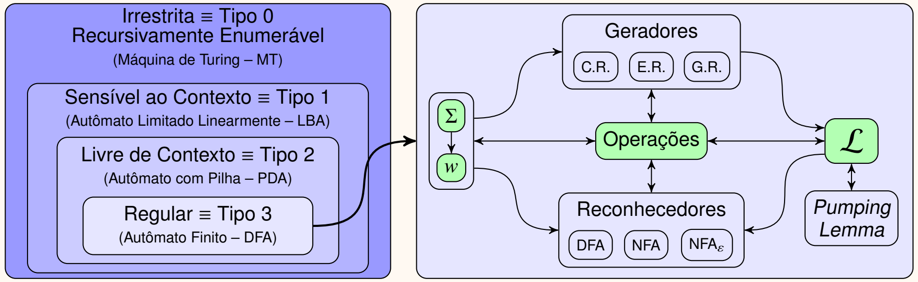
Linguagens Formais e Autômatos

Humberto Longo

Instituto de Informática
Universidade Federal de Goiás

Bacharelado em Ciência da Computação, 2022/2

Roteiro



Definições

Alfabeto: conjunto finito de *símbolos* ou *caracteres*.
- $\{0,1\}$, $\{a,b\}$, $\{a,b,c,\ldots,z\}$.

Cadeia de símbolos: sequência de zero ou mais símbolos (de um alfabeto) justapostos.
- ε, 0, 1, 010, $1010101010101010\ldots$, $abbbbb$, $abc\ldots$

Palavra: cadeia finita de símbolos.
- ε, 0, 1, 010, 1010101010101010, $abbbbb$, $abc\ldots xyz$.

Notação

- Σ : conjunto de símbolos (um alfabeto).
- ε : cadeia ou palavra vazia.
- Σ^* : conjunto de todas as cadeias possíveis sobre Σ.
- Σ^+ : $\Sigma^* - \{\varepsilon\}$.
- $|w|$: comprimento ou tamanho da cadeia w (número de símbolos que compõem w)

Palavra, prefixo, sufixo, tamanho

Prefixo : subsequência inicial de símbolos de uma palavra.

Sufixo : subsequência final de símbolos de uma palavra.

Subpalavra : sequência de símbolos contíguos de uma palavra.

Exemplo 1.1

- Se $\Sigma = \{a,b\}$, então:
  - $\Sigma^+ = \{a, b, aa, ab, ba, bb, aaa, \ldots\}$.
  - $\Sigma^* = \{\varepsilon, a, b, aa, ab, ba, bb, aaa, \ldots\}$.
- Se $abcb$ é uma palavra sobre o alfabeto $\Sigma = \{a,b,c\}$, então:
  - $|abcb| = 4$ e $|\varepsilon| = 0$.
  - $\varepsilon, a, ab, abc, abcb$ são os prefixos;
  - $\varepsilon, b, cb, bcb, abcb$ são os sufixos;
  - ε, ab, bc, bcb são exemplos de subpalavras.

Concatenação de palavras

- Operação binária, definida sobre uma linguagem $\mathcal{L}$, que associa a cada par de palavras uma palavra formada pela justaposição da primeira com a segunda.
  - Se $v, w \in \mathcal{L}$, então vw é a concatenação de v e w.
- Não é necessariamente fechada em $\mathcal{L}$:
  - a concatenação de duas palavras de uma linguagem não necessariamente resulta em uma palavra da linguagem.
- É associativa:
  - Se $t, v, w \in \mathcal{L}$, então $v(wt) = (vw)t = vwt$
- A palavra vazia é o elemento neutro à esquerda e à direita:
  - $\varepsilon w = w = w\varepsilon$.

Comprimento de uma cadeia

$|w|$: Definição recursiva.

$$|w| = \begin{cases} 0 & \text{se } w = \varepsilon; \\ |v| + 1 & \text{se } w = va, \text{ tal que } v \in \Sigma^* \text{ e } a \in \Sigma. \end{cases}$$

$|w|_a$: Nr. de ocorrências do símbolo a na palavra w.

