída correspondente à sequência de entrada 2110? Mostrar em tabela e em grafo.

Ciclo	$t_0$	$t_1$	$t_2$	$t_3$	$t_4$
Entrada	<b>2</b>	<b>1</b>	<b>1</b>	<b>0</b>	
Estado	$s_0$	$s_1$	$s_0$	$s_1$	$s_1$
Saída	<b>0</b>	<b>1</b>	<b>0</b>	<b>1</b>	<b>1</b>



# Breve resumo do material

- ◆ Apresentou autômatos finitos como modelos matemáticos usados para projetar softwares e hardwares, consistindo em um número finito de estados e sendo capazes de reconhecer padrões em cadeias de texto.
- ◆ Mostrou os componentes de um AF, incluindo o conjunto de estados, alfabeto, função de transição, estado inicial e estados de aceitação.



# Breve resumo do material

- ◆ Apresentou o papel dos autômatos finitos no reconhecimento de linguagens formais, enfatizando como processam strings de entrada para determinar se pertencem a uma linguagem específica.
- ◆ Destacou a importância dos AF em várias áreas da ciência da computação, engenharia e outras disciplinas técnicas, incluindo a modelagem de comportamentos e processos, desenvolvimento de software e hardware, sistemas de controle, telecomunicações e redes, linguagens de programação e compiladores, e segurança cibernética.

# Breve resumo do material

- ◆ Apresentou como os grafos de estados são usados para definir as funções de transição e saída de autômatos finitos
- ◆ Apresentou os métodos de representação de autômatos finitos, incluindo diagramas de estados e tabelas de transição, com exemplos específicos.

# Exercício 1

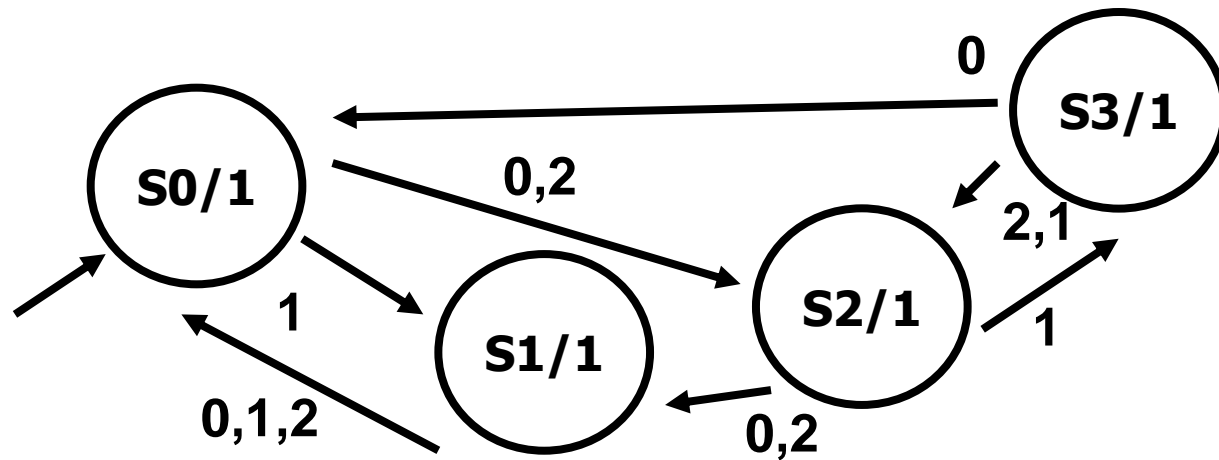
- ♦ Uma máquina M é descrita pela tabela de estados:

Estado atual	Próximo Estado			Saída
	Entrada Atual			
	<i>0</i>	<i>1</i>	<i>2</i>	
$s_0$	$s_0$	$s_2$	$s_1$	<i>0</i>
$S_1$	$S_1$	$S_0$	$S_2$	<i>1</i>
$S_2$	$S_1$	$S_2$	$S_0$	<b>0</b>

- ♦ A – Criar o grafo de estado para M.
- ♦ B – Qual a sequência de saída que corresponde a entrada 211201
- ♦ C – Qual a sequência de saída que corresponde a entrada 212221

## Exercício 2

- ♦ Uma máquina M é descrita pelo seguinte grafo:



- ♦ A – Criar a tabela de estados para M.
- ♦ B – Qual a sequência de saída que corresponde a entrada 211201
- ♦ C – Qual a sequência de saída que corresponde a entrada 212221

# Exercício 3

- ♦ Com base na tabela de estados abaixo:
  - A – criar o grafo.
  - B – verificar a saída gerada a partir da entrada 00110101110, para isso gerar a tabela de transição.

Estado Atual	Próximo Estado		Saída
	Entrada Atual		
	0	1	
$S_0$	$S_1$	$S_0$	0
$S_1$	$S_3$	$S_2$	1
$S_2$	$S_4$	$S_5$	1
$S_3$	$S_3$	$S_4$	0
$S_4$	$S_2$	$S_0$	0
$S_5$	$S_3$	$S_4$	0

# Exercício 4

- ◆ Baseado na tabela de estados:

Estado Atual	Próximo Estado				Saída
	Entrada Atual				
	0	1	2	3	
$S^0$	$S^1$	$S^5$	$S^0$	$S^0$	0
$S^1$	$S^3$	$S^4$	$S^3$	$S^2$	1
$S^2$	$S^4$	$S^5$	$S^4$	$S^5$	1
$S^3$	$S^3$	$S^3$	$S^0$	$S^0$	2
$S^4$	$S^2$	$S^0$	$S^2$	$S^4$	2
$S^5$	$S^3$	$S^4$	$S^1$	$S^0$	0

## Exercício 4

- ◆ Criar o grafo.
- ◆ Criar a tabela de transição para verificar a saída gerada pela máquina com as entradas:
  - 0000013333
  - 333012222
  - 012301233210

# Referência desta aula

- ◆ HOPCROFT, John E.; MOTWANI, Rajeev; ULLMAN, Jeffrey D. Introdução a teoria de autômatos, linguagens e computação. Rio de Janeiro: Campus, 2002.

FIM

Obrigado

Rodrigo