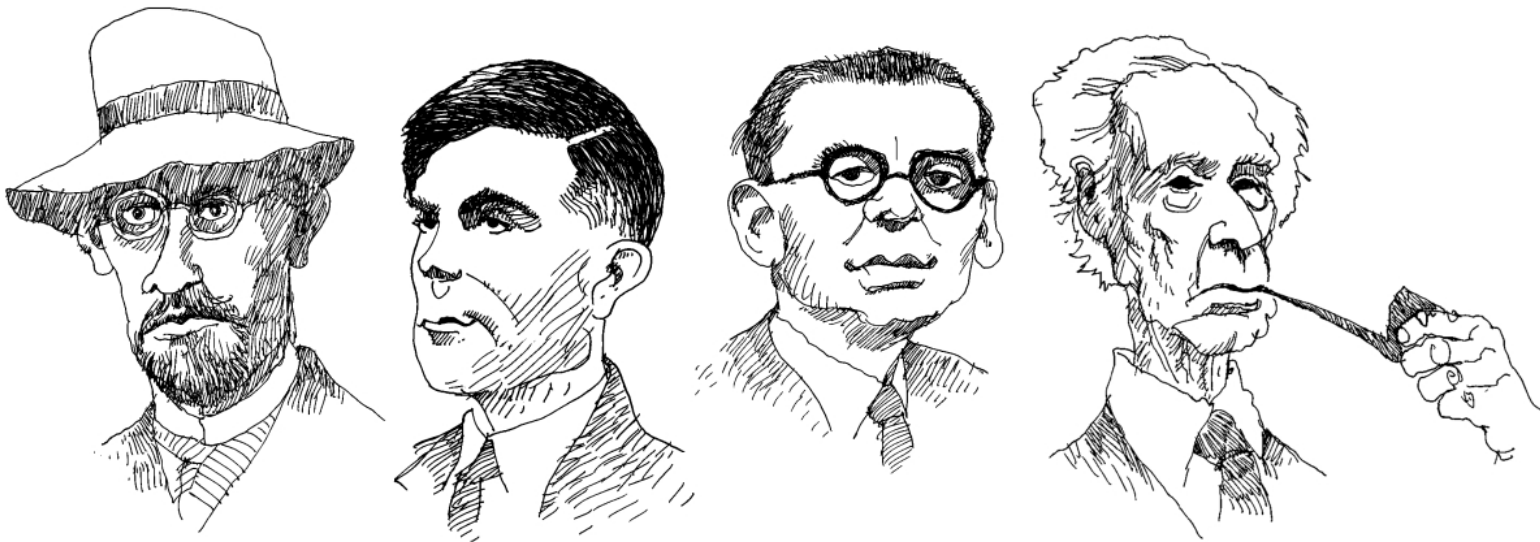


Valdigleis S. Costa

Computação Formal

Um Compêndio de Notas sobre os Fundamentos Matemáticos da Computação



Valdigleis S. Costa

Professor Adjunto, Colegiado de Ciência da Computação

Universidade Federal do Vale do São Francisco

Salgueiro, PE

Copyright © 2019-2022 Valdigleis S Costa

Este texto NÃO possui qualquer tipo de vínculo editorial, e não possui fins lucrativos.

Página pessoal do autor <https://profvaldi.site>

As imagens usadas na capa pode ser encontradas no ótimo *paper* de Chaitin [23].



Este material é licenciado sob a Licença Atribuição-NãoComercial-CompartilhaIgual 3.0 Não Adaptada (CC BY-NC-SA 3.0). Você pode obter uma copia da licença acessando a página:

https://creativecommons.org/licenses/by-nc-sa/3.0/deed.pt_BR

ou enviando uma carta para Creative Commons, 444 Castro Street, Suite 900, Mountain View, California, 94041, USA.

Este manuscrito é redigido usando um *template* desenvolvido pelo próprio autor. Este texto foi escrito com \LaTeX e \LaTeX 2\epsilon nos ambientes de trabalho Debian/Ubuntu e Mac OS usando o *software* TeXstudio e as distribuições TeXLive e MacTeX.

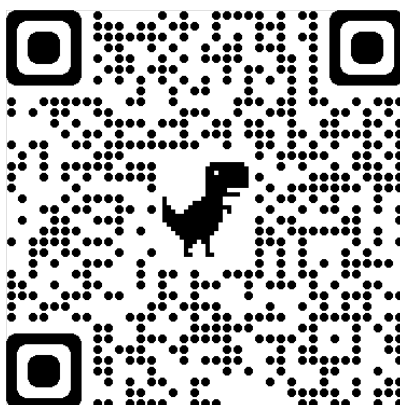
Release, 11 de outubro de 2022–20:45:02

Disclaimer

Este manuscrito está sendo construído tendo como base diversas notas de aula que eu preparei para cursos de:

- | | |
|--|---|
| (i) Matemática Discreta | (iv) Análise de Algoritmos |
| (ii) Lógica para Ciência da Computação | (v) Computabilidade |
| (iii) Linguagens Formais e Autômatos | (vi) Teoria dos Números para Computação |

Uma vez que este manuscrito ainda é um projeto em andamento e possivelmente sua escrita nunca será realmente concluída com total aprovação de seu autor, é claro que você poderá encontrar diversos erros, que com toda certeza você leitor irá me enviar e-mails¹ ou *issues*² com reports de tais erros. Para acessar a página da nova versão deste manuscrito basta escanear o **QR code** abaixo.



¹E-mail do autor: valdigleis.costa@univasf.edu.br

²Páginas de *issues*: <https://github.com/valdigleis/Manuscrito/issues>

Este trabalho é dedicado aos meus professores e alunos, vocês são um conjunto enumerável de bons motivos para continuar escrevendo!

Agradecimentos

Este manuscrito está sendo construído aos poucos e diversas pessoas com o tempo irão contribuir direta ou indiretamente para que ele fique cada vez melhor, assim dedico esta(s) página(s) a agradecer nominalmente a qualquer um que tenha ajudado:

- **Raul da Silva Martins** ([Discente univasf/2021](#)) - Apontou correções de conteúdo no texto.
- **Matheus Barros Rosa** ([Discente univasf/2022](#)) - Apontou correções ortográficas no texto.
- **Gabriel Ferreira Rodrigues** ([Discente univasf/2022](#)) - Apresentou problemas de nomenclatura em uma definição.

“Ciência da Computação está tão
relacionada aos computadores quanto a
Astronomia aos telescópios, Biologia aos
microscópios, ou Química aos tubos de
ensaio...”

Edsger Wybe Dijkstra (1930-2002)

Conteúdo

I Ferramentas e Linguagem Básica

1	Conjuntos	13
1.1	Conjuntos e Elementos	13
1.2	Operações sobre conjuntos	19
1.3	Operações generalizadas	30
1.4	Partes e Partições	32
1.5	Questionário	33
2	Métodos de Demonstração	43
2.1	Introdução	43
2.2	Demonstrando Implicações	46
2.3	Demonstração por Absurdo	51
2.4	Demonstrando Generalizações	53
2.5	Demonstrando Existência e Unicidade	55
2.6	Demonstração Guiada por Casos	59
2.7	Outras Formas de Representação de Provas	61
2.8	Demonstração de Suficiência e Necessidade	61
2.9	Refutação por Contraexemplo	61
2.10	Questionário	62
3	Relações	65
3.1	Noções Básicas de Relações	65
3.2	Pares Ordenados e Produto Cartesiano	66

3.3	Relações	72
3.4	Tipos ou Propriedades das Relações Binárias	77
3.5	Fecho das Relações Binárias	85
3.6	Relações e Grafos	88
3.7	Questionário	88
4	Equivalência e Ordem	93
4.1	Introdução	93
4.2	Relações de Equivalência e Espaço Quociente	94
4.3	Relações de Ordem	96
4.4	<i>Posefs</i> e Diagramas de Hasse	99
4.5	Elementos Notáveis de um <i>Posef</i>	103
4.6	Princípio da Boa Ordenação	110
4.7	Somas Ordinais, Ordem Produto e Ordem Lexicográfica	110
4.8	Questionário	110
5	Funções	117
5.1	Conceitos, Definições e Nomenclaturas	117

II Álgebra Universal

III Lógica

6	Introdução à Lógica	127
6.1	O que é Lógica?	127
6.2	Um Pouco de História	129
6.3	Argumentos, Proposições e Predicados	131
6.4	Conectivos, Quantificadores e Negação	133
6.5	Representação simbólica	136
6.6	Lógica e Ciência da Computação	138
6.7	Questionário	139

7	Lógica Proposicional	143
7.1	Introdução	143
7.2	A linguagem Proposicional	143
7.3	Sobre o Sistema de Dedução Natural	145
7.4	Regras de Dedução Natural	147
7.5	Construção de Demonstrações em Dedução Natural	153
7.6	Propriedades do Sistema de Dedução Natural	153
7.7	Sistemas Axiomáticos ao Estilo de Hilbert	153
7.8	A Semântica da Linguagem L_{Prop}	153
7.9	Propriedades do Sistema Semântico	153
7.10	Corretude, Consistência e Completude	153
8	Lógica de Primeira Ordem	155

IV Linguagens Formais e Autômatos

9	Introdução	159
9.1	Sobre as Linguagens Formais	159
9.2	Noções Fundamentais	159
9.3	Sobre Autômatos Finitos	167
9.4	Sobre Gramática Formais	168
9.5	Questionário	169
10	Autômatos Finitos e Linguagens Regulares	171
10.1	Autômato Finito Determinístico	171
10.2	Autômato Finito Não-determinístico	178
10.3	Autômatos Finitos Não-determinísticos com Movimentos Vazios	188
10.4	Teorema Myhill-Nerode e a Minimização de AFD	195
10.5	Máquinas de Mealy	204
10.6	A Notação Matricial	210
10.7	Questionário	210

11	Expressões e Gramáticas Regulares	217
11.1	Expressões Regulares	217
11.2	Gramática Regulares	227
11.3	Questionário	233
12	Álgebra das Linguagens Regulares	239
12.1	Operadores de Fecho	239
13	Linguagens Livres do Contexto	243
14	Computabilidade à Turing	245

Parte I

Ferramentas e Linguagem Básica

Capítulo 1

Conjuntos

“-Comece pelo começo”, disse o Rei de maneira severa,
“-E continue até chegar ao fim, então pare!”

Lewis Carroll, Alice no País das Maravilhas.

1.1 Conjuntos e Elementos

A ideia de conjunto é provavelmente o conceito mais fundamental compartilhado pelos mais diversos ramos da matemática. O primeiro grande estudioso que apresentou uma forma relativamente precisa para o conceito de conjunto foi o matemático alemão George Cantor (1845-1918) em seu seminal trabalho [20]. A seguir será apresentada uma tradução não literal da definição original de Cantor.

Definição 1.1 — Cantor. Um **conjunto** A é uma **coleção** numa totalidade M de certos **objetos** distintos e bem-definidos n que fazem parte da nossa percepção ou pensamento, tais objetos são chamados de **elementos** de A .

Note que a definição apresentada por Cantor exige dois aspectos sobre a natureza dos elementos em um conjunto: (1) que eles sejam distintos entre si, ou seja, em um conjunto não poderá haver repetição de elementos e (2) os elementos devem ser bem-definidos.

A definição de Cantor permite que sejam criados conjuntos com qualquer coisa que o indivíduo racional possa pensar ou perceber pelos seus sentidos. Agora, entretanto, deve-se questionar o que significa dizer que algo é bem-definido? Uma resposta satisfatória para essa pergunta é dizer que algo é bem-definido se esse algo pode ser descrito sem ambiguidades.

É claro que qualquer coisa pode ser descrita a partir de suas propriedades, características ou atributos,

sendo que essas propriedades sempre podem ser verificadas pelos sentidos no caso de objetos físicos, e sempre pode-se pensar e argumentar sobre elas no caso de objetos abstratos. Assim pode-se modificar um pouco a definição de Cantor para a definição que se segue.

Definição 1.2 — Definição de Cantor Modificada. Um **conjunto** A é uma **coleção** numa totalidade M de certos **objetos** n distintos e que satisfazem certas propriedades, tais objetos são chamados de **elementos** de A .