- Ex.: $|\varepsilon|_a = 0$, $|abb|_a = 1$, $|aba|_a = 2, \ldots$
- Definição recursiva:

$$|w|_a = \begin{cases} 0 & \text{se } w = \varepsilon; \\ |v|_a & \text{se } w = vb \text{ e } a \neq b \quad (v \in \Sigma^* \text{ e } a, b \in \Sigma); \\ |v|_a + 1 & \text{se } w = va \quad (v \in \Sigma^* \text{ e } a \in \Sigma). \end{cases}$$

Concatenação de palavras

Definição 1.2

- Sejam $u, v \in \Sigma^*$. A concatenação de u e v (uv) é uma operação binária em Σ^* definida como:

Base: Se $|v| = 0$, então $v = \varepsilon$ e $uv = u$.

Recursão: Se v é uma cadeia com $|v| = n > 0$, então $uv = (uw)a$ e $v = wa$, para alguma cadeia w tal que $|w| = n - 1$ e $a \in \Sigma$.

Definição 1.3

- w^n : n concatenações sucessivas da palavra $w \in \mathcal{L}$:

Base: $w^0 = \varepsilon$.

Recursão: $w^n = w^{n-1}w$, para $n > 0$.

Concatenação de palavras

Teorema 1.4

- *Se $u, v, w \in \Sigma^*$, então $(uv)w = u(vw)$.*

Demonstração (Indução no comprimento de w).

Base: Se $|w| = 0$, então $w = \varepsilon$ e $(uv)w = uv$ (pela definição de concatenação). Por outro lado, $u(vw) = u(v) = uv$.

Hipótese: Suponha que $(uv)w = u(vw)$ para toda cadeia w com $|w| = n$.

□

Concatenação de palavras

Teorema 1.4

- *Se $u, v, w \in \Sigma^*$, então $(uv)w = u(vw)$.*

Demonstração (Indução no comprimento de w).

Passo indutivo: Seja uma cadeia w com $|w| = n + 1$. Então $w = xa$ para alguma cadeia x de comprimento n e $a \in \Sigma$.

$\square$

Concatenação de palavras

Teorema 1.4

- *Se $u, v, w \in \Sigma^*$, então $(uv)w = u(vw)$.*

Demonstração (Indução no comprimento de w).

Passo indutivo: Seja uma cadeia w com $|w| = n + 1$. Então $w = xa$ para alguma cadeia x de comprimento n e $a \in \Sigma$.

$$
\begin{aligned}
(uv)w &= (uv)(xa) && \text{(substituição, } w = xa\text{)} \\
&= (uv)xa && \text{(definição de concatenação)} \\
&= ((uv)x)a && \text{(definição de concatenação)} \\
&= (u(vx))a && \text{(hipótese indutiva, } |x| = n\text{)} \\
&= u((vx)a) && \text{(definição de concatenação)} \\
&= u(v(xa)) && \text{(definição de concatenação)} \\
&= u(v(w)) && \text{(substituição, } xa = w\text{)} \\
&= u(vw).
\end{aligned}
$$

$\square$

Reverso de uma cadeia

Definição 1.5

- Seja a cadeia $u \in \Sigma^*$. A cadeia reversa u^R de u é definida como:

Base: Se $|u| = 0$, então $u = \varepsilon$ e $\varepsilon^R = \varepsilon$.

Recursão: Se $|u| = n > 0$ e $u = wa$, com $w \in \Sigma^*$, $|w| = n - 1$ e $a \in \Sigma$, então $u^R = aw^R$.

Exemplo 1.6

- $(abc)^R = c(ab)^R = c(b(a)^R) = c(b(a(\varepsilon)^R)) = c(b(a(\varepsilon))) \equiv cba$.

Reverso de uma cadeia

Teorema 1.7

- *Seja $u, v \in \Sigma^*$. Então $(uv)^R = v^R u^R$.*

Demonstração (Indução no comprimento de v).

