Linguagens Formais e Problemas de Decisão

Área de Conhecimento em Algoritmos e Teoria - DCC/UFMG

Fundamentos de Teoria da Computação

2021/2

Introdução

- No dia-a-dia estamos acostumados com **linguagens naturais**, como português, inglês, francês, ...

  Estas linguagens nos permitem descrever objetos, através de **cadeias** (ou **palavras**).

- Aqui vamos estudar **linguagens formais**, que são conjuntos de cadeias matematicamente construídas para descrever objetos abstratos.

- Uma das aplicações mais relevantes de linguagens formais em ciência da computação é a <u>definição de linguagens de programação</u>.

  - Você já pensou que existe uma "linguagem dos programas em C", e que cada "cadeia" é um programa válido em C?

  - Como definimos esta linguagem? Qual é a sua "gramática"?

  Como aqui vamos lidar apenas com linguagens formais, podemos nos referir a elas apenas como "linguagens".

- Em particular, vamos ver como linguagens formais estão associadas a **problemas de decisão**, que são um dos cernes da ciência da computação.

Linguagens Formais

Linguagens: Conceitos fundamentais

- Um **alfabeto** é um conjunto finito não-vazio de elementos chamados **símbolos**.

- $\boxed{\text{Exemplo 1}}$ Exemplos de alfabetos:

  (a) $\{0, 1\}$ é o **alfabeto binário**,

  (b) $\{\mathtt{a}, \mathtt{b}, \mathtt{c}, \ldots, \mathtt{z}\}$ é o alfabeto de letras latinas minúsculas,

  (c) $\{\mathtt{a}_1, \mathtt{a}_2, \mathtt{a}_3\}$ é um **alfabeto ternário** (de apenas três símbolos),

  (d) $\{\square, \triangle, \bigcirc, \times\}$ é o alfabeto das 4 teclas principais do controle do Playstation,

  (e) o conjunto de caracteres do teclado é um alfabeto.

$\triangleleft$

Notação: Normalmente usamos a letra Σ (sigma) para denotar um alfabeto.

- Uma **cadeia** (ou **palavra**) de um alfabeto Σ é uma sequência finita de símbolos de Σ.

- Exemplo 2 Exemplos de cadeias no alfabeto $\{0, 1\}$:

  (a) 0

  (b) 011

  (c) 1101010001

  (d) ϵ

$\triangleleft$

- O símbolo ϵ (épsilon) denota a **cadeia vazia**, que é formada por 0 símbolos.
- $|w|$ denota o **tamanho** (i.e., o número de símbolos) da cadeia w.

Exemplo 3

(a) $|0| = 1$

(b) $|011| = 3$

(c) $|1101010001| = 10$

(d) $|\epsilon| = 0$

$\triangleleft$

Linguagens: Conceitos fundamentais

- $\mathtt{a}^n$ denota n ocorrências do símbolos a em sequência.

- $\boxed{\text{Exemplo 4}}$

  (a) $1^4 = 1111$ (b) $1^0 = \epsilon$ (c) $\mathtt{a}^2\mathtt{b}^3\mathtt{a}^2 = \mathtt{aabbbaa}$

  $\triangleleft$

- Σ^* (lê-se "sigma estrela") denota o conjunto de todas as cadeias que podem ser formadas com o alfabeto Σ.

- $\boxed{\text{Exemplo 5}}$

  (a) $\{\mathtt{a}\}^* = \{\epsilon, \mathtt{a}, \mathtt{aa}, \mathtt{aaa}, \mathtt{a}^4, \mathtt{a}^5, \ldots\}$

  (b) $\{0, 1\}^* = \{\epsilon, 0, 1, 00, 01, 10, 11, 000, 001, 010, 011, 100, 101, \ldots\}$

  $\triangleleft$

Linguagens: Conceitos fundamentais

- Uma **linguagem** sobre um alfabeto Σ é um subconjunto de Σ^*.
- $\boxed{\text{Exemplo 6}}$ Exemplos de linguagens sobre o alfabeto $\{0, 1\}$:

(a) $\{0, 1, 00, 11\}$

(b) $\emptyset$

(c) $\{\epsilon\}$

(d) $\{\epsilon, 0, 1\}$

(e) $\{\epsilon, 0^2, 0^4, 0^6, 0^8, 0^{10}, \dots\}$

(f) $\{w \in \Sigma^* \mid |w| \text{ é múltiplo de } 3\}$

(g) $\{0^n 1^n \mid n \geq 0\}$

(h) $\{0^n \mid n \text{ é um número primo}\}$

(i) $\{0, 1\}^*$

$\triangleleft$

Linguagens: Conceitos fundamentais

- Como linguagens são conjuntos, operações de conjuntos também se aplicam a linguagens.