Observação 1.1 A partir desse ponto será usado a nomenclatura discurso em vez de totalidade na especificação de conjuntos.

Note que a Definição 1.2 permite concluir que um conjunto pode ser visto como o agrupamento de entidades (os elementos) que satisfazem certas propriedades, ou ainda que, as propriedades definem os conjuntos. Prosseguindo nesse texto serão apresentadas as convenções da **teoria ingênua dos conjuntos** de forma usual, mas também serão apresentados os aspectos sintáticos e semânticos da teoria.

Definição 1.3 — Notações Básicas. As letras maiúsculas do alfabeto latino $A, B, \dots, M, N, \dots, Z$ como e sem indexação serão usadas como variáveis para representar conjuntos e as letras minúsculas $a, b, \dots, m, n, \dots, z$ como e sem indexação serão usadas como meta-variáveis para representar elementos.

Assim a sintaxe da teoria ingênua dos conjuntos diz que letras minúsculas sempre representam elementos e letras maiúsculas sempre representam os conjuntos.

Observação 1.2 O termo variável é usado para designar símbolos (ou palavras) de uma linguagem responsáveis por representar de forma genérica as entidades da teoria que a linguagem descreve (para detalhes leia [95]), ou seja, são “apelidos” ou “rótulos” para as entidades.

Como dito anteriormente uma propriedade \mathbf{P} é responsável por definir um conjunto, pois todos os elementos no conjunto devem satisfazer (ou possuir) tal propriedade. Tendo isso em mente pode-se introduzir a definição a seguir.

Definição 1.4 — Notação compactada. Um conjunto A definido por alguma propriedade \mathbf{P} é representada na **forma compacta** como:

$$A = \{x \mid \mathbf{P}\} \tag{1.1}$$

Na notação compacta $A = \{x \mid \mathbf{P}\}$ o símbolo A é chamado de rótulo do conjunto, e a parte $\{x \mid \mathbf{P}\}$ será chamada neste manuscrito de forma estrutural do conjunto.

Observação 1.3 A notação compacta $A = \{x \mid \mathbf{P}\}$ é na verdade uma palavra da linguagem da teoria ingênua dos conjuntos, a semântica de tal palavra pode ser interpretada como: “ A é o conjunto de todos os x 's que satisfazem (ou possuem) a propriedade \mathbf{P} ”.

■ **Exemplo 1.1** Os seguintes conjuntos estão bem representados na notação compacta.

- (a) $X = \{a \mid a \text{ é uma cidade do Brasil}\}.$
- (b) $K = \{m \mid m \text{ é um animal mamífero}\}.$
- (c) $L = \{x \mid 0 \leq x < 10 \text{ e } x \text{ é um número ímpar}\}.$
- (d) $C = \{b \mid b \text{ é uma vogal}\}.$

Para continuar o desenvolvendo da linguagem da teoria dos conjuntos, é conveniente relembrar ao leitor os símbolos usados como rótulos para representar os conjuntos numéricos mais importantes da matemática.

Definição 1.5 — Símbolos dos conjuntos numéricos. O conjunto dos números naturais^a, inteiros, racionais, irracionais, reais e complexos são representados respectivamente por \mathbb{N} , \mathbb{Z} , \mathbb{Q} , \mathbb{I} , \mathbb{R} e \mathbb{C} .

^aNeste manuscrito é considerado que o conjuntos dos naturais corresponde ao conjunto $\{0, 1, 2, \dots\}$.

Observação 1.4 Neste manuscrito \mathbb{N}_* , \mathbb{Z}_* , \mathbb{Q}_* e \mathbb{R}_* irão denotar receptivamente o conjunto dos naturais, inteiros, racionais e reais sem o 0. Já \mathbb{Z}^+ , \mathbb{Q}^+ e \mathbb{R}^+ irão denotar receptivamente o conjunto dos inteiros, racionais e reais positivos. E por fim, \mathbb{Z}^- , \mathbb{Q}^- e \mathbb{R}^- irão denotar receptivamente o conjunto dos inteiros, racionais e reais.

Seguindo com o desenvolvimento da teoria dos conjuntos a definição a seguir estabelece um relacionamento (ou relação) de pertinência entre os conjuntos e os elementos do discurso.

Definição 1.6 — Pertinência. Seja A um conjunto definido sobre um discurso M por uma propriedade \mathbf{P} e seja x um elemento do discurso. Se o elemento x possui (ou satisfaz) a propriedade \mathbf{P} , então é dito que x pertence a A , denotado por $x \in A$. Caso x não possui (ou satisfaça) a propriedade \mathbf{P} , então é dito que x não pertence a A , denotado por $x \notin A$.

A Definição 1.6 está introduzindo novas entidades da linguagem da teoria dos conjuntos, sendo tais objetos as palavras da forma $_ \in _$ e da forma $_ \notin _$. Em tais palavras o símbolo do lado esquerdo de \in e \notin sempre será visto como sendo um elemento do discurso ou uma variável que representa tal elemento, por outro lado, o símbolo do lado direito de \in e \notin sempre devem ser o rótulo ou a forma estrutural de um conjunto.

Observação 1.5 Quando $x \in A$, em alguns textos como em [63] é comum o uso das interpretações semânticas: “A possui x ” ou que “ x faz parte de A ”, durante este manuscrito possa ser que uma dessas (ou ambas) interpretações sejam usadas, além da semântica padrão: x pertence a A .

■ **Exemplo 1.2** Seja A o conjunto definido sobre a propriedade “é professor de Ciência da Computação na univasf” tem-se que o professor Rodrigo $\in A$. Já para os professores Regivan e Benjamin tem-se que Regivan, Benjamin $\notin A$.

■ **Exemplo 1.3** Seja A_1 o conjunto definido pela propriedade “Clubes da primeira divisão do campeonato brasileira de futebol do ano 2021” tem-se então que Vasco $\notin A_1$.

Há casos entretanto, que a notação compacta é descartada e assim os conjuntos podem ser escritos simplesmente listando seus elementos entre as chaves da forma estrutural, isso em geral acontece quando o conjunto é finito¹ e possui um número não muito grande de elementos.

■ **Exemplo 1.4** A seguir são listados alguns conjuntos finitos escritos descartando a notação compacta.

- (a) O conjunto das vogais pode ser representado como $A = \{a, e, i, o, u\}$.
- (b) O conjunto das siglas dos estados nordestinos pode ser escrito como $E = \{RN, PE, PB, MA, CE, SE, AL, BA, PI\}$.
- (c) O conjunto dos naturais menores que 10 é escrito como $N_{10} = \{0, 1, 2, 3, 4, 5, 6, 7, 8, 9\}$.

Observação 1.6 Quando se optar por escrever um conjunto finito apenas listando os seus elementos entre chaves a ordem com que os elementos aparecem não importa, assim tem-se que os conjuntos $\{a, e, i, o, u\}$ e $\{e, u, i, a, o\}$ são na verdade o mesmo conjunto.

Note que a relação de pertinência apresentada anteriormente (Definição 1.6) relaciona elementos e conjuntos, existe também uma relação extremamente fundamental dentro da teoria dos conjuntos que é definida entre dois conjuntos.

A relação entre conjuntos recebe o nome de **relação de inclusão**, entretanto, como dito em [63], é comum que quando um conjunto A estiver relacionado com um conjunto B pela relação de inclusão, se usar a interpretação semântica “ A é subconjunto de B ”, em vez de, “ A está incluso em B ”. A seguir é apresentado formalmente esta relação.

Definição 1.7 — Relação de inclusão. Dado dois conjuntos A e B quaisquer, é dito que A é subconjunto de B , denotado por $A \subseteq B$, quando todo $x \in A$ é tal que $x \in B$.

¹Por hora o leitor deve considerar que um conjunto finito é aquele que o leitor poderia contar o número de elementos, em capítulos futuros serão formalizados os conceitos de conjuntos finitos e infinitos.

■ **Exemplo 1.5** Dado o conjunto \mathbb{Z} tem-se que o conjunto $N = \{x \mid x \in \mathbb{Z} \text{ e } x = 2k \text{ para algum } k \in \mathbb{Z}\}$ é claramente um subconjunto de \mathbb{Z} pois todo número par é também um número inteiro.

■ **Exemplo 1.6** As seguintes relações de inclusão se verificam:

- (a) $\{a, e, u\} \subseteq \{a, e, o, i, u\}$.
- (b) $\{x \mid x \text{ é uma cidade do PE}\} \subseteq \{x \mid x \text{ é uma cidade do Brasil}\}$.
- (c) $\{x \mid x = 2k \text{ para algum } k \in \mathbb{N}\} \subseteq \mathbb{N}$.
- (d) $\{\text{Brasil}\} \subseteq \{x \mid x \text{ é um país do continente americano}\}$

É obvio que a Definição 1.7 está introduzindo novas entidades da linguagem da teoria dos conjuntos, sendo tais objetos as palavras da forma $_ \subseteq _$. Em tais palavras os elementos à esquerda e à direita do símbolo \subseteq sempre devem ser o rótulos ou as formas estruturais de conjuntos. Em oposição a relação de inclusão existe a relação de não inclusão descrita a seguir.

Definição 1.8 — Relação de não inclusão. Dado dois conjuntos A e B quaisquer, é dito que A é não subconjunto de B , denotado por $A \not\subseteq B$, quando existe pelo menos um $x \in A$ tal que $x \notin B$.

■ **Exemplo 1.7** Seja $A = \{-1, 0, 1\}$ tem-se que $A \not\subseteq \mathbb{N}$.

Existe também a possibilidade de todos os elementos de A serem elementos de B , mas que B possua outros elementos que não fazem parte de A , nesse caso é dito que A é um **subconjunto próprio** de B , e isto é denotado como $A \subset B$.

■ **Exemplo 1.8** As seguintes relações de inclusão se verificam:

- (a) $\{1, 2\} \subset \mathbb{R}$.
- (b) $\{x \mid x \text{ é uma cidade do PE}\} \subset \{x \mid x \text{ é uma cidade do Brasil}\}$.
- (c) $\mathbb{Z}_+ \subset \mathbb{Z}$.

Observação 1.7 Note que todo subconjunto A de um conjunto B pode ser visto como um conjunto construído sobre os elementos de B que satisfazem uma certa propriedade **P**, isto é, tem-se que todo subconjunto A é um conjunto da seguinte forma:

$$A = \{x \mid x \in B \text{ e } x \text{ satisfaz } \mathbf{P}\}$$

também é possível encontrar a notação $A = \{x \in B \mid x \text{ satisfaz } \mathbf{P}\}$, sempre que possível esse manuscrito irá adotar a segunda notação.

Usando a ideia de subconjunto pode-se como apresentado na literatura em obras como [2, 43, 63] introduzir a ideia de igualdade entre conjuntos, esta noção é apresentada formalmente como se segue.

Definição 1.9 — Igualdade de conjuntos. [2] Dois conjuntos A e B são iguais, denotado por $A = B$, se e somente se, $A \subseteq B$ e $B \subseteq A$.

Teorema 1.1 — Teorema da igualdade. Sejam A, B e C conjuntos quaisquer. Então:

1. $A = A$.
2. Se $A = B$, então $B = A$.
3. Se $A = B$ e $B = C$, então $A = C$.

Dentro da teoria dos conjuntos alguns conjuntos possuem tanta importância e destaque que eles recebem nomes e símbolos próprios.

Definição 1.10 — Conjunto Universo. O conjunto universo, ou universo do discurso, denotado por \mathbb{U} , é um conjunto que possui todos os elementos sobre os quais se “fala”^a.

^aO termo fala aqui diz respeito ao ato pensar ou argumentar sobre os objetos.

O universo do discurso não é único, de fato o mesmo muda em função sobre o que se está “discursando”, por exemplo, pode-se pensar em um universo do discurso para falar sobre números, carros, pessoas, animais, palavras, times de futebol e etc.

Definição 1.11 — Conjunto vazio. O conjunto vazio, denotado por \emptyset , corresponde a um conjunto que não possui nenhum elemento.