Base: Se $|v| = 0$, então $v = \varepsilon$ e $(uv)^R = u^R$. De forma semelhante, $v^R u^R = \varepsilon^R u^R = u^R$.

Hipótese: Suponha que $(uv)^R = v^R u^R$ para toda cadeia v com $|v| = n$.

$\square$

Reverso de uma cadeia

Teorema 1.7

- *Seja* $u, v \in \Sigma^*$. *Então* $(uv)^R = v^R u^R$.

Demonstração (Indução no comprimento de v).

Passo indutivo: Se v é uma cadeia com $|v| = n + 1$, então $v = wa$ para alguma cadeia w de comprimento n e $a \in \Sigma$.

$$\begin{aligned}
(uv)^R &= (u(wa))^R \\
&= ((uw)a)^R && \text{(associatividade da concatenação)} \\
&= a(uw)^R && \text{(definição de reverso)} \\
&= a(w^R u^R) && \text{(hipótese indutiva: } |w| = n) \\
&= (aw^R)u^R && \text{(associatividade da concatenação)} \\
&= (wa)^R u^R && \text{(definição de reverso)} \\
&= v^R u^R. && \square
\end{aligned}$$

Linguagens formais

- Dado um alfabeto Σ, uma linguagem em Σ é um conjunto de sequências de símbolos (palavras) do alfabeto.
- Se $\Sigma = \{a, b\}$, então são linguagens sobre Σ:
  - Finitas: o conjunto vazio e o conjunto formado pela palavra vazia. (Atenção: $\{\ \} \neq \{\varepsilon\} \neq \varepsilon$).
  - Finitas: $\{a, b, aa, ab, ba, bb\}, \{\varepsilon, aaa, bbb\}, \{aaa, aab, aba, abb\}$.
  - Infinitas: o conjunto $\{\varepsilon, a, b, aa, bb, aaa, aba, bab, bbb, aaaa, \dots\}$ de palíndromos sobre Σ.
- Linguagem Σ^* : conjunto de todas as sequências de símbolos do alfabeto Σ.
  - $\varepsilon \in \Sigma^*$.
  - $\mathcal{L} \subseteq \Sigma^*$, se $\mathcal{L}$ é uma linguagem em Σ.

Linguagens formais

- Especificação de uma linguagem:
  - Descrição não ambígua das cadeias da linguagem.
- Linguagem finita:
  - Enumeração de suas cadeias.
- Linguagem infinita:
  - Definição recursiva das cadeias (para linguagens com estrutura sintática simples).
  - Construção a partir de conjuntos finitos através dos operadores de conjuntos.
  - Definição por expressões regulares.
  - Definição por gramática regular.

Conjuntos definidos por indução

- Uma definição indutiva/recursiva de um conjunto C tem a seguinte forma:

  Base: Especificação de um ou mais elementos “iniciais” de C (todos os elementos de menor tamanho possível).

  Recursão: Uma ou mais regras para construção de “novos” elementos de C a partir de elementos “antigos” de C.

  Fecho: O conjunto C consiste exatamente dos elementos que podem ser obtidos, começando-se com os elementos iniciais de C, aplicando-se as regras de recursão para a construção de novos elementos.
- **Obs.:** A condição de fechamento é frequentemente omitida, uma vez que é sempre assumida nas definições indutivas.

Definição recursiva de linguagens

Exemplo 1.8

- Seja Σ um alfabeto. A definição recursiva do conjunto Σ^*, das cadeias definidas sobre Σ, é:

  Base: $\varepsilon \in \Sigma^*$.
  Recursão: Se $w \in \Sigma^*$ e $a \in \Sigma$, então $wa \in \Sigma^*$.
  Fecho: $w \in \Sigma^*$ se w pode ser obtida a partir de ε com um número finito de aplicações do passo recursivo.

