

Formulário

Responda o formulário (anônimo) para que eu possa saber um pouco mais sobre vocês 🙏.



<http://abre.ai/2020-lfa1>

Introdução ao curso

MCTA015-13 - Linguagens Formais e Autômatas

Profa. Carla Negri Lintzmayer

`carla.negri@ufabc.edu.br`

`www.professor.ufabc.edu.br/~carla.negri`

Centro de Matemática, Computação e Cognição – Universidade Federal do ABC



É uma área da Ciência da Computação que tenta responder
“quais são as capacidades e limitações fundamentais dos computadores?”

É uma área da Ciência da Computação que tenta responder
“quais são as capacidades e limitações fundamentais dos computadores?”

- O que é computação?

É uma área da Ciência da Computação que tenta responder
“quais são as capacidades e limitações fundamentais dos computadores?”

- O que é computação?
 - Teoria dos Autômatos

É uma área da Ciência da Computação que tenta responder
“quais são as capacidades e limitações fundamentais dos computadores?”

- O que é computação?
 - Teoria dos Autômatos
- O que pode ser computado?

É uma área da Ciência da Computação que tenta responder
“quais são as capacidades e limitações fundamentais dos computadores?”

- O que é computação?
 - Teoria dos Autômatos
- O que pode ser computado?
 - Teoria da Computabilidade

É uma área da Ciência da Computação que tenta responder
“quais são as capacidades e limitações fundamentais dos computadores?”

- O que é computação?
 - Teoria dos Autômatos
- O que pode ser computado?
 - Teoria da Computabilidade
- O quão bem pode ser computado?

É uma área da Ciência da Computação que tenta responder
“quais são as capacidades e limitações fundamentais dos computadores?”

- O que é computação?
 - Teoria dos Autômatos
- O que pode ser computado?
 - Teoria da Computabilidade
- O quão bem pode ser computado?
 - Teoria da Complexidade

Computação

O que é Computação (informal)?

O que é Computação (informal)?

O que é Computação (informal)?

O que é Computação (informal)?

- É o processo de solução de um **problema** por meio de um **algoritmo**

Tipos de problemas

Problemas vêm em vários sabores:

- Decisão
- Otimização
- Contagem
- Busca
- etc



Decisão: problema cuja resposta é sim ou não.

Decisão: problema cuja resposta é sim ou não.

- Dado um número, ele é par?

Decisão: problema cuja resposta é sim ou não.

- Dado um número, ele é par?
- Dado um número, ele é primo?

Decisão: problema cuja resposta é sim ou não.

- Dado um número, ele é par?
- Dado um número, ele é primo?
- Dados dois vértices em um grafo, existe caminho entre eles?

Decisão: problema cuja resposta é sim ou não.

- Dado um número, ele é par?
- Dado um número, ele é primo?
- Dados dois vértices em um grafo, existe caminho entre eles?
- Dados dois vértices em um grafo, existe caminho com no máximo k arestas entre eles?

Otimização: problema que procura a melhor resposta dentre todas as possíveis.

Otimização: problema que procura a melhor resposta dentre todas as possíveis.

- Dados dois números, qual o maior divisor comum de ambos?

Otimização: problema que procura a melhor resposta dentre todas as possíveis.

- Dados dois números, qual o maior divisor comum de ambos?
- Qual o menor caminho entre dois vértices?

Otimização: problema que procura a melhor resposta dentre todas as possíveis.

- Dados dois números, qual o maior divisor comum de ambos?
- Qual o menor caminho entre dois vértices?
- Qual o maior caminho entre dois vértices?

Otimização: problema que procura a melhor resposta dentre todas as possíveis.

- Dados dois números, qual o maior divisor comum de ambos?
- Qual o menor caminho entre dois vértices?
- Qual o maior caminho entre dois vértices?
- Dado um conjunto de itens valorados e um recipiente com capacidade finita, qual o maior valor obtido ao escolher itens que caibam no recipiente?

Contagem: problema cuja resposta é o número de soluções possíveis.

Contagem: problema cuja resposta é o número de soluções possíveis.

- Dado um número, quantos divisores ele possui?

Contagem: problema cuja resposta é o número de soluções possíveis.

- Dado um número, quantos divisores ele possui?
- Dado um número, quantos fatores primos ele possui?

Contagem: problema cuja resposta é o número de soluções possíveis.

- Dado um número, quantos divisores ele possui?
- Dado um número, quantos fatores primos ele possui?
- Dados dois vértices em um grafo, quantos caminhos existem entre ambos?

Tipos de problemas

Qualquer tipo de problema pode ser transformado em (reduzido a) um problema de decisão equivalente.