Uma propriedade interessante sobre o conjunto vazio é apresentada a seguir, tal propriedade garante que o conjunto vazio está presente em qualquer outro conjunto existente.

Teorema 1.2 Para todo conjunto A tem-se que $\emptyset \subseteq A$.

Demonstração. Suponha por absurdo que existe um conjunto A tal que $\emptyset \not\subseteq A$, assim por definição existe pelo menos um $x \in \emptyset$ tal que $x \notin A$, mas isto é um absurdo já que o vazio não possui elementos, e portanto, a afirmação que $\emptyset \not\subseteq A$ é falsa, logo, $\emptyset \subseteq A$ é verdadeiro para qualquer que seja o A . \square



Nota 1.1 Neste manuscrito ao final das demonstrações será sempre colocado o símbolo \square , tal símbolo é conhecido como túmulo de Halmos^a, este símbolo será usado para substituir a notação q.e.d. (“*quod erat demonstrandum*”) usando por outras fontes bibliográficas para marcar o ponto de finalização de uma demonstração.

^aEm inglês esse símbolo é conhecido como *tombstone*, e tal símbolo foi usado para marcar o final de uma demonstração inicialmente pelo matemático Paul Halmos (1916-2006).

Agora que foram apresentados os conjuntos universo e vazio, é conveniente comentar sobre uma situação específica da teoria dos conjuntos como apresentada até aqui. Pelo que foi apresentado até agora já se sabe que os itens em um conjunto são chamados de elementos, entretanto, não existe qualquer restrição, além de ser bem definido, para a natureza (ou tipo) dos elementos em um conjunto. Isso possibilita que seja possível definir por exemplo um conjunto de conjuntos, isto é, um conjunto em que os elementos são também conjuntos.

Definição 1.12 — Família de Conjuntos. Um conjunto A cujo os elementos são todos conjuntos, isto é, um conjunto da forma $A = \{x \mid x \text{ é um conjunto}\}$, é chamado de **família de conjuntos**.

■ **Exemplo 1.9** Os conjuntos:

$$A_1 = \{\mathbb{Z}_+^*, \mathbb{Z}_-, \{\pi, \sqrt{-1}\}\} \text{ e } A_2 = \{\{a, b\}, \{\clubsuit, \spadesuit, \heartsuit, \diamondsuit\}, \mathbb{R}\}$$

são ambas famílias.

Observação 1.8 Além do termo família algumas obras como [63] também usam a nomenclatura classe, neste manuscrito só será usado o termo classe em situações bem específicas como por exemplo, as classes de equivalência em um espaço quociente.

1.2 Operações sobre conjuntos

Seguindo a mesma organização de conteúdo apresentada em [64], pode-se agora introduzir uma série de operações conjuntistas, isto é, operações que agem diretamente sobre conjuntos de “entrada” produzindo como “saída” novos conjuntos.

Definição 1.13 — União de conjuntos. Sejam A e B dois conjuntos quaisquer, a união de A com B , denotada por $A \cup B$, corresponde ao seguinte conjunto.

$$A \cup B = \{x \mid x \in A \text{ ou } x \in B\}$$

■ **Exemplo 1.10** Dados os dois conjuntos $A = \{x \in \mathbb{N} \mid x = 2i \text{ para algum } i \in \mathbb{N}\}$ e $B = \{x \in \mathbb{N} \mid x = 2j + 1 \text{ para algum } j \in \mathbb{N}\}$ tem-se que $A \cup B = \mathbb{N}$.

■ **Exemplo 1.11** Seja $N = \{1, 2, 3, 6\}$ e $L = \{4, 6\}$ tem-se que $N \cup L = \{1, 2, 3, 4, 6\}$.

Como apontado em [63] alguns livros usam a notação $A + B$ para representar a união, é comum nesse caso não usar a nomenclatura união, em vez disso, é usado o termo soma de conjunto, entretanto, trata-se da mesma operação de união apresentada na definição anterior.

Definição 1.14 — Interseção de conjuntos. Sejam A e B dois conjuntos quaisquer, a interseção de A com B , denotada por $A \cap B$, corresponde ao seguinte conjunto.

$$A \cap B = \{x \mid x \in A \text{ e } x \in B\}$$

■ **Exemplo 1.12** Dado $A_1 = \{x \in \mathbb{N} \mid x \text{ é múltiplo de } 2\}$ e $A_2 = \{x \in \mathbb{N} \mid x \text{ é múltiplo de } 3\}$ tem-se que $A_1 \cap A_2 = \{x \in \mathbb{N} \mid x \text{ é múltiplo de } 6\}$.

■ **Exemplo 1.13** Seja $A = \{1, 2, 3\}$, $B = \{2, 3, 4, 5\}$ e $C = \{5\}$ tem-se que:

- (a) $A \cap B = \{2, 3\}$.
- (b) $A \cap C = \emptyset$.
- (c) $B \cap C = \{5\}$.
- (d) $B \cap B = \{2, 3, 4, 5\} = B$.

Com respeito as propriedades equacionais das operações de união e interseção tem-se como exposto em [64] os seguintes resultados para todo A, B e C .

identificador	None	União	Interseção
p_1	Idempotência	$A \cup A = A$	$A \cap A = A$
p_2	Comutatividade	$A \cup B = B \cup A$	$A \cap B = B \cap A$
p_3	Associatividade	$A \cup (B \cup C) = (A \cup B) \cup C$	$A \cap (B \cap C) = (A \cap B) \cap C$
p_4	Distributividade	$A \cup (B \cap C) = (A \cup B) \cap (A \cup C)$	$A \cap (B \cup C) = (A \cap B) \cup (A \cap C)$
p_5	Neutralidade	$A \cup \emptyset = A$	$A \cap \mathbb{U} = A$
p_6	Absorção	$A \cup \mathbb{U} = \mathbb{U}$	$A \cap \emptyset = \emptyset$

Tabela 1.1: Tabela das propriedades das operações de união e interseção.

Além das propriedades apresentadas pela Tabela 1.1, a união e a interseção possuem propriedades ligadas a relação de inclusão.

Teorema 1.3 Para quaisquer conjuntos A e B tem-se que:

- i. $A \subseteq (A \cup B)$.
- ii. $(A \cap B) \subseteq A$

Demonstração. Direta das Definições 1.7, 1.13 e 1.14. □

A partir da definição de interseção é estabelecido um conceito de extrema valia para a teoria dos conjuntos e suas aplicações, tal conceito é o estado de disjunção entre dois conjuntos.

Definição 1.15 — Conjuntos disjuntos. Dois conjuntos A e B são ditos disjuntos sempre que $A \cap B = \emptyset$.

■ **Exemplo 1.14** Seja $A = \{1, 2, 3\}$, $B = \{2, 3, 5\}$ e $C = \{5\}$ tem-se que A e C são disjuntos, por outro lado, A e B não são disjuntos entre si, além disso, B e C também não são disjuntos entre si.

Definição 1.16 — Complemento de conjuntos. Seja $A \subseteq \mathbb{U}$ para algum universo \mathbb{U} , o complemento de A , denotado por \bar{A} , corresponde ao seguinte conjunto:

$$\bar{A} = \{x \in \mathbb{U} \mid x \notin A\}$$

■ **Exemplo 1.15** Dado $P = \{x \in \mathbb{Z} \mid x = 2k \text{ para algum } k \in \mathbb{Z}\}$ tem-se então o seguinte complemento $\bar{P} = \{x \in \mathbb{Z} \mid x = 2k + 1 \text{ para algum } k \in \mathbb{Z}\}$.

■ **Exemplo 1.16** Dado um universo do discurso \mathbb{U} tem-se direto da definição que $\bar{\bar{\mathbb{U}}} = \emptyset$, e obviamente, $\bar{\emptyset} = \mathbb{U}$.

Teorema 1.4 Dado um conjunto A tem-se que:

i. $A \cup \bar{A} = \mathbb{U}$.

ii. $A \cap \bar{A} = \emptyset$.

iii. $\bar{\bar{A}} = A$.

Demonstração. Direta das Definições 1.13, 1.14 e 1.16. □

Observação 1.9 A propriedade (iii) apresentada no Teorema 1.4 costuma ser chamada involução, como dito em [63].

Além das propriedades apresentadas no Teorema 1.4 o complemento também apresenta propriedades ligadas diretamente a união e a interseção, tais propriedades são uma versão conjuntistas das famosas leis De Morgan (ver [21, 70, 64]) muito conhecidas pelos estudiosos da área de lógica, a seguir são apresentadas as leis De Morgan para a linguagem teoria dos conjuntos.

$$\begin{aligned} \text{(DM1) Lei De Morgan 1ª forma: } & \overline{(A \cup B)} = \bar{A} \cap \bar{B} \\ \text{(DM2) Lei De Morgan 2ª forma: } & \overline{(A \cap B)} = \bar{A} \cup \bar{B} \end{aligned}$$

Seguindo com este manuscrito, uma outra importante operação sobre conjuntos é a diferença entre conjuntos. A diferença entre conjunto apresenta duas formas, a primeira considerada por muito com a diferença natural [21] e existe também a diferença simétrica, que em um certo sentido, pode ser usada para medir a dissimetria entre conjuntos, ambas as operações são definidas formalmente a seguir.

Definição 1.17 — Diferença de conjuntos. Dado dois conjuntos A e B , a diferença de A e B , denotado por $A - B$ corresponde ao seguinte conjunto:

$$A - B = \{x \in A \mid x \notin B\}$$

■ **Exemplo 1.17** Dado os conjuntos $S = \{a, b, c, d\}$ e $T = \{f, b, g, d\}$ tem-se os seguintes conjuntos de diferença: $S - T = \{a, c\}$ e $T - S = \{f, g\}$.

■ **Exemplo 1.18** Dado os conjuntos \mathbb{Z} e \mathbb{Z}_+^* tem-se que $\mathbb{Z} - \mathbb{Z}_+^* = \mathbb{Z}_-$.

Observação 1.10 Note que o Exemplo 1.17 mostra que a operação de diferença de conjuntos não é comutativa.

Teorema 1.5 Para todo A e B tem-se que:

- i. $A - B = A \cap \overline{B}$.
- ii. Se $B \subset A$, então $A - B = \overline{B}$.

Demonstração. Dado os conjuntos A e B segue que:

- i. Por definição para todo $x \in A - B$ tem-se que $x \in A$ e $x \notin B$, mas isto só é possível se, e somente se, $x \in A$ e $x \in \overline{B}$, e por sua vez, isto só é possível se, e somente se, $x \in A \cap \overline{B}$, portanto, tem-se que $A - B = A \cap \overline{B}$.
- ii. Suponha que $B \subset A$, ou seja, todo $x \in B$ e tal que $x \in A$. Agora note que todo $x \in A - B$ é tal que $x \in A$ e $x \notin B$, e portanto, pela Definição 1.17 e pela hipótese de $B \subset A$ é claro que $A - B = \overline{B}$.

□

A seguir são apresentadas duas séries de igualdades notáveis relacionadas a diferença entre conjuntos.

Teorema 1.6 Sejam A e B subconjuntos de um universo \mathbb{U} , tem-se que:

- a. $A - \emptyset = A$ e $\emptyset - A = \emptyset$.
- b. $A - \mathbb{U} = \emptyset$ e $\mathbb{U} - A = \overline{A}$.
- c. $A - A = \emptyset$.
- d. $A - \overline{A} = A$.
- e. $\overline{(A - B)} = \overline{A} \cup B$.
- f. $A - B = \overline{B} - \overline{A}$.