Definição recursiva de linguagens

Exemplo 1.9

- Linguagem $\mathcal{L}$, sobre o alfabeto $\Sigma = \{a, b\}$, que contém cadeias de comprimento par e começam com a:

  Base: $aa, ab \in \mathcal{L}$.
  Recursão: Se $u \in \mathcal{L}$, então $uaa, uab, uba, ubb \in \mathcal{L}$.
  Fecho: Uma cadeia $u \in \mathcal{L}$ se pode ser obtida a partir das cadeias básicas, com a aplicação um número finito de vezes da recursão.

Definição recursiva de linguagens

Exemplo 1.10

- Linguagem $\mathcal{L}$, sobre o alfabeto $\Sigma = \{a, b\}$, que contém cadeias de comprimento múltiplo de 3:

  Base: $\varepsilon \in \mathcal{L}$.
  Recursão: Se $u \in \mathcal{L}$, então $aaau, aabu, abau, abbu, baau, babu, bbau, bbbu \in \mathcal{L}$.
  Fecho: Uma cadeia $u \in \mathcal{L}$ se pode ser obtida a partir das cadeias básicas, com a aplicação um número finito de vezes da recursão.

Definição recursiva de linguagens

Exemplo 1.11

- Linguagem $\mathcal{L}$, sobre o alfabeto $\Sigma = \{a, b\}$, cujas cadeias contém quantidade ímpar de ocorrências de ab:

  Base: $ab \in \mathcal{L}$.
  Recursão: Seja $u \in \mathcal{L}$. Se $u = av$, $v \in \Sigma^+$, $au, bu \in \mathcal{L}$; se $u = bv$, $v \in \Sigma^+$, $bu, auab, abau \in \mathcal{L}$; se $u = va$, $v \in \Sigma^+$, $ua, abub, ubab \in \mathcal{L}$; se $u = vb$, $v \in \Sigma^+$, $ua, ub \in \mathcal{L}$.
  Fecho: Uma cadeia $u \in \mathcal{L}$ se pode ser obtida a partir das cadeias básicas, com a aplicação um número finito de vezes da recursão.

Definição recursiva de linguagens

Exemplo 1.12

- Linguagem $\mathcal{L}$, sobre o alfabeto $\Sigma = \{a, b\}$, que contém cadeias em que cada ocorrência de um b é precedida de um a:
  - Base: $\varepsilon \in \mathcal{L}$.
  - Recursão: Se $u \in \mathcal{L}$, então $ua, uab \in \mathcal{L}$.
  - Fecho: Uma cadeia $u \in \mathcal{L}$ se pode ser obtida a partir das cadeias básicas, com a aplicação um número finito de vezes da recursão.
- Linguagem $\mathcal{L}$, sobre o alfabeto $\Sigma = \{a, b\}$, cujas cadeias não contém b's consecutivos:
  - Base: $\varepsilon, b \in \mathcal{L}$.
  - Recursão: Se $u \in \mathcal{L}$, então $ua, uab \in \mathcal{L}$.
  - Fecho: Uma cadeia $u \in \mathcal{L}$ se pode ser obtida a partir das cadeias básicas, com a aplicação um número finito de vezes da recursão.

Livros texto

R. P. Grimaldi
Discrete and Combinatorial Mathematics – An Applied Introduction.
Addison Wesley, 1994.

D. J. Velleman
How To Prove It – A Structured Approach.
Cambridge University Press, 1996.

J. E. Hopcroft; J. Ullman.
Introdução Ã Teoria de Autômatos, Linguagens e Computação.
Ed. Campus.

T. A. Sudkamp.
Languages and Machines – An Introduction to the Theory of Computer Science.
Addison Wesley Longman, Inc. 1998.

J. Carroll; D. Long.
Theory of Finite Automata – With an Introduction to Formal Languages.
Prentice-Hall, 1989.

M. Sipser.
Introduction to the Theory of Computation.
PWS Publishing Company, 1997.

H. R. Lewis; C. H. Papadimitriou
Elementos de Teoria da Computação.
Bookman, 2000.