Tipos de problemas

Qualquer tipo de problema pode ser transformado em (reduzido a) um problema de decisão equivalente.

Caixeiro Viajante – Otimização

- **Entrada:** Grafo G e função de custos $c : E(G) \rightarrow \mathbb{R}^+$.
- **Saída:** Um ciclo hamiltoniano sobre G de custo mínimo.

Tipos de problemas

Qualquer tipo de problema pode ser transformado em (reduzido a) um problema de decisão equivalente.

Caixeiro Viajante – Otimização

- **Entrada:** Grafo G e função de custos $c : E(G) \rightarrow \mathbb{R}^+$.
- **Saída:** Um ciclo hamiltoniano sobre G de custo mínimo.

Caixeiro Viajante – Decisão

- **Entrada:** Grafo G , função de custos $c : E(G) \rightarrow \mathbb{R}^+$, e número $r \in \mathbb{R}^+$.
- **Saída:** SIM se existe ciclo hamiltoniano em G de custo menor do que r ; NÃO caso contrário.

Tipos de problemas

Qualquer tipo de problema pode ser transformado em (reduzido a) um problema de decisão equivalente.

Caixeiro Viajante – Otimização

- **Entrada:** Grafo G e função de custos $c : E(G) \rightarrow \mathbb{R}^+$.
- **Saída:** Um ciclo hamiltoniano sobre G de custo mínimo.

Caixeiro Viajante – Decisão

- **Entrada:** Grafo G , função de custos $c : E(G) \rightarrow \mathbb{R}^+$, e número $r \in \mathbb{R}^+$.
- **Saída:** SIM se existe ciclo hamiltoniano em G de custo menor do que r ; NÃO caso contrário.

Por isso, podemos nos restringir a problemas de decisão.

Sobre o curso

Tudo é número string!

Strings podem representar qualquer objeto:

- Números: 5 \rightarrow "5" ou "00110101" ou "101", etc ...

Tudo é número string!

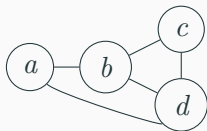
Strings podem representar qualquer objeto:

- Números: $5 \rightarrow "5"$ ou $"00110101"$ ou $"101"$, etc ...
- Polinômios: $4x^2 + 3y - 5 \rightarrow "4x^2 + 3y - 5"$ ou $"3,2,4x2+3y1-5y0"$...

Tudo é número string!

Strings podem representar qualquer objeto:

- Números: $5 \rightarrow "5"$ ou $"00110101"$ ou $"101"$, etc ...
- Polinômios: $4x^2 + 3y - 5 \rightarrow "4x^2 + 3y - 5"$ ou $"3,2,4x2+3y1-5y0"$...
- Grafos:

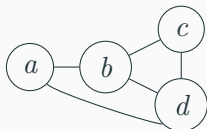


$\rightarrow "(\{a,b,c,d\}, \{ab, bc, cd, bd, ad\})"$

Tudo é número string!

Strings podem representar qualquer objeto:

- Números: $5 \rightarrow "5"$ ou $"00110101"$ ou $"101"$, etc ...
- Polinômios: $4x^2 + 3y - 5 \rightarrow "4x^2 + 3y - 5"$ ou $"3,2,4x^2+3y1-5y0"$...
- Grafos:



$\rightarrow "(\{a,b,c,d\}, \{ab, bc, cd, bd, ad\})"$

- Imagens:



107	102	174	149	150	152	120	151	172	161	166	166
106	162	143	74	76	62	66	57	112	210	180	184
100	140	62	14	34	6	12	22	40	100	100	101
206	100	0	134	131	111	120	204	160	10	50	100
194	60	110	200	207	200	200	220	217	67	101	
173	100	207	200	207	214	205	200	200	16	74	206
100	60	176	200	180	216	211	180	190	74	80	100
100	0	144	140	80	140	100	11	30	62	22	100
100	100	181	160	150	227	170	140	160	63	90	100
200	174	140	200	206	207	160	170	220	63	210	204
100	216	140	140	200	167	100	100	70	60	210	241
100	204	147	160	227	210	160	90	111	100	204	
100	214	172	60	140	140	90	7	100	100	210	
107	100	200	100	1	10	67	0	217	100	211	
100	100	207	100	6	12	100	200	100	140	100	
100	200	200	207	100	100	100	170	10	100	100	
107	102	174	149	150	152	120	151	172	161	166	166
106	162	143	74	76	62	66	57	112	210	180	184
100	140	62	14	34	6	12	22	40	100	100	101
206	100	0	134	131	111	120	204	160	10	50	100
194	60	110	200	207	200	200	220	217	67	101	
173	100	207	200	207	214	205	200	200	16	74	206
100	60	176	200	180	216	211	180	190	74	80	100
100	0	144	140	80	140	100	11	30	62	22	100
100	100	181	160	150	227	170	140	160	63	90	100
200	174	140	200	206	207	160	170	220	63	210	204
100	216	140	140	200	167	100	100	70	60	210	241
100	204	147	160	227	210	160	90	111	100	204	
100	214	172	60	140	140	90	7	100	100	210	
107	100	200	100	1	10	67	0	217	100	211	
100	100	207	100	6	12	100	200	100	140	100	
100	200	200	207	100	100	100	170	10	100	100	
107	102	174	149	150	152	120	151	172	161	166	166
106	162	143	74	76	62	66	57	112	210	180	184
100	140	62	14	34	6	12	22	40	100	100	101
206	100	0	134	131	111	120	204	160	10	50	100
194	60	110	200	207	200	200	220	217	67	101	
173	100	207	200	207	214	205	200	200	16	74	206
100	60	176	200	180	216	211	180	190	74	80	100
100	0	144	140	80	140	100	11	30	62	22	100
100	100	181	160	150	227	170	140	160	63	90	100
200	174	140	200	206	207	160	170	220	63	210	204
100	216	140	140	200	167	100	100	70	60	210	241
100	204	147	160	227	210	160	90	111	100	204	
100	214	172	60	140	140	90	7	100	100	210	
107	100	200	100	1	10	67	0	217	100	211	
100	100	207	100	6	12	100	200	100	140	100	
100	200	200	207	100	100	100	170	10	100	100	

Linguagem formal

Qualquer *conjunto de strings* sobre um alfabeto.

$$L = \{010, 11, 0, 100, 01\}$$

$$T = \{\textit{banana}, \textit{melancia}, \textit{abacate}, \textit{cereja}\}$$

$$C = \{\textit{if}, \textit{while}, \textit{int}, \textit{main}\}$$

Qualquer problema de decisão pode ser representado por um conjunto que contém strings que representam suas instâncias sim.

- $\mathcal{P} = \{2, 3, 5, 7, 11, 13, 17, 19, \dots\}$
- Decidir se um número n é primo \equiv decidir se $n \in \mathcal{P}$

Qualquer problema de decisão pode ser representado por um conjunto que contém strings que representam suas instâncias sim.

- $\mathcal{P} = \{2, 3, 5, 7, 11, 13, 17, 19, \dots\}$
- Decidir se um número n é primo \equiv decidir se $n \in \mathcal{P}$

Em outras palavras, **linguagens formalizam problemas.**

Qualquer problema de decisão pode ser representado por um conjunto que contém strings que representam suas instâncias sim.

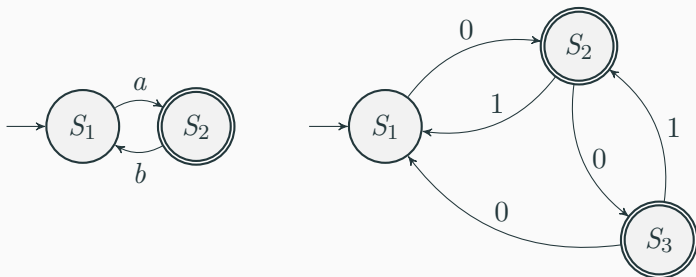
- $\mathcal{P} = \{2, 3, 5, 7, 11, 13, 17, 19, \dots\}$
- Decidir se um número n é primo \equiv decidir se $n \in \mathcal{P}$

Em outras palavras, **linguagens formalizam problemas.**

Computação (formal): decidir se uma string pertence à linguagem.

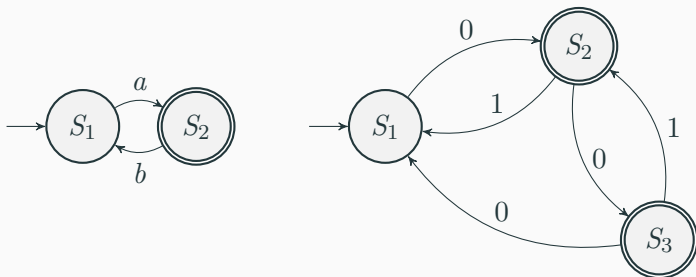
Autômatos

Máquinas (modelos matemáticos) abstratas que reconhecem palavras de uma linguagem.