Demonstração. Para todas as equações a seguir suponha que A e B são subconjuntos de um universo \mathbb{U} assim segue que:

a.

$$\begin{aligned} A - \emptyset &\stackrel{\text{Teo. 1.5(i)}}{=} A \cap \overline{\emptyset} \\ &= A \cap \mathbb{U} \\ &\stackrel{\text{Tab. 1.1}(p_5)}{=} A \end{aligned}$$

e também tem-se que,

$$\begin{aligned} \emptyset - A &\stackrel{\text{Teo. 1.5(i)}}{=} \emptyset \cap \overline{A} \\ &\stackrel{\text{Tab. 1.1}(p_6)}{=} \emptyset \end{aligned}$$

b. A prova tem um raciocínio similar a demonstração do item anterior, assim será deixado como exercício ao leitor.

c. Trivial pela própria Definição 1.17.

d.

$$\begin{aligned} A - \overline{A} &\stackrel{\text{Teo. 1.5(i)}}{=} A \cap \overline{\overline{A}} \\ &\stackrel{\text{Teo. 1.4(iii)}}{=} A \cap A \\ &\stackrel{\text{Tab. 1.1}(p_1)}{=} A \end{aligned}$$

e.

$$\begin{aligned} \overline{(A - B)} &\stackrel{\text{Teo. 1.5(i)}}{=} \overline{(A \cap \overline{B})} \\ &\stackrel{\text{(DM2)}}{=} \overline{A} \cup \overline{\overline{B}} \\ &\stackrel{\text{Teo. 1.4(iii)}}{=} \overline{A} \cup B \end{aligned}$$

f.

$$\begin{aligned}
A - B &\stackrel{\text{Teo. 1.5(i)}}{=} A \cap \overline{B} \\
&\stackrel{\text{Tab. 1.1}(p_2)}{=} \overline{B} \cap A \\
&\stackrel{\text{Teo. 1.4(iii)}}{=} \overline{B} \cap \overline{\overline{A}} \\
&\stackrel{\text{Teo. 1.5(i)}}{=} \overline{B} - \overline{A}
\end{aligned}$$

E assim a prova está concluída. □



Nota 1.2 Na demonstração do Teorema 1.6 apresentada anteriormente, algumas vezes foi escrito o símbolo de $=$ com um texto acima, isso é uma técnica comum na escrita de demonstrações matemáticas, o entendimento que leitor precisa ter é que ao escrever $\stackrel{\kappa}{=}$ significa que a igualdade segue (ou é garantida) pela propriedade ou resultado κ . Durante este manuscrito em algumas demonstrações uma escrita similar irá aparecer para outros símbolos (implicações, bi-implicações e etc.) que serão introduzidos no decorrer deste manuscrito.

Teorema 1.7 Sejam A, B e C subconjuntos de um universo \mathbb{U} , tem-se que:

- a. $(A - B) - C = A - (B \cup C)$.
- b. $A - (B - C) = (A - B) \cup (A \cap C)$.
- c. $A \cup (B - C) = (A \cup B) - (C - A)$.
- d. $A \cap (B - C) = (A \cap B) - (A \cap C)$.
- e. $A - (B \cup C) = (A - B) \cap (A - C)$.
- f. $A - (B \cap C) = (A - B) \cup (A - C)$.
- g. $(A \cup B) - C = (A - C) \cup (B - C)$.
- h. $(A \cap B) - C = (A - C) \cap (B - C)$.
- i. $A - (A - B) = A \cap B$.
- j. $(A - B) - B = A - B$.

Demonstração. Para todas as equações a seguir suponha que A, B e C são subconjuntos de um universo \mathbb{U} assim segue que:

a.

$$\begin{aligned}
(A - B) - C &\stackrel{\text{Teo. 1.5(i)}}{=} (A \cap \bar{B}) \cap \bar{C} \\
&\stackrel{\text{Tab. 1.1}(p_3)}{=} A \cap (\bar{B} \cap \bar{C}) \\
&\stackrel{\text{(DM1)}}{=} A \cap \overline{(B \cup C)} \\
&\stackrel{\text{Teo. 1.5(i)}}{=} A - (B \cup C)
\end{aligned}$$

b.

$$\begin{aligned}
A - (B - C) &\stackrel{\text{Teo. 1.5(i)}}{=} A \cap \overline{(B - C)} \\
&\stackrel{\text{Teo. 1.6(e)}}{=} A \cap (\bar{B} \cup C) \\
&\stackrel{\text{Tab. 1.1}(p_4)}{=} (A \cap \bar{B}) \cup (A \cap C) \\
&\stackrel{\text{Teo. 1.5(i)}}{=} (A - B) \cup (A \cap C)
\end{aligned}$$

c.

$$\begin{aligned}
A \cup (B - C) &\stackrel{\text{Teo. 1.5(i)}}{=} A \cup (B \cap \bar{C}) \\
&\stackrel{\text{Tab. 1.1}(p_4)}{=} (A \cup B) \cap (A \cup \bar{C}) \\
&\stackrel{\text{Tab. 1.1}(p_2)}{=} (A \cup B) \cap (\bar{C} \cup A) \\
&\stackrel{\text{Teo. 1.4(iii)}}{=} (A \cup B) \cap (\bar{C} \cup \bar{\bar{A}}) \\
&\stackrel{\text{(DM2)}}{=} (A \cup B) \cap \overline{(C \cap \bar{A})} \\
&\stackrel{\text{Teo. 1.5(i)}}{=} (A \cup B) - (C \cap \bar{A}) \\
&\stackrel{\text{Teo. 1.5(i)}}{=} (A \cup B) - (C - A)
\end{aligned}$$

d.

$$\begin{aligned}
A \cap (B - C) &\stackrel{\text{Teo. 1.5}(i)}{=} A \cap (B \cap \bar{C}) \\
&= \emptyset \cup (A \cap (B \cap \bar{C})) \\
&\stackrel{\text{Tab. 1.1}(p_2)}{=} \emptyset \cup ((A \cap B) \cap \bar{C}) \\
&\stackrel{\text{Tab. 1.1}(p_6)}{=} (\emptyset \cap B) \cup ((A \cap B) \cap \bar{C}) \\
&\stackrel{\text{Teo. 1.4}(ii)}{=} ((A \cap \bar{A}) \cap B) \cup ((A \cap B) \cap \bar{C}) \\
&\stackrel{\text{Tab. 1.1}(p_2, p_3)}{=} ((A \cap B) \cap \bar{A}) \cup ((A \cap B) \cap \bar{C}) \\
&\stackrel{\text{Tab. 1.1}(p_4)}{=} (A \cap B) \cap (\bar{A} \cup \bar{C}) \\
&\stackrel{\text{(DM2)}}{=} (A \cap B) \cap \overline{(A \cap C)} \\
&\stackrel{\text{Teo. 1.5}(i)}{=} (A \cap B) - (A \cap C)
\end{aligned}$$

e.

$$\begin{aligned}
A - (B \cup C) &\stackrel{\text{Teo. 1.5}(i)}{=} A \cap \overline{(B \cup C)} \\
&\stackrel{\text{(DM1)}}{=} A \cap (\bar{B} \cap \bar{C}) \\
&\stackrel{\text{Tab. 1.1}(p_1)}{=} (A \cap A) \cap (\bar{B} \cap \bar{C}) \\
&\stackrel{\text{Tab. 1.1}(p_3)}{=} ((A \cap A) \cap \bar{B}) \cap \bar{C} \\
&\stackrel{\text{Tab. 1.1}(p_2, p_3)}{=} ((A \cap \bar{B}) \cap A) \cap \bar{C} \\
&\stackrel{\text{Tab. 1.1}(p_3)}{=} (A \cap \bar{B}) \cap (A \cap \bar{C}) \\
&\stackrel{\text{Teo. 1.5}(i)}{=} (A - B) \cap (A - C)
\end{aligned}$$

f.

$$\begin{aligned}
A - (B \cap C) &\stackrel{\text{Teo. 1.5}(i)}{=} A \cap \overline{(B \cap C)} \\
&\stackrel{\text{(DM2)}}{=} A \cap (\bar{B} \cup \bar{C}) \\
&\stackrel{\text{Tab. 1.1}(p_4)}{=} (A \cap \bar{B}) \cup (A \cap \bar{C}) \\
&\stackrel{\text{Teo. 1.5}(i)}{=} (A - B) \cup (A - C)
\end{aligned}$$

g.

$$\begin{aligned}
(A \cup B) - C &\stackrel{\text{Teo. 1.5(i)}}{=} (A \cup B) \cap \bar{C} \\
&\stackrel{\text{Tab. 1.1}(p_4)}{=} (A \cap \bar{C}) \cup (B \cap \bar{C}) \\
&\stackrel{\text{Teo. 1.5(i)}}{=} (A - C) \cup (B - C)
\end{aligned}$$

h.

$$\begin{aligned}
(A \cap B) - C &\stackrel{\text{Teo. 1.5(i)}}{=} (A \cap B) \cap \bar{C} \\
&\stackrel{\text{Tab. 1.1}(p_4)}{=} (A \cap B) \cap (\bar{C} \cap \bar{C}) \\
&\stackrel{\text{Tab. 1.1}(p_2, p_3)}{=} (A \cap \bar{C}) \cap (B \cap \bar{C}) \\
&\stackrel{\text{Teo. 1.5(i)}}{=} (A - C) \cap (B - C)
\end{aligned}$$

i.

$$\begin{aligned}
A - (A - B) &\stackrel{\text{Teo. 1.5(i)}}{=} A \cap \overline{(A \cap \bar{B})} \\
&\stackrel{\text{(DM2)}}{=} A \cap (\bar{A} \cup \bar{\bar{B}}) \\
&\stackrel{\text{Tab. 1.1}(p_4)}{=} (A \cap \bar{A}) \cup (A \cap \bar{\bar{B}}) \\
&\stackrel{\text{Teo. 1.4(ii)}}{=} \emptyset \cup (A \cap \bar{\bar{B}}) \\
&\stackrel{\text{Tab. 1.1}(p_5)}{=} A \cap \bar{\bar{B}} \\
&\stackrel{\text{Teo. 1.4(iii)}}{=} A \cap B
\end{aligned}$$

j.

$$\begin{aligned}
(A - B) - B &\stackrel{\text{Teo. 1.5(i)}}{=} (A \cap \bar{B}) \cap \bar{B} \\
&\stackrel{\text{Tab. 1.1}(p_3)}{=} A \cap (\bar{B} \cap \bar{B}) \\
&\stackrel{\text{Tab. 1.1}(p_1)}{=} A \cap \bar{B} \\
&\stackrel{\text{Teo. 1.5(i)}}{=} A - B
\end{aligned}$$

□

Para prosseguir com esta seção sobre as operações definidas sobre conjuntos será agora apresentada a última operação “clássica”, sendo esta a diferença simétrica.

Definição 1.18 — Diferença simétrica. Dado dois conjuntos A e B , a diferença simétrica de A e B , denotado por $A \ominus B$, corresponde ao seguinte conjunto:

$$A \ominus B = \{x \mid x \in (A - B) \text{ ou } x \in (B - A)\}$$

Olhando atentamente a definição anterior é fácil notar que o conjunto da diferença simétrica é exatamente a união das possíveis diferenças entre os conjuntos, isto é, a diferença simétrica corresponde a seguinte igualdade: $A \ominus B = (A - B) \cup (B - A)$.

■ **Exemplo 1.19** Seja $A = \{1, 2, 3\}$ e $B = \{3, 4, 5, 2\}$ tem-se que $A \ominus B = \{1, 4, 5\}$.

A seguir será apresentada uma série de importantes resultados com respeito a diferença simétrica.

Teorema 1.8 Sejam A e B subconjuntos quaisquer de um determinado universo \mathbb{U} , tem-se que $A \ominus B = (A \cup B) \cap \overline{(A \cap B)}$.

Demonstração. Dado A e B dois subconjuntos quaisquer de um determinado universo \mathbb{U} segue que:

$$\begin{aligned} A \ominus B &= (A - B) \cup (B - A) \\ &\stackrel{\text{Teo. 1.5(i)}}{=} (A \cap \overline{B}) \cup (B \cap \overline{A}) \\ &\stackrel{\text{Tab. 1.1}(p_4)}{=} (A \cup (B \cap \overline{A})) \cap (\overline{B} \cup (B \cap \overline{A})) \\ &\stackrel{\text{Tab. 1.1}(p_4)}{=} ((A \cup B) \cap (A \cup \overline{A})) \cap ((\overline{B} \cup B) \cap (\overline{B} \cup \overline{A})) \\ &\stackrel{\text{Teo. 1.4(i)}}{=} ((A \cup B) \cap \mathbb{U}) \cap (\mathbb{U} \cap (\overline{B} \cup \overline{A})) \\ &\stackrel{\text{Tab. 1.1}(p_1, p_5)}{=} (A \cup B) \cap (\overline{B} \cup \overline{A}) \\ &\stackrel{\text{(DM2)}}{=} (A \cup B) \cap \overline{(B \cap A)} \\ &\stackrel{\text{Tab. 1.1}(p_2)}{=} (A \cap B) \cap \overline{(A \cap B)} \end{aligned}$$

□

Corolário 1.1 Sejam A e B subconjuntos quaisquer de um determinado universo \mathbb{U} , tem-se que $A \ominus B = (A \cup B) - (A \cap B)$.

