# 1. Linguagens

Copiar link

## Símbolos

**Exemplos:**
a, b, 0, 1

As unidades mais básicas de comunicação e construção de linguagens.

As “letras” (caracteres) ou algarismos (dígitos) com que construímos palavras e números são exemplo de símbolos.

## Alfabetos

Um alfabeto é um **conjunto finito** e **não-vazio** de símbolos. É comum utilizar-se a letra Grega $\Sigma$ (“Sigma”) para designar alfabetos.

**Exemplos:**
- $\Sigma = \{0, 1\}$
- $\Sigma = \{a, b, c\}$
- $\Sigma = \{a\}$
- $\Sigma = \{a, \ldots, z, A, \ldots, Z\}$

## Palavras

Uma palavra (“string”) é uma **sequência** de **comprimento finito**, **possivelmente vazia**, de símbolos de um determinado alfabeto.

A palavra vazia denota-se por $\varepsilon$ (“epsilon”)

**Exemplos**
- Qualquer palavra portuguesa, construída sobre o alfabeto $\Sigma = \{a, \ldots, z, A, \ldots, Z\}$
- Qualquer número natural, construído sobre o alfabeto $\Sigma = \{0, 1, \ldots, 9\}$

**Operações sobre palavras**

Veremos a bem conhecida operação de **concatenação** (ou justaposição) de strings em termos abstratos (algébricos) como uma operação de *multiplicação*.

Por exemplo, se $s_1 = abc$ e $s_2 = d0e$, então escreveremos $s_1 s_2 = abcd0e$

A operação tem como *elemento neutro* (ou "1") a string vazia, uma vez que para qualquer palavra $s$ temos $\varepsilon s = s\varepsilon = s$

A **exponenciação** de palavras tem a definição matemática usual, tomando a operação de concatenação como multiplicação.

**Exemplos**

- $(ab)^2 = abab$
- $(ab)^3 = ababab$
- $a^3 = aaa$
- $(ab)^2 ab^4 = ababab bbb$

Note que os símbolos "(" e ")" não pertencem à linguagem! São símbolos auxiliares.

A exponenciação com expoente 0, tal como em aritmética, tem como resultado o elemento neutro da multiplicação:

> $s^0 = \varepsilon$

**Definição Recursiva**

A seguinte definição de exponenciação não surpreenderá ninguém com experiência em Programação Funcional:

> $s^0 = \varepsilon$
> $s^n = ss^{n-1}$

Por exemplo, $s^4 = ss^3 = sss^2 = ssss^1 = sssss\varepsilon = ssss$

# Linguagens

Uma linguagem é um **conjunto**, **possivelmente vazio e possivelmente infinito**, de palavras.

Note-se que as palavras são finitas! Mas uma linguagem pode conter um número infinito de palavras.

**Exemplos**

- $L = \{\varepsilon, ab, bb, abc\}$
- $L_{01Rep} = \{(01)^i \mid i \geq 0\}$

Esta segunda linguagem é definida, de forma muito conveniente, recorrendo a notação de *construção de conjuntos por compreensão*. Está implícito que $i$ é um número inteiro. Temos assim $L_{01Rep} = \{\varepsilon, 01, 0101, 010101, \ldots\}$ (linguagem infinita).

Um outro exemplo:

- $L_{a\leq b} = \{a^i b^j \mid i, j \geq 0 \wedge i < j\}$

**Exercício**

> *Escreva alguns exemplos de palavras desta linguagem*

**Operações sobre Linguagens**

A **concatenação** de duas linguagens define-se da seguinte forma:

$$L_1 L_2 = \{xy \mid x \in L_1 \wedge y \in L_2\}$$

A operação de concatenação de linguagens tem como *elemento neutro* (ou "1", se virmos a concatenação como multiplicação de linguagens) a linguagem $\{\varepsilon\}$, uma vez que para qualquer palavra $s$ temos $\varepsilon s = s\varepsilon = s$, logo $\{\varepsilon\}L = L\{\varepsilon\} = L$ para qualquer linguagem $L$.

**Exercício**

> *Seja $\emptyset$ a linguagem vazia. Como se caracteriza a linguagem $\emptyset L$ para uma qualquer linguagem $L$?*

A **exponenciação** de linguagens tem mais uma vez a definição matemática esperada, tomando a operação de concatenação como multiplicação:

> $L^0 = \{\varepsilon\}$
> $L^n = LL^{n-1}$

**Exemplo**

Seja $M = \{a, bc\}$. Então $M^2 = \{aa, abc, bca, bcbc\}$.