  Sejam L_1 e L_2 linguagens sobre os alfabetos Σ_1 e Σ_2, respectivamente.

  Então:

  - $L_1 \cup L_2$ é uma linguagem sobre o alfabeto $\Sigma_1 \cup \Sigma_2$,
  - $L_1 \cap L_2$ é uma linguagem sobre o alfabeto $\Sigma_1 \cap \Sigma_2$,
  - $\overline{L_1} = \{w \in \Sigma_1^* \mid w \notin L_1\}$ é uma linguagem sobre o alfabeto Σ_1,
  - $L_1 - L_2 = L_1 \cap \overline{L_2}$ é uma linguagem sobre o alfabeto Σ_1,
  - $\mathcal{P}(L_1)$ é um <u>conjunto</u> de linguagens sobre Σ_1,
  - $\mathcal{P}(\Sigma_1^*)$ é o **conjunto de todas as linguagens sobre** Σ_1.

- $\boxed{\text{Exemplo 7}}$ Sejam as linguagens

$$L_1 = \{w \in \{0,1\}^* \mid |w| \text{ é par}\}$$

e

$$L_2 = \{w \in \{0,1\}^* \mid w \text{ começa com } 0\}.$$

(a) $L_1 \cup L_2 = \{w \in \{0,1\}^* \mid |w| \text{ é par ou } w \text{ começa com } 0\}$.

(b) $L_1 \cap L_2 = \{w \in \{0,1\}^* \mid |w| \text{ é par e } w \text{ começa com } 0\}$.

(c) $\overline{L_1} = \{w \in \{0,1\}^* \mid |w| \text{ é ímpar}\}$.

(d) $L_1 - L_2 = \{w \in \{0,1\}^* \mid |w| \text{ é par e } w \text{ não começa com } 0\}$.

(e) $\mathcal{P}(L_1)$ é o conjunto de todas as linguagens em que as cadeias da linguagem têm tamanho par.

(f) $\mathcal{P}(\{0,1\}^*)$ é o conjunto de todas as linguagens binárias.

- Dadas duas cadeias

$$x = a_1 a_2 \dots a_n \qquad \text{e} \qquad y = b_1 b_2 \dots b_m,$$

sua **concatenação** é

$$xy = a_1 a_2 \dots a_n b_1 b_2 \dots b_m.$$

- Exemplo 8

  (a) Se $x = 010$ e $y = 1111$, então

  $$xy = 0101111.$$

  (b) Para toda cadeia x:

  $$\epsilon x = x\epsilon = x.$$

  (c) Para toda cadeia x, y, z:

  $$(xy)z = x(yz) = xyz.$$

Linguagens: Conceitos fundamentais

- Um **prefixo** de uma cadeia w é uma cadeia x tal que $w = xy$ para algum y.
- Um **sufixo** de uma cadeia w é uma cadeia y tal que $w = xy$ para algum x.
- Uma **subcadeia** de uma cadeia w é uma cadeia z tal que $w = xzy$ para algum x, y.
- Exemplo 9

  (a) Prefixos de `abc`: ϵ, `a`, `ab`, `abc`.

  (b) Sufixos de `abc`: ϵ, `c`, `bc`, `abc`.

  (c) Subcadeias de `abc`: ϵ, `a`, `b`, `c`, `ab`, `bc`, `abc`. $\triangleleft$

Linguagens: Conceitos fundamentais

- O **reverso** de uma cadeia $w = a_1 a_2 \dots a_n$ é a cadeia $w^R = a_n \dots a_2 a_1$.
- $\boxed{\text{Exemplo 10}}$

(a) $(\texttt{abc})^R = \texttt{cba}$ (b) $\texttt{a}^R = \texttt{a}$ (c) $\epsilon^R = \epsilon$

$\lhd$

- Um **palíndromo** é uma cadeia w tal que $w = w^R$.
- $\boxed{\text{Exemplo 11}}$ As seguintes cadeias são palíndromos.