Demonstração. Pelo Teorema 1.8 tem-se que $A \ominus B = (A \cup B) \cap \overline{(A \cap B)}$, mas pelo Teorema 1.5 (i) segue que $(A \cup B) \cap \overline{(A \cap B)} = (A \cup B) - (A \cap B)$, e portanto, $A \ominus B = (A \cup B) - (A \cap B)$. □

O próximo resultado mostra que a operação de diferença simétrica entre conjunto possui elemento neutro, isto é, existe um conjunto que quando operado com qualquer outro conjunto A , o resultado é o próprio conjunto A .

Teorema 1.9 Para todo A tem-se que $A \ominus \emptyset = A$.

Demonstração. Dado um conjunto A qualquer pelo Corolário 1.1 tem-se que $A \ominus \emptyset = (A \cup \emptyset) - (A \cap \emptyset)$, mas pelas propriedades apresentadas na Tabela 1.1 tem-se: $A \cup \emptyset = A$ e $A \cap \emptyset = \emptyset$. Logo $A \ominus \emptyset = A - \emptyset$, por fim, pelo Teorema 1.6 (a) tem-se que $A - \emptyset = A$, consequentemente, $A \ominus \emptyset = A$. \square

Seguindo com as propriedades que a operação de diferença simétrica possui, o próximo resultado mostra a existência de um elemento que neste manuscrito será chamado de **alternador**, isto é, existe um conjunto que quando operado com qualquer outro conjunto A , o resultado é o complemento deste conjunto A .

Teorema 1.10 Para todo A tem-se que $A \ominus \mathbb{U} = \bar{A}$.

Demonstração. Similar a demonstração do Teorema 1.9, ficando assim como exercício ao leitor. \square

O teorema a seguir mostra que a diferença simétrica entre um conjunto A e seu complementar \bar{A} é exatamente igual a totalidade do universo do discurso em que estes conjuntos estão inseridos.

Teorema 1.11 Para todo A tem-se que $A \ominus \bar{A} = \mathbb{U}$.

Demonstração. Dado um conjunto A qualquer e seu complementar \bar{A} tem-se pelo Corolário 1.1 que $A \ominus \bar{A} = (A \cup \bar{A}) - (A \cap \bar{A})$, mas pelo Teorema 1.4 tem-se que $A \cup \bar{A} = \mathbb{U}$ e $A \cap \bar{A} = \emptyset$, consequentemente, $A \ominus \bar{A} = \mathbb{U} - \emptyset$, mas pelo Teorema 1.6 tem-se que $\mathbb{U} - \emptyset = \mathbb{U}$, e portanto, $A \ominus \bar{A} = \mathbb{U}$. \square

Continuando a estudar a diferença simétrica o próximo teorema mostra que a diferença simétrica entre um conjunto A e ele mesmo é exatamente igual ao conjunto vazio.

Teorema 1.12 Para todo A tem-se que $A \ominus A = \emptyset$.

Demonstração. Dado um conjunto A qualquer tem-se pelo Corolário 1.1 que vale a seguinte igualdade, $A \ominus A = (A \cup A) - (A \cap A)$. Mas pelas propriedades apresentadas na Tabela 1.1 tem-se que $(A \cup A) = (A \cap A) = A$, logo $A \ominus A = A - A$, mas pelo Teorema 1.6 tem-se que $A - A = \emptyset$, portanto, $A \ominus A = \emptyset$. \square

Anteriormente foi mostrado que a diferença entre conjuntos não era comutativa (Exemplo 1.17), o próximo resultado contrasta esse fato com respeito a diferença simétrica.

Teorema 1.13 Para todo A e B tem-se que $A \ominus B = B \ominus A$.

Demonstração. Dado dois conjuntos A e B tem-se pelo Corolário 1.1 que vale a seguinte igualdade, $A \ominus B = (A \cup B) - (A \cap B)$, mas pela propriedade de comutatividade de \cup e de \cap (ver Tabela 1.1) tem-se que $A \cup B = B \cup A$ e $A \cap B = B \cap A$, logo tem-se que $A \ominus B = (B \cup A) - (B \cap A)$, mas pelo Corolário 1.1 tem-se que $(B \cup A) - (B \cap A) = B \ominus A$, e portanto, $A \ominus B = B \ominus A$. \square

Teorema 1.14 Para todo A, B e C tem-se que $(A \ominus B) \ominus C = A \ominus (B \ominus C)$.

Demonstração. A prova deste teorema sai direto da definição de diferença simétrica e assim ficará como exercício ao leitor. \square

Teorema 1.15 Para todo A e B tem-se que $\overline{(A \ominus B)} = (A \cap B) \cup (\bar{A} \cap \bar{B})$.

Demonstração. Para todo A e B segue que:

$$\begin{aligned}
 \overline{(A \ominus B)} &\stackrel{\text{Teo. 1.8}}{=} \overline{((A \cup B) \cap \overline{(A \cap B)})} \\
 &\stackrel{\text{(DM2)}}{=} \overline{(A \cup B)} \cup \overline{\overline{(A \cap B)}} \\
 &\stackrel{\text{(DM1)}}{=} (\bar{A} \cap \bar{B}) \cup \overline{\overline{(A \cap B)}} \\
 &\stackrel{\text{(DM2)}}{=} (\bar{A} \cap \bar{B}) \cup \overline{(A \cup B)} \\
 &\stackrel{\text{(DM1)}}{=} (\bar{A} \cap \bar{B}) \cup (\bar{\bar{A}} \cap \bar{\bar{B}}) \\
 &\stackrel{\text{Tab. 1.1}(p_2)}{=} (\bar{\bar{A}} \cap \bar{\bar{B}}) \cup (\bar{A} \cap \bar{B}) \\
 &\stackrel{\text{Teo. 1.4}(iii)}{=} (A \cap B) \cup (\bar{A} \cap \bar{B})
 \end{aligned}$$

\square

1.3 Operações generalizadas

Agora após a apresentação de todas as operações básicas sobre conjuntos e suas principais propriedades, este manuscrito irá continuar o estudo da teoria ingênua dos conjuntos pela forma generalizada das operações de união e interseção.

Definição 1.19 — União generalizada. Dado uma família A então a união generalizada dos conjuntos em A corresponde respectivamente a:

$$A_{\cup} = \bigcup_{x \in A} x$$

■ **Exemplo 1.20** Dado a família $A = \{\{2, 4\}, \{-1, 2\}, \{4, 9, 8, -1\}\}$ tem-se que:

$$A_{\cup} = \{2, 4, -1, 9, 8\}$$

■ **Exemplo 1.21** Seja $A = \{\{a, b\}, \{a\}, \{b\}, \{c\}\}$ tem-se que a união generalizada dos elementos de A corresponde ao conjunto $A_{\cup} = \{a, b, c\}$.

É fácil perceber pela própria definição que a união generalizada só será vazia se todos os membros da família A forem exatamente iguais ao conjunto vazio. De forma dual tem-se a definição generalizada da interseção como se segue.

Definição 1.20 — Interseção generalizada. Dado uma família A então a interseção generalizada dos conjuntos em A corresponde respectivamente a:

$$A_{\cap} = \bigcap_{x \in A} x$$

■ **Exemplo 1.22** Seja $D = \{\mathbb{Z}_+, \{0, -1, -2, -3\}, (\mathbb{Z}_- \cup \{0\})\}$, a interseção generalizada de D corresponde ao conjunto $D_{\cap} = \{0\}$.

■ **Exemplo 1.23** Dado $A = \{\{a, t, c, g\}, \{v, x, a, g, d\}, \{z, b, a, y, g\}, \{g, b, a\}\}$ tem-se que $A_{\cap} = \{a, g\}$.

Observação 1.11 Vale destacar que as igualdades nas Definições 1.19 e 1.20 são sustentadas pelas propriedades da idempotência, associatividade e comutatividade descritas na Tabela 1.1, para mais detalhes consulte [21].

Como dito em [21, 64], quando A é uma família com uma quantidade de n conjuntos, isto é, quanto tem-se que $A = \{x_1, \dots, x_n\}$, é comum reescrever a definição da união e da interseção generalizada usando as seguintes igualdades:

$$A_{\cup} = x_1 \cup \dots \cup x_n$$

e

$$A_{\cap} = x_1 \cap \dots \cap x_n$$

ou ainda:

$$A_{\cup} = \bigcup_{i=1}^n x_i$$

e

$$A_{\cap} = \bigcap_{i=1}^n x_i$$

Teorema 1.16 Se A é uma família, então:

i. $\overline{A_{\cup}} = \bigcap_{x \in A} \bar{x}.$

ii. $\overline{A_{\cap}} = \bigcup_{x \in A} \bar{x}.$

Demonstração. A prova segue da aplicação das leis De Morgan, e ficará como exercício ao leitor. \square

1.4 Partes e Partições

Como já mencionando algumas vezes anteriormente uma família é um conjunto cujo os elementos são também conjuntos. Agora dado um conjunto A qualquer, em algum momento possa ser que seja necessário (por interesse prático ou teórico) trabalhar com a família dos subconjuntos deste conjunto A , note porém, que qualquer elemento desta família é uma parte do conjunto A , ou seja, a família reuni as partes de A , a seguir é definido formalmente o conceito de família das partes obtida a partir de um determinado conjunto.

Definição 1.21 — Conjunto das partes. Seja A um conjunto. O conjunto das partes^a de A , é denotada por $\wp(A)$, e corresponde a seguinte família de conjuntos:

$$\wp(A) = \{x \mid x \subseteq A\}$$

^aEm alguns livros é usado o termo conjunto potência em vez do termo conjunto das partes, nesse caso é usado a notação 2^A para denotar tal família de conjuntos, por exemplo ver [64].

Uma propriedade interessante do conjuntos das partes como dito em [63], é que se A for da forma $A = \{x_1, \dots, x_n\}$ para algum $n \in \mathbb{N}$, então pode-se mostrar que $\wp(A)$ terá exatamente 2^n elementos.

■ **Exemplo 1.24** Seja $A = \{a, b, c\}$ tem-se que o conjunto das parte de A corresponde a família de conjunto $\{\emptyset, \{a\}, \{b\}, \{c\}, \{a, b\}, \{a, c\}, \{c, b\}, \{a, b, c\}\}$.

■ **Exemplo 1.25** Dado o conjunto $X = \{1\}$ tem-se que $\wp(X) = \{\emptyset, \{1\}\}$.

■ **Exemplo 1.26** Seja $A = \emptyset$ tem-se que $\wp(A) = \{\emptyset\}$.

Além da ideia de conjunto das partes, uma outra família muito importante dentro da teoria dos conjuntos é a família das partições de um conjunto.

Definição 1.22 — Partição. Seja A um conjunto não vazio, uma partição é uma família não vazia de subconjuntos disjuntos de A , ou seja, uma família $\{x_i \mid x_i \subseteq A\}$ tal que as seguintes condições são satisfeitas:

- (1) Para todo $y \in A$ tem-se que existe um único i tal que $y \in x_i$ para algum $x_i \subseteq A$.
- (2) Para todo i e todo j sempre que $i \neq j$, então $x_i \cap x_j = \emptyset$.

É fácil notar pela Definição 1.22 que partições são famílias, além disso, como dito em [64] os elementos em uma partição são chamados de **células**, isto é, dado um conjunto A os subconjuntos na partição de A são vistos como as células que formam o próprio conjunto A . O resultado a seguir garante que sempre é possível obter pelo menos uma partição de um conjunto.

Teorema 1.17 Se A é um conjunto não vazio, então existe pelo menos uma partição de A .