Autômatos

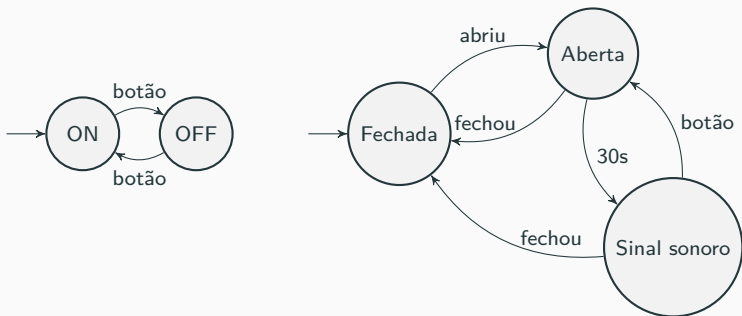
Máquinas (modelos matemáticos) abstratas que reconhecem palavras de uma linguagem.



- Quais linguagens podem ser reconhecidas em cada autômato?

Autômatos (ou máquinas de estados)

- Abstraem detalhes de implementação dos computadores.
- Introduzem conceitos relevantes a outras áreas e aplicações:
 - Projeto de linguagens de programação (compiladores)
 - Processamento de linguagem natural
 - Análise de texto (Parsing)
 - Projeto e verificação de circuitos e sistemas digitais



Máquinas de Turing

Autômatos com a mesma capacidade de computação dos computadores de hoje.

Máquinas de Turing

Autômatos com a mesma capacidade de computação dos computadores de hoje.

Explorar os limites da computação:



Máquinas de Turing

Autômatos com a mesma capacidade de computação dos computadores de hoje.

Explorar os limites da computação:



- Decidibilidade (o que pode ser computado?)

Máquinas de Turing

Autômatos com a mesma capacidade de computação dos computadores de hoje.

Explorar os limites da computação:



- Decidibilidade (o que pode ser computado?)
 - Problema da Parada: Dado um programa P e uma entrada I para P . O programa P quando alimentado com a entrada I para ou entre em *loop*?

Máquinas de Turing

Autômatos com a mesma capacidade de computação dos computadores de hoje.


Explorar os limites da computação:



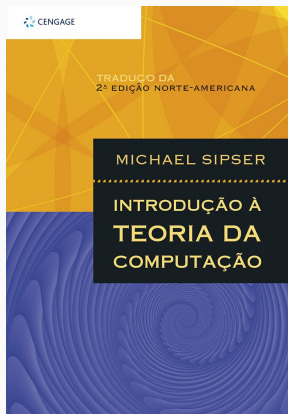
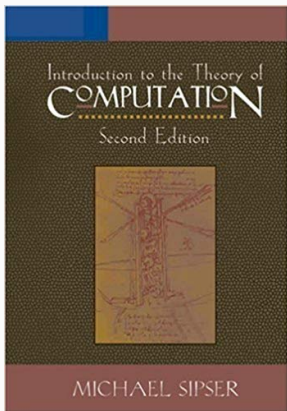
- Decidibilidade (o que pode ser computado?)
 - Problema da Parada: Dado um programa P e uma entrada I para P . O programa P quando alimentado com a entrada I para ou entre em *loop*?
- Tratabilidade (quão eficiente algo pode ser computado?)
 - Complexidade
 - \mathcal{P} vs \mathcal{NP}

- **Oficial:** Programação Estruturada

- **Oficial:** Programação Estruturada
- **Real:** Matemática Discreta e Algoritmos e Estruturas de Dados 1

- **Oficial:** Programação Estruturada
- **Real:** Matemática Discreta e Algoritmos e Estruturas de Dados 1
- **Sonho da professora** : Matemática Discreta, Algoritmos e Estruturas de Dados 1, Análise de Algoritmos e Teoria dos Grafos.

- Conteúdo no quadro
 - Recursos extras sempre estarão disponíveis no site
 - Seguem o livro do Sipser, “Introdução à teoria da computação”, disponível na biblioteca



- Dependem da sua participação
 - Qualquer pergunta é sempre bem-vinda
 - Feedbacks também
 - Não deixe dúvidas acumularem

- Dependem da sua participação
 - Qualquer pergunta é sempre bem-vinda
 - Feedbacks também
 - Não deixe dúvidas acumularem
- Espero que você
 - Respeite os horários
 - Faça as listas de exercícios
 - Seja autor das suas soluções
 - Use os horários de atendimento (procure ajuda)
 - Não assine a lista de presença por outros



professor.ufabc.edu.br/~carla.negri/cursos/2020Q1-LFA/

- Estude bem o conteúdo do site.
- Verifique-o com frequência!