(a) 01010 (b) $\texttt{a}$ (c) ϵ

$\lhd$

Linguagens: Conceitos fundamentais

- Dadas duas linguagens L_1 e L_2, sua **concatenação** é

$$L_1 L_2 = \{xy \mid x \in L_1 \text{ e } y \in L_2\}.$$

- $\boxed{\text{Exemplo 12}}$ Sejam

$$L_1 = \{w \in \{0,1\}^* \mid |w| = 5\}$$

e

$$L_2 = \{0y \mid y \in \{0,1\}^*\}.$$

(a) $L_1 L_1 = \{w \in \{0,1\}^* \mid |w| = 10\}$

(b) $L_1 L_2 = \{w \in \{0,1\}^* \mid |w| \geq 6 \text{ e o sexto símbolo de } w \text{ é } 0\}$

(c) $L_2 L_1 = \{w \in \{0,1\}^* \mid |w| \geq 6 \text{ e } w \text{ começa com } 0\}$

(d) $L_2 L_2 = \{0y \mid y \in \{0,1\}^* \text{ e } y \text{ contém pelo menos um } 0\}$

(e) $L_1\{\epsilon\} = L_1$

(f) $L_1\emptyset = \emptyset$

Linguagens: Conceitos fundamentais

- L^n denota a linguagem L concatenada com ela mesma n vezes.

  Em particular,

  - $L^1 = L$
  - $L^0 = \{\epsilon\}$

- $\boxed{\text{Exemplo 13}}$

  (a) $\{0, 1\}^3 = \{000, 001, 010, 011, 101, 101, 110, 111\}$

  (b) $\{\texttt{aa}, \texttt{b}\}^2 = \{\texttt{aaaa}, \texttt{aab}, \texttt{baa}, \texttt{bb}\}$

  $\triangleleft$

- O **fecho de Kleene**, ou **estrela**, de uma linguagem L é

$$L^* = L^0 \cup L^1 \cup L^2 \cup L^3 \cup L^4 \ldots$$

- O **fecho positivo de Kleene**, ou **mais**, de uma linguagem L é

$$L^+ = LL^* = L^1 \cup L^2 \cup L^3 \cup L^4 \ldots$$

Linguagens: Conceitos fundamentais

- $\boxed{\text{Exemplo 14}}$ Exemplos de fechos de Kleene e fechos positivos de Kleene:

(a) $\{0,1\}^* = \{\epsilon, 0, 1, 00, 01, 10, 11, 000, 001, \ldots\}$

(b) $\{0\}^* = \{0^n \mid n \geq 0\}$

(c) $\{0\}^+ = \{0^n \mid n \geq 1\}$

(d) $\{00\}^* = \{w \in \{0\}^* \mid |w| \text{ é par}\}$

(e) $\{00\}^+ = \{w \in \{0\}^* \mid |w| \text{ é par e } |w| \geq 2\}$

(f) $\{01, 1\}^* = \{w \in \{0,1\}^* \mid \text{todo 0 em } w \text{ é seguido de pelo menos um 1}\}$

(g) $\{\epsilon, 00, 11\}^* = \{\epsilon, 00, 11\}^+ = \{\epsilon\} \cup \{00, 11\}^+$

(h) $\{\epsilon\}^* = \{\epsilon\}^+ = \{\epsilon\}$

(i) $\emptyset^* = \{\epsilon\}$

(j) $\emptyset^+ = \emptyset$

$\triangleleft$

Linguagens: Conceitos fundamentais

- Como todo alfabeto Σ é finito, o conjunto Σ^* é enumerável. (Por quê?)

  Como toda linguagem é um subconjunto de Σ^*,
  toda linguagem é enumerável.

- Logo, pelo menos em princípio, é possível dar uma definição recursiva para algumas linguagens.

  (Não é possível dar definição recursiva para conjuntos não-enumeráveis. Por quê?)

  Mais para frente veremos como prover definições recursivas para linguagens, usando o conceito de **gramáticas**.

Problemas de Decisão

Problemas de decisão

- Um **problema de decisão (PD)** é uma questão que:
  1. faz referência a um conjunto de parâmetros, e que
  2. para cada atribuição de valores a todos os parâmetros, possui como resposta "<u>SIM</u>" ou "<u>NÃO</u>".

- $\boxed{\text{Exemplo 15}}$ Exemplos de problemas de decisão:
  (a) Determinar se um natural n é primo. (1 parâmetro: n.)
  (b) Determinar se $x + y = z$ para $x, y, z \in \mathbb{Z}$. (3 parâmetros: x, y, z.)
  (c) Determinar se 234 456 777 é primo. (0 parâmetros.)
  (d) Determinar se existe um ciclo de tamanho pelo menos k num grafo G. (2 parâmetros: k, G.)
  (e) Determinar se uma cadeia w pertence a uma linguagem L. (2 parâmetros: w, L.)

$\triangleleft$

Problemas de decisão

- Uma instância de um PD é uma questão obtida ao atribuirmos a todos os parâmetros do PD valores específicos.