Demonstração. Suponha que o conjunto A seja não vazio, assim defina o conjunto $PT_A = \{\{x\} \mid x \in A\}$, agora claramente tem-se que PT_A é uma família e satisfaz todas as condições da Definição 1.22 e, portanto, PT_A é uma partição do conjunto A . \square



Nota 1.3 Neste manuscrito a partição descrita no Teorema 1.17 será chamada de **partição trivial**.

■ **Exemplo 1.27** Dado o conjunto $A = \{0, 1, 2, 3, 4, 5\}$ tem-se que:

- (a) $R = \{\{1, 5\}, \{2, 1, 4\}, \{0, 3\}\}$ não é uma partição de A pois $\{1, 5\} \cap \{2, 1, 4\} = \{1\}$, e portanto, não são disjuntos.
- (b) $S = \{\{1, 5\}, \{0, 4\}, \{3\}\}$ não é uma partição de A pois o elemento $2 \in A$ não pertence a nenhum dos conjuntos em S .
- (c) $T_1 = \{\{0, 5\}, \{1, 3, 4\}, \{2\}\}$ e $T_2 = \{\{0, 1\}, \{4, 5\}, \{3, 2\}\}$ são ambas partições do conjunto A pois satisfazem todas as condições apresentadas na Definição 1.22.

Observação 1.12 É claro que uma partição de um conjunto A é vazia se, e somente se, $A = \emptyset$.

1.5 Questionário

■ **Exercício 1.1** Para cada um dos conjuntos a seguir, determine uma propriedade que define o conjunto e escreva os conjuntos na notação compacta.

- (a). $\{0, 2, 4, 6, 8, 1, 3, 5, 7, 9\}$.
- (b). $\{-2, -4, -6, -8, 0, 6, 4, 8, 2\}$.
- (c). $\{3, 5, 7, 9, 11, 13, 15, 17, \dots\}$.
- (d). $\{a, c, s\}$
- (e). $\{2, 3, 5, 7, 11, 13, 17, 19, \dots\}$.
- (f). $\{1, 4, 9, 16, 25, 36, 64, 81, 100\}$.
- (g). $\{3, 6, 9, 12, 15, 18, 21, \dots\}$.
- (h). $\{\frac{1}{2}, \frac{2}{4}, \frac{3}{6}, \frac{4}{8}, \frac{5}{10}, \dots\}$

■ **Exercício 1.2** Escreva os seguintes conjuntos em notação compacta.

- (a). Conjunto de todos os países da América do sul.
- (b). Conjunto de planetas do sistema solar.
- (c). Conjunto dos números reais maiores que 1 e menores que 2.
- (d). Conjunto de estados brasileiros cujo nome começa com a letra “R”.
- (e). Conjunto dos times nordestinos que já foram campeões da primeira divisão do campeonato brasileiro de futebol.

■ **Exercício 1.3** Escreva as sentença a seguir de forma apropriada usando a linguagem da teoria dos conjuntos.

- (a). x não pertence ao conjunto A .
- (b). -2 não é um número natural.
- (c). O símbolo π representa um número real.
- (d). O conjunto das vogais não é subconjunto do conjunto das consoantes.
- (e). y é um número inteiro, porém não é um número maior que 10.
- (f). D é o conjunto de todos os múltiplos de -3 que são maiores que 1.

■ **Exercício 1.4** Considere o conjunto de letras $K = \{b, t, s\}$ responda falso ou verdadeiro e justifique sua resposta:

- (a). $s \in K$?
- (b). $t \subset K$?
- (c). $K \not\subseteq K$?
- (d). $\{b\} \in K$?
- (e). $K - \{a\} = K$?

■ **Exercício 1.5** Considere cada conjunto a seguir e escreva todos os seus subconjuntos.

- (a). $B = \{1, 2, 3\}$.
- (b). $F = \{a, b, c, d\}$.
- (c). $N = \{\emptyset\}$.

(d). $R = \{\emptyset, \{\emptyset\}\}$.

(e). $P = \{\{a, b\}, \{c, d\}, \{a, f\}, \{a, b, c\}, \emptyset\}$.

■ **Exercício 1.6** Considerando o universo dos números naturais dado os subconjuntos: $A = \{1, 2, 3, 4, 5\}$, $B = \{x \in \mathbb{N} \mid x^2 = 9\}$, $C = \{x \in \mathbb{N} \mid x^2 - 4x + 6 = 0\}$ e $D = \{x \in \mathbb{N} \mid x = 2k \text{ para algum } k \in \mathbb{N}\}$, complete a frase com os símbolos \subseteq e $\not\subseteq$.

(a). $A \underline{\hspace{1cm}} B$.

(b). $C \underline{\hspace{1cm}} B$.

(c). $D \underline{\hspace{1cm}} C$.

(d). $B \underline{\hspace{1cm}} A$.

(e). $A \underline{\hspace{1cm}} D$.

(f). $C \underline{\hspace{1cm}} A$.

(g). $D \underline{\hspace{1cm}} B$.

(h). $B \underline{\hspace{1cm}} \mathbb{N}$.

(i). $\mathbb{N} \underline{\hspace{1cm}} D$.

(j). $A \underline{\hspace{1cm}} \mathbb{N}$.

(k). $A \underline{\hspace{1cm}} \mathbb{Z}_-$.

(l). $\mathbb{N} \underline{\hspace{1cm}} D$.

(m). $\mathbb{N} \underline{\hspace{1cm}} C$.

(n). $C \underline{\hspace{1cm}} \mathbb{N}$.

(o). $\{6\} \underline{\hspace{1cm}} C$.

■ **Exercício 1.7** Complete as sentença da teoria dos conjuntos com \in , \subseteq e $\not\subseteq$.

(a). $2 \underline{\hspace{1cm}} \{1, 2, 3\}$.

(b). $\{2\} \underline{\hspace{1cm}} \{1, 2, 3\}$.

(c). $\{1\} \underline{\hspace{1cm}} \{\{1\}, \{2\}, \{3\}\}$.

(d). $\emptyset \underline{\hspace{1cm}} \{1\}$.

- (e). $\emptyset ___ \{\emptyset\}$.
- (f). $\{3\} ___ \emptyset$.
- (g). $\mathbb{N} ___ \{2, 3, 6\}$.
- (h). $\{\{\emptyset\}, \emptyset\} ___ \{\{\{\emptyset\}, \emptyset\}, \emptyset\}$.
- (i). $a ___ \{\{a\}, b\} ___ \{a, b, c\}$.
- (j). $0 ___ \mathbb{Z}_+^*$.
- (k). $\frac{1}{0} ___ \mathbb{Q}$.
- (l). $\{0, 1\} ___ \{0, 1, 2, 5\} ___ \mathbb{N}$.
- (m). $\mathbb{N} ___ \mathcal{P}(\mathbb{N})$
- (n). $\emptyset ___ \emptyset$.
- (o). $\{1, 2, 4\} ___ \{2, 4, 6\} ___ \{y \mid y = 2x \text{ para algum } x \in \mathbb{N}\}$.
- (p). $\{1\} ___ \mathbb{R}$.
- (q). $\frac{3}{4} ___ \mathbb{N}$.

■ **Exercício 1.8** Justifique as seguintes afirmações.

- (a). $\{\frac{2}{x} \mid x - 1 > 0 \text{ com } x \in \mathbb{N}\}$ não é subconjunto de \mathbb{N} .
- (b). $\{2, 3, 4, 6, 8\}$ não é subconjunto de $\{x \in \mathbb{N} \mid x = 2k \text{ para algum } k \in \mathbb{N}\}$.
- (c). $\{1, 2, 3\}$ é um subconjunto próprio do conjunto $\{1, 2, 3, 4, 5, 6, 7, 8, 9, 0\}$.
- (d). $\{0, 5\}$ é subconjunto de \mathbb{Z} mas não é subconjunto de \mathbb{Z}^* .
- (e). $\{x \mid x + x = x\}$ é subconjunto próprio de \mathbb{N} .
- (f). Existem exatamente 15 subconjuntos próprios do conjunto $\{2, 3, 5, 7\}$.
- (g). Não existem subconjuntos próprios do conjunto \emptyset .
- (h). Sempre que $A \subset B$ e $A_0 \subset A$, tem-se que A_0 é também um subconjunto próprio de B .
- (i). O conjunto $\{2\}$ tem um único subconjunto próprio.
- (j). O conjunto $\{x \in \mathbb{N} \mid 0 < x < 3\}$ tem exatamente 3 subconjuntos próprios.

■ **Exercício 1.9** Considerando o universo $\mathbb{U} = \{1, 2, 3, 4, 5, 6, 7, 8, 9, 0\}$ e seus subconjuntos $A = \{2, 4, 6, 8\}$, $B = \{1, 3, 5, 7, 9\}$, $C = \{1, 2, 3, 4, 0\}$ e $D = \{0, 1\}$ exiba os conjuntos a seguir.

- (a). $A \cup B$.
- (b). $C \cup D$.
- (c). $D \cap A$.
- (d). $B \cap C$.
- (e). $A \cap (B \cup D)$.
- (f). $D \cap (A \cap C)$.
- (g). $(A \cap B) \cup (D \cap C)$.
- (h). $(\mathbb{U} \cap A) \cup D$.
- (i). $(D \cup A) \cap C$.
- (j). $D \cap (B \cup A)$.
- (k). $A \cup \bar{B}$.
- (l). $\overline{(C \cap B)} \cup D$.
- (m). $\overline{D \cap A}$.
- (n). $B \cap \bar{C}$.
- (o). $A \cap \overline{(B \cup D)}$.
- (p). $D \cap (A \cap C)$.
- (q). $\overline{(A \cap B)} \cup (\bar{D} \cap C)$.
- (r). $\overline{(\mathbb{U} \cap \bar{A})} \cup D$.
- (s). $(D \cup A) \cap \bar{C}$.
- (t). $\overline{D \cap (B \cup A)}$.
- (u). $\overline{D - A}$.
- (v). $(A - B) \cap \bar{C}$.
- (w). $A \cap \overline{(B - D)}$.

(x). $D \cap (A - C)$.

(y). $\overline{C} - D$.

(z). $D - A$.

■ **Exercício 1.10** Considerando o universo $\mathbb{U} = \{a, b, c, d, e, f, g, h, i, j\}$ e seus subconjuntos $A = \{b, d, f, h\}$, $B = \{a, c, e, g, i\}$, $C = \{a, b, c, d, j\}$ e $D = \{a, j\}$ exiba os seguintes conjuntos.

(a). $\overline{B} - C$.

(b). $A - (B \cup D)$.

(c). $(A - (A \cap B)) - ((\overline{D} \cap C) - A)$

(d). $\overline{(\mathbb{U} - \overline{C})} - D$

(e). $A \oplus (B \cup D)$.

(f). $(A \oplus (A \cap B)) \oplus ((\overline{D} \cap C) \oplus A)$

(g). $\overline{(\mathbb{U} \oplus \overline{C})} \oplus D$

(h). $\overline{D \oplus A}$.

(i). $(A \oplus B) \cap \overline{C}$.

(j). $\overline{C} \oplus D$.

(k). $D \oplus A$.

(l). $\overline{B} \oplus C$.

(m). $A \cap \overline{(\overline{B} \oplus D)}$.

(n). $D \cap (A \oplus C)$.

■ **Exercício 1.11** Uma aluna do curso de Ciência da Computação realizou uma pesquisa sobre três ritmos (A, B e C) presentes no aplicativo de música *Spotify* com seus colegas de classe para seu trabalho na disciplina de estatística, e levantou os dados expostos na Tabela 1.2.

Total de entrevistados	Ouvem A	Ouvem B	Ouvem C	Ouvem A e B	Ouvem A e C	Ouvem B e C	Ouvem A, B e C	Não ouvem nenhum dos ritmos
23	8	4	6	2	3	1	1	10

Tabela 1.2: Tabela com dados fictício da pesquisa sobre ritmos no *Spotify*.

- (a). Qual é o número de entrevistados que escutam apenas o ritmo A?
- (b). Qual é o número de entrevistados que escutam o ritmo A e não escutam o ritmo B?
- (c). Quantos entrevistados não escutam o ritmo C?
- (d). Qual é o número de entrevistados que escutam algum dos ritmos?
- (e). Quantos entrevistados escutam o ritmo B ou C, mas não escutam o ritmo A?

■ **Exercício 1.12** Dado os conjuntos $A = \{1, 2, 3\}$, $B = \{3, 4, 5\}$ e $C = \{1, 5, 6\}$ construa um conjunto X com exatamente 4 elementos tal que $A \cap X = \{3\}$, $B \cap X = \{3, 5\}$ e $C \cap X = \{5, 6\}$.

■ **Exercício 1.13** Considere o banco de dados representado na Tabela 1.3. para esboçar o conjunto gerado por cada *Query* detalhada abaixo, relacionando as mesmas com as operações sobre conjuntos..

id	Nome	Salário	Idade	Sexo
23	Júlio	2.300,00	34	M
102	Patrícia	4.650,00	23	F
33	Daniel	1.375,00	20	M
43	Renata	6.400,00	24	F
53	Rafaela	1.800,00	19	F
57	Tadeu	14.450,00	54	M

Tabela 1.3: Uma base de dados representada como uma tabela.

- (a). O conjuntos dos id's onde o sexo é igual a F e o salário não é inferior a 2.000,00.
- (b). O conjunto dos salários em que a idade não é superior a 35 ou o sexo é igual a M.
- (c). O conjunto de todos os nome em que a idade não é maior que 30 ou id é menor que 65.

■ **Exercício 1.14** Exiba os seguintes conjuntos.

- (a). $\mathcal{P}(\{1, 2, 3\})$.
- (b). $\mathcal{P}(\mathcal{P}(\{0, 1\}))$.
- (c). $\mathcal{P}(\{\mathbb{N}\})$.
- (d). $\mathcal{P}(\{1, \{2\}, \{1, \{2\}\}\})$.
- (e). $\mathcal{P}(\{1, \{1\}, \{2\}, \{3, 4\}\})$.
- (f). $\mathcal{P}(\mathcal{P}(\{1, 2\})) - \mathcal{P}(\{0, 1\})$.
- (g). $\mathcal{P}(\{a, b, c, g\} \ominus \{g, e, f, d\})$.

(h). $\mathcal{P}(\mathcal{P}(\mathcal{P}(\{0,1\})) \cup \mathcal{P}(\{1,2,3\})).$

(i). $\mathcal{P}(\mathcal{P}(\emptyset) - \emptyset).$

(j). $\mathcal{P}(\{2,3,4\} \cap (\{-1,3\} \cup \{-5\})).$

■ **Exercício 1.15** Considere o universo $\mathbb{U} = \{a,b,c,d,e,f,g\}$ e seus subconjuntos $A = \{d,e,g\}, B = \{a,c\}, C = \{b,e,g\}$ calcule e exiba os seguintes conjuntos.

(a). $\mathcal{P}(C).$

(b). $\mathcal{P}(A) - \mathcal{P}(\overline{B}).$

(c). $\mathcal{P}((A \cup B) \ominus C).$

(d). $\mathcal{P}((\overline{A} \cup B)) \ominus \mathcal{P}(C).$

(e). $\mathcal{P}(\overline{(C \cap B)} - (\overline{A} \cap C))$

(f). $\mathcal{P}(C) - (\mathcal{P}(A) \ominus \mathcal{P}(B)).$

(g). $\mathcal{P}(\overline{A}) \ominus ((\mathcal{P}(C) \cap \mathcal{P}(B)) - \mathcal{P}(A)).$

(h). $\mathcal{P}(\mathcal{P}(A)) - \mathcal{P}(\mathcal{P}(B)).$

(i). $\mathcal{P}(\mathcal{P}(\overline{C})) \ominus \mathcal{P}(\mathcal{P}(B)).$

(j). $\mathcal{P}(\mathbb{U}).$

■ **Exercício 1.16** Dado o conjunto $A = \{a,b,c,d,e,f,g\}$ diga se as famílias de conjuntos a seguir são ou não partições de A , justifique todas as suas resposta.

(a). $P_1 = \{\{a,c,e\}, \{b\}, \{d,g\}\}.$

(b). $P_2 = \{\{a,g,e\}, \{c,d\}, \{b,e,f\}\}.$

(c). $P_3 = \{\{a,b,e,g\}, \{c\}, \{d,f\}\}.$

(d). $P_4 = \{\{a,b,c,d,e,f,g\}\}.$

(e). $P_5 = \{\{a,b,d,e,g\}, \{f,c\}\}.$

(f). $P_6 = \{\{a,b,c,d,e\}, \{e,f,g\}\}.$

(g). $P_7 = \{\{b,c,d,e,f,g\}, \{a\}, \{b,a,c\}\}.$

(h). $P_8 = \{\{a,b,c,d,e,f,g\}, \{e,d\}\}.$

(i). $P_9 = \{\{a\}, \{b\}, \{c\}, \{d, e\}, \{f\}, \{g\}\}.$

(j). $P_{10} = \{\{a\}, \{b\}, \{c\}, \{d\}, \{e\}, \{f\}, \{g\}\}.$

■ **Exercício 1.17** Considere o universo $\mathbb{U} = \{a, b, c, d, e, f, g\}$ e seus subconjuntos $A = \{d, e, g\}, B = \{a, c\}, C = \{b, e, g\}$ exiba duas partições diferentes da partição trivial para cada um dos conjuntos a seguir.

(a). $C - \bar{A}.$

(b). $A - \bar{B}.$

(c). $(A \cup B) \ominus C.$

(d). $(\bar{A} \cup B) \ominus C.$

(e). $\overline{(C \cap B)} - (\bar{A} \cap C)$

(f). $C - (A \ominus B).$

(g). $\bar{A} \ominus ((C \cap B) - A).$

(h). $A - B.$

(i). $\bar{C} \ominus B.$

(j). $\wp(\mathbb{U}).$

Capítulo 2

Métodos de Demonstração

“Mais um colchão, mais uma demonstração”.

Paul Erdős

“Um matemático é uma máquina que transforma café em teoremas”.

Paul Erdős

2.1 Introdução

No capítulo anterior, o leitor encontrou diversas demonstrações dentro da teoria intuitiva (ou Cantoriana) dos conjuntos. Para um leitor iniciante talvez tenha sido um tanto quanto complicado entender a metodologia usada para construir tais demonstrações. E desde que, as demonstrações são figuras de interesse central no cotidiano dos matemáticos, cientistas da computação e engenheiros de software, em especial aqueles que trabalham com métodos formais, este texto irá fazer uma breve pausa no estudo da teoria dos conjuntos, para apresentar um pouco de teoria da prova ao leitor.

Este capítulo começa então com o seguinte questionamento: Do ponto de vista da ciência da computação qual a importância das demonstrações? Bem a resposta a essa pergunta pode ser dada de dois pontos de vista, um teórico (purista) e um prático (aplicado ou de engenharia).

Na perspectiva de um cientista da computação puro, as demonstrações de teoremas, proposições, lema, corolários e propriedades são a principal ferramenta para investigar os limites dos diferentes modelos de computação propostos [49, 61], assim sendo é de suma importância que o estudante de graduação em ciência da computação receba em sua formação pelo menos o básico para dominar a “arte” de provar teoremas, sendo assim preparado para o estudo e a pesquisa pura em computação e(ou) matemática.

Já na visão prática, só existe uma forma segura de garantir que um *software* está livre de erros, essa “tecnologia” é exatamente a demonstração das propriedades do *software*. É claro que, mostrar que um *software* não possui erros vai exigir que o *software* seja visto através de um certo nível de formalismo e rigor matemático, mas após essa modelagem através de demonstrações pode-se garantir que um *software* não apresentará erros (quando bem especificado), e assim se algo errado ocorrer foi por fatores externos, tais como defeito no *hardware* por exemplo, e não por falha ou erros com a implementação. Este conceito é o cerne de uma área da engenharia de *software* [90], chamada métodos (ou especificações) formais, sendo essa área o ponto crucial no desenvolvimento de *softwares* para sistemas críticos [101]. Isto já mostra a grande importância de programadores e engenheiros de *software* terem em sua formação as bases para o domínio das técnicas de demonstração.

Nas próximas seções deste manuscrito serão descritas as principais técnicas de demonstração de interesse de matemáticos, cientistas da computação e engenheiros formais de *software*.

Observação 2.1 Para o leitor que nunca antes teve contato com a lógica matemática recomenda-se que antes de estudar este capítulo, o leitor faça pelo menos um rápido estudo do Capítulo 6.

Para poder falar sobre métodos de demonstração e poder então descrever como os matemáticos, lógicos e cientistas da computação justificam propriedades usando apenas a argumentação matemática, será necessário fixar algumas nomenclaturas e falar sobre alguns conceitos importantes.

Definição 2.1 — Asserção. Uma **asserção** é qualquer frase declarativa que possa ser expressa na linguagem da lógica simbólica.

Observação 2.2 O leitor que conheça lógica nota facilmente que uma asserção é uma proposição ou predicado, para detalhes ver o Capítulo 6.

Os métodos (ou estratégias) de demonstrações apresentadas neste manuscrito seguem as ideias e a ordem de apresentação similar ao que foi exposto em [108]. Em [108] antes de apresentar as provas formais, era necessário a construção de um rascunho de prova, este rascunho possui similaridades com as demonstrações em provadores de teoremas tais como Coq [16, 88] e Lean [80], isto é, existe uma separação clara entre dados (hipótese) e os objetivos (em inglês *Goal*) que se quer demonstrar.

Neste manuscrito por outro lado, não será utilizado a ideia de um rascunho de prova, em vez disso, será usado aqui a noção de **diagrama de blocos** [18]. Aqui tais diagramas serão encarados como as demonstrações em si, assim diferente de [108] não haverá a necessidade de escrever um texto formal após o diagrama da prova ser completado.

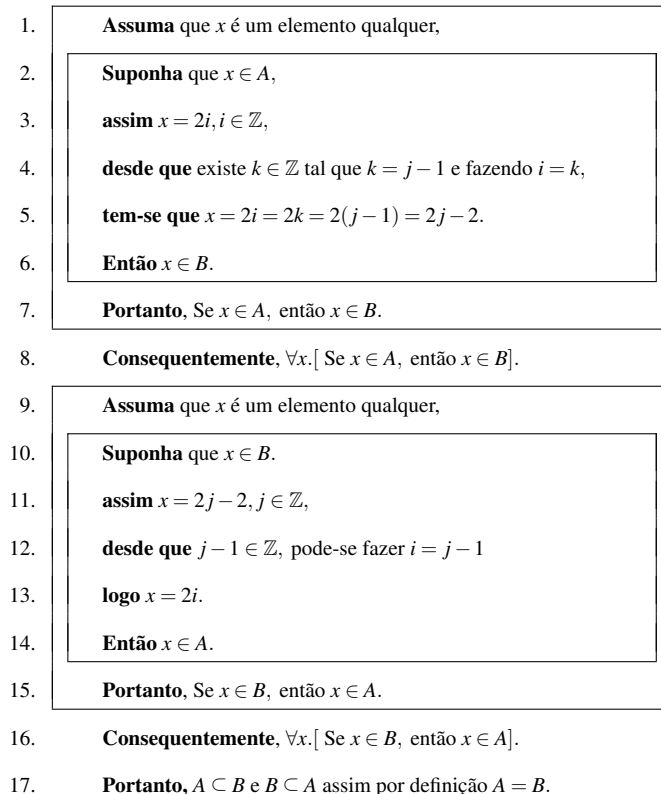
Sobre o diagrama de blocos é conveniente explicar sua estrutura, ele consiste de uma série de linhas numeradas de 1 até m , em cada linha está uma informação, sendo esta uma hipótese assumida como verdadeira ou deduzida a partir das informações anteriores a ela ou ainda um resultado (ou definição)

válido(a) conhecido(a). Um diagrama de bloco representa uma prova, porém, uma prova pode conter n subprovas. Cada **prova** é delimitada no diagrama por um **bloco**, assim se existe uma subprova p' em uma prova p , significa que o diagrama de bloco de p' é interno ao diagrama de bloco de p . Na linha abaixo de todo bloco sempre estará a conclusão que se queria demonstrar, isto é, abaixo de cada bloco está a asserção que tal bloco demonstra.