**Exercício**

> *Calcule $M^0$, $M^1$, e $M^3$*

A **união** e **intersecção** de linguagens são as habituais operações sobre conjuntos:

$$L_1 \cup L_2 = \{x \mid x \in L_1 \vee x \in L_2\}$$

$$L_1 \cap L_2 = \{x \mid x \in L_1 \wedge x \in L_2\}$$

**Exercício**

> *Sejam $L_e = \{(aa)^i \mid i \geq 0\}$ e $L_o = \{a^{2i+1} \mid i \geq 0\}$. Apresente uma definição por compreensão para a linguagem $L_e \cup L_o$.*

## Operação Estrela (*Kleene Star*) e Linguagem Universal

$L^* = L^0 \cup L^1 \cup L^2 \cup L^3 \cup \dots$

Ou equivalentemente: $L^* = \bigcup_{i=o}^{\infty} L^i$

Ou ainda, $L^* = \{x \mid x \in L^k \text{ para algum } k \geq 0\}$

Esta linguagem contém pois todas as palavras obtidas por concatenação de palavras de $L$, captando a noção de *repetição*.

Tendo em conta que um alfabeto pode ser visto como uma linguagem (contendo apenas palavras de comprimento 1), este operador pode também ser aplicado a alfabetos.

Dado um alfabeto $\Sigma$, a linguagem $\Sigma^*$ é conhecida como a *linguagem universal* sobre $\Sigma$.

**Exemplo**
Dado o alfabeto $\Sigma = \{0, 1\}$, temos
$\Sigma^* = \{\varepsilon, 0, 1, 00, 01, 000, 001, 010, 011, 100, 101, 110, 1000, \dots\}$

Trata-se do conjunto, *infinito*, de todas as palavras de *comprimento finito* formadas a partir de $\Sigma$.

**Exercícios**

1. Qual será a linguagem $\emptyset^*$ ?
2. Recorde as linguagens $L_e = \{(aa)^i \mid i \geq 0\}$ e $L_o = \{a^{2i+1} \mid i \geq 0\}$.
   Quais das seguintes igualdades serão verdadeiras?
   a. $L_e^* = L_e$
   b. $L_o^* = L_o$
   c. $L_e^* = \{a\}^*$
   d. $L_o^* = \{a\}^*$
   e. $L_e \cup L_0 = \{a\}^*$

**Noções Relacionadas**

É possível definir recursivamente a construção por repetição $L_n^*$ até um determinado comprimento máximo $n$:

$$L_0^* = \{\varepsilon\}$$
$$L_n^* = L^n \cup L_{n-1}^*$$

A noção de **subtração** de linguagens é definida como a usual subtração de conjuntos:

$$L_1 - L_2 = \{x \mid x \in L_1 \wedge x \notin L_2\}$$

Com esta operação podemos definir a noção de **complemento** de uma linguagem, como sendo a diferença para a linguagem universal sobre o alfabeto:

$$\overline{L} = \Sigma^* - L$$

(Poderíamos alternativamente definir primeiro por compreensão a noção de complemento, e depois a noção de subtração usando a operação de complemento)

# Linguagens Regulares

Dado um alfabeto $\Sigma$, as linguagens regulares de $\Sigma^*$ definem-se recursivamente como se segue:

- $\emptyset$ é uma LR
- $\{\varepsilon\}$ é uma LR
- Se $x \in \Sigma$ então $\{x\}$ é uma LR
- Se $A$ e $B$ são LRs então $AB$, $A \cup B$, $A^*$são LRs

Um exemplo de uma linguagem que não é regular é $L = \{a^n b^n \mid n \geq 0\}$.

## Ordem e Enumeração de Palavras

Como listar de forma exaustiva as palavras de uma linguagem?

Vamos assumir que existe uma ordem associada ao alfabeto subjacente, por exemplo para $\Sigma = \{0, 1\}$ podemos asumir $0 < 1$, e podemos para $\Sigma = \{a, b\}$ assumir $a < b$.

Esta ordem pode ser estendida às palavras da linguagem universal $\Sigma^*$. A ordem **lexicográfica** é a habitualmente conhecida por ordem "alfabética". Por exemplo, aaa < aab e aa < aaa.

Podemos listar as palavras de uma linguagem usando esta ordem lexicográfica, no entanto isto apresenta problemas, porque sendo a linguagem infinita, algumas strings finitas podem nunca ter oportunidade de ser listadas.

Por exemplo para a linguagem $\{a, b\}^*$ teríamos $\varepsilon, a, aa, aaa, aaaa, \ldots$
As palavras contendo $b$ nunca seriam listadas.

É mais útil então produzir uma **enumeração** das palavras, por ordem crescente de comprimento, podendo para cada grupo de comprimento utilizar a ordem lexicográfica.

Para a mesma linguagem $\{a, b\}^*$ teríamos agora
$\varepsilon, a, b, aa, ab, ba, bb, aaa, aab, \ldots, bbb, aaaa, \ldots$