- $\boxed{\text{Exemplo 16}}$ O PD de se determinar se um natural n é primo tem infinitas instâncias:
  - Determinar se 0 é primo.
  - Determinar se 1 é primo.
  - Determinar se 2 é primo.
  - Determinar se 3 é primo.
  - . . . $\lhd$

- $\boxed{\text{Exemplo 17}}$ O PD de se determinar se 234 456 777 é primo tem uma única instância. $\lhd$

Problemas de decisão

- Uma **solução para um problema de decisão** é um procedimento (ou algoritmo) que retorna a resposta correta para toda e qualquer instância do problema.
- Um PD é **decidível** se ele tem solução, e **indecidível** em caso contrário.
- Todo PD com finitas instâncias é decidível. (Por quê?)

Linguagens Formais e Problemas de Decisão

Linguagens Formais e Problemas de Decisão

- Aqui nos concentramos em um tipo muito importante de problema de decisão:

  "A cadeia w pertence à linguagem L?"

- Um algoritmo capaz de responder corretamente a todas as instâncias de w do PD acima é um **decisor** para a linguagem L.
  - Mais para a frente no curso vamos estudar também o conceito de um **reconhecedor** para a linguagem, que é um algoritmo que:
    1. responde "SIM" para toda cadeia pertencente à linguagem,
    2. para cadeias não-precentes à linguagem, ou responde "NÃO", ou apenas "se cala" e não responde nada.

Linguagens Formais e Problemas de Decisão

- Decidir linguagens é equivalente a resolver problemas de decisão.

  Ou seja, existe uma correspondência entre:

  - um decisor para uma linguagem e
  - uma solução para um problema de decisão com um número enumerável de instâncias.

- Para ver a equivalência, primeiro introduzimos a notação $\langle x \rangle$ para a **representação** de um objeto x como uma cadeia de um alfabeto Σ.

- Assim, um problema de decisão com um número enumerável de instâncias

  "A resposta para o PD com parâmetros x é 'SIM' ou 'NÃO'?"

  é equivalente ao problema de linguagem

  "A cadeia $\langle x \rangle$ pertence à linguagem L?"

  para uma linguagem L apropriada.

- $\boxed{\text{Exemplo 18}}$ O problema de decisão

  "n é primo, para $n \in \mathbb{N}$?"

  é equivalente ao problema de decidir a pertinência de uma cadeia $\langle n \rangle$ à linguagem

  $$L_{\text{primos}} = \{\langle n \rangle \mid n \in \mathbb{N} \text{ e } n \text{ é primo}\}.$$

  Uma maneira de fazer isto é definir a seguinte representação para os naturais

  $$\langle n \rangle = 0^n,$$

  de forma que $\langle 1 \rangle = 0$, $\langle 5 \rangle = 00000$, $\langle 0 \rangle = 0^0 = \epsilon$, etc.

  Assim, o problema de decisão *"n é primo?"* equivalente ao problema de decidir a pertinência de uma cadeia 0^n à linguagem

  $$L_{\text{primos}} = \{0^n \mid n \text{ é primo}\}.$$

- $\boxed{\text{Exemplo 19}}$ O problema de decisão

$$\text{“}x + y = z, \textit{ para } x, y, z \in \mathbb{Z}\text{?”}$$

é equivalente ao problema de decidir a pertinência de uma cadeia à linguagem

$$L_{x+y=z} = \{\langle x, y, z\rangle \mid x, y, z \in \mathbb{Z} \text{ e } x + y = z\}.$$

$\triangleleft$

- $\boxed{\text{Exemplo 20}}$ O problema de decisão

“O grafo G possui um ciclo de tamanho pelo menos k?”

é equivalente ao problema de decidir a pertinência de uma cadeia à linguagem

$$L_{grafo} = \{\langle G, k\rangle \mid G \text{ é um grafo com um ciclo de tamanho pelo menos } k\}.$$

$\triangleleft$

Linguagens Formais e Problemas de Decisão

- Como dissemos, reconhecer linguagens é equivalente a resolver problemas de decisão com um número enumerável de instâncias.
- Neste curso vamos construir <u>máquinas</u> que decidem se cadeias pertencem ou não a uma dada linguagem.

  Cada máquina uma destas máquinas é uma solução para o problema de decisão equivalente à linguagem!
- Então, é importante entender quais linguagens podem ser decididas (ou reconhecidas) por quais máquinas, pois assim estamos estudando quais problemas de decisão têm solução (ou têm solução parcial)!

A Hierarquia de Chomsky

- As linguagens podem ser organizadas em níveis de complexidade.
- A **hierarquia de Chomsky** organiza as linguagens em termos da complexidade as máquinas necessárias para dedicidí-las/reconhecê-las.
- Neste curso vamos começar com as máquinas mais simples, e ir subindo na Hierarquia de Chomsky.