Cada linha no diagrama começa com algum termo reservado (em um sentido similar ao de palavra reservada de linguagem de programação [4, 25]) escrito em negrito¹, esses termos reservados tem três naturezas distintas: inicialização de bloco, ligação e conclusão de blocos. Tais palavras podem variar a depender do material sobre demonstrações que o leitor possa encontrar na literatura neste manuscrito serão usando os seguintes conjuntos de palavras:

- Termos de inicialização de bloco: **Suponha**, **Deixe ser**, **Assuma** e **Considere**;
- Termos de ligação: **mas**, **tem-se que**, **então**, **assim**, **logo**, **além disso**, **desde que** e **dessa forma**;
- Termos de conclusão de bloco: **Portanto**, **Dessa forma**, **Consequentemente** e **Logo por contra positiva**.

■ **Exemplo 2.1** Demonstre a asserção: Dado $A = \{x \in \mathbb{Z} \mid x = 2i, i \in \mathbb{Z}\}$ e $B = \{x \in \mathbb{Z} \mid x = 2j - 2, j \in \mathbb{Z}\}$ tem-se que $A = B$.



¹A escrita dos termos reservados em negrito em geral será usada para que o leitor consiga identificar o que é informação último da prova e o que é apenas um artifício textual para dá melhor entendimento a demonstração.

Neste momento o exemplo anterior serve apenas para esboçar a ideia de um diagrama de bloco para uma demonstração, note que fica evidente que a depender da situação alguns termos de ligação são melhores que outros, e o mesmo também é válido para os termos de inicialização e conclusão de bloco.

Aqui não será detalhando a aplicação dos métodos de demonstração usado na demonstração do Exemplo 2.1, mas nas próximas seções serão apresentados cada um dos métodos de demonstração, e seguida será gradativamente apresentados exemplos para esboçar ao leitor como é usado o diagrama de blocos e relação a cada método de demonstração..

Como o leitor pode ter notado pelo diagrama de bloco no Exemplo 2.1, é possível enxergar o diagrama como ambiente muito similar a ideia de um programa imperativo em uma linguagem de programação estruturada (como Pascal ou C), no sentido de que, uma demonstração pode ser visto como a combinação de diversos blocos, em que os blocos respeito uma hierarquia e podem está aninhados entre si, a hierarquia dos bloco é determinar por uma indentação².

2.2 Demonstrando Implicações

Este manuscrito irá iniciar a apresentação dos métodos de demonstração a partir das estratégias usadas para demonstrar a implicação, isto é, as estratégias usadas para provar asserção da forma: “se α , então β ”.

Definição 2.2 — Prova Direta (PD). Dado uma asserção da forma: “se α , então β ”. A metodologia de prova direta para tal asserção consiste em supor α como sendo verdade e a partir disto deduzir β .

Observação 2.3 Obviamente ao fazer a demonstração é necessário identificar quem são os componentes α e β da implicação.

Esta estratégia é provavelmente a técnicas mais famosa e usada dentre todos os métodos de demonstração que existem, um conhecedor de lógica pode notar facilmente que tal estratégia nada mais é do que a regra de dedução natural chamada de introdução da implicação[70].

No que diz respeito ao diagrama tal estratégia consistem em: (1) criar um bloco, e dentro deste bloco na primeira linha irá conter a afirmação de que α está sendo assumido com uma hipótese verdadeira; (2) nas próximas n linhas irão acontecer as deduções necessárias até que na linha $n + 2$ seja deduzido o β e o bloco é fechado e (3) na linha $n + 3$ será adicionada a conclusão do bloco. A seguir serão apresentados exemplos do uso do método de demonstração direto para implicações e seu uso junto com o diagrama de bloco.

²Indentação é um termo utilizando em código fonte de um programa, serve para ressaltar ou definir a estrutura do algoritmo. Na maioria das linguagens de programação, a indentação é empregada com o objetivo de ressaltar a estrutura do algoritmo, aumentando assim a legibilidade do código, porém a linguagem de programação em que a indentação é parte da própria da gramática da linguagem.

■ **Exemplo 2.2** Para demonstrar da asserção “Se x é ímpar, então $x^2 + x$ é par” deve-se usar o método de demonstração **PD**, assim a prova começa abrindo um bloco e inserido na primeira linha a hipótese de que o antecedente x é um número ímpar é verdadeira, ou seja, tem-se:

1.

Suponha que $x \in \mathbb{N}$,

- 2.

em seguida pode-se na linha 2 deduzir a forma de x , mudando o diagrama para:

1.

Suponha que $x \in \mathbb{N}$,

2.

logo $x = 2k + 1, k \in \mathbb{N}$,
--
- 3.

agora nas próximas duas linhas pode-se deduzir respectivamente as formas (ou valores) de x^2 e $x^2 + x$, assim o diagrama é atualizado para:

1.

Suponha que x é um número ímpar,

2.

logo $x = 2k + 1, k \in \mathbb{Z}$,
--
3.

assim $x^2 = 4k^2 + 4k + 1, k \in \mathbb{Z}$,
--
4.

dessa forma $x^2 + x = 2((2k^2 + 2k) + k + 1), k \in \mathbb{Z}$.

- 5.

agora note que $x^2 + x = 2((2k^2 + 2k) + k + 1)$ pode ser reescrito (por substituição) como $x^2 + x = 2j$ com $j = (2k^2 + 2k) + k + 1$, essa dedução é inserida na linha de número 5 atualizando o diagrama para:

1.

Suponha que x é um número ímpar,

2.

logo $x = 2k + 1, k \in \mathbb{Z}$,
--
3.

assim $x^2 = 4k^2 + 4k + 1, k \in \mathbb{Z}$,
--
4.

dessa forma $x^2 + x = 2((2k^2 + 2k) + k + 1), k \in \mathbb{Z}$.

5.

logo $x^2 + x = 2j$ com $j = (2k^2 + 2k) + k + 1, k \in \mathbb{Z}$.
--
- 6.

agora na sexta linha do diagrama pode-se então deduzir a partir da informação na linha de número 5 que $x^2 + x$ é um número par, assim o diagrama muda para a forma:

1.

Suponha que x é um número ímpar,

2.

logo $x = 2k + 1, k \in \mathbb{Z}$,
--
3.

assim $x^2 = 4k^2 + 4k + 1, k \in \mathbb{Z}$,
--
4.

dessa forma $x^2 + x = 2((2k^2 + 2k) + k + 1), k \in \mathbb{Z}$.

5.

logo $x^2 + x = 2j$ com $j = (2k^2 + 2k) + k + 1, k \in \mathbb{Z}$,
--
6.

então $x^2 + x$ por definição é um número par.

- 7.

note porém que a informação na deduzida na linha de número 6 é exatamente o consequente da implicação que se queria deduzir. Portanto, o objetivo interno ao bloco foi atingido, pode-se então fechar o bloco introduzindo abaixo dele a conclusão do bloco, ou seja, na linha de número 7 é escrito que o antecedente de fato implica no consequente, assim o diagrama fica da forma:

1. **Suponha** que x é um número ímpar,
2. **logo** $x = 2k + 1, k \in \mathbb{Z}$,
3. **assim** $x^2 = 4k^2 + 4k + 1, k \in \mathbb{Z}$,
4. **dessa forma** $x^2 + x = 2((2k^2 + 2k) + k + 1), k \in \mathbb{Z}$.
5. **logo** $x^2 + x = 2j$ com $j = (2k^2 + 2k) + k + 1, k \in \mathbb{Z}$,
6. **então** $x^2 + x$ por definição é um número par.
7. **Portanto**, Se x é ímpar, então $x^2 + x$ é par.

assim o objetivo a ser demonstrado foi atingido e, portanto, a prova está completa.



Nota 2.1 Na demonstração apresentada no Exemplo 2.2 as justificativas da evolução do diagrama fora apresentadas passo a passo e separadas do diagrama, isso foi adotado nesse primeiro exemplo para detalhar a evolução da demonstração ao leitor, entretanto, isso não é o padrão, o normal (que será adotado) é que a justificativa (caso necessário^a) da dedução de uma linha seja inserida a direita da informação deduzida, separada desta pelo símbolo —. A partir deste ponto até o fim deste capítulo será usado esta forma de escrita.

^aQuando a justificativa for trivial, ela não será inserida na prova.

Observação 2.4 Nas justificativas das provas as palavras definição, associatividade, comutatividade serão abreviadas para DEF, ASS, COM respectivamente

■ **Exemplo 2.3** Demonstração da asserção: Se n é múltiplo de 4, então também é múltiplo de 2.

- | | | |
|----|--|------------------------|
| 1. | Suponha que n é múltiplo de 4, | — Hipótese |
| 2. | logo $x = 4k, k \in \mathbb{Z}$, | — DEF de múltiplo de 4 |
| 3. | assim $x = (2 \cdot 2)k, k \in \mathbb{Z}$, | — Reescrita da linha 2 |
| 4. | dessa forma $x = 2(2k), k \in \mathbb{Z}$. | — ASS da multiplicação |
| 5. | logo $x = 2i$, com $i = 2k, k \in \mathbb{Z}$. | — Reescrita da linha 4 |
| 6. | então x é múltiplo de 2. | — DEF de múltiplo de 2 |
| 7. | Portanto , Se n é múltiplo de 4, então também é múltiplo de 2. — Conclusão da PD (linhas 1-7) | |

O leitor deve ter notado nos Exemplos 2.2 e 2.3 as demonstrações sempre iniciam das hipótese que estão sendo assumidas, isto é, os antecedentes das implicações, isso ocorrer por que nenhuma informação adicional (necessária) é apresentada como premissa, há caso entretanto, que as premissas são importantes para o desenvolvimento da prova, como será visto no próximo exemplo.

■ **Exemplo 2.4** Demonstração da asserção: Dado m inteiro maior ou igual que 5 e n um número ímpar positivo. Se n é ímpar, então $m + n$ é um par maior ou igual que 6.

- | | | |
|-----|--|---------------------------------|
| 1. | $m \geq 5, m \in \mathbb{Z}$ | — Premissa |
| 2. | $n = 2k + 1, k \in \mathbb{Z}^+$ | — Premissa |
| 3. | Suponha que m é ímpar, | — Hipótese |
| 4. | logo $m = 2j + 1, j \in \mathbb{Z}$, | — DEF de número ímpar |
| 5. | assim $m + n = 2(j + k + 1)$ | |
| 6. | desde que $m \geq 5$ tem-se que $j \geq 2$, | |
| 7. | mas como $n \in \mathbb{Z}_+$ tem-se que $k \geq 0$, | |
| 8. | assim $j + k + 1 \geq 3$, | — Direto das linhas 6 e 7 |
| 9. | logo $2(j + k + 1) \geq 6$ | |
| 10. | então $m + n \geq 6$. | — Reescrita da linha 9 |
| 11. | Portanto , Se n é ímpar, então $m + n$ é um par maior ou igual 6. | — Conclusão da PD (linhas 3-10) |

■ **Exemplo 2.5** Demonstração da asserção: Dado $m, n \in \mathbb{R}$ e $3m + 2n \leq 5$. Se $m > 1$, então $n < 1$.

- | | | |
|-----|--|---------------------------------|
| 1. | $m, n \in \mathbb{R}$ | — Premissa |
| 2. | $3m + 2n \leq 5$ | — Premissa |
| 3. | Suponha que $m > 1$, | — Hipótese |
| 4. | assim $3m > 3$, | |
| 5. | desde que $3m \leq 5 - 2n$, | |
| 6. | tem-se que $3 < 3m \leq 5 - 2n$ | — Direto das linhas 4 e 5 |
| 7. | logo $3 < 5 - 2n$, | |
| 8. | assim $3 + 2n < 5$, | |
| 9. | então $n < 1$. | |
| 10. | Portanto , Se $n > 1$, então $n < 1$. | — Conclusão da PD (linhas 3-10) |

Além do método de prova direta asserções que são implicações podem ser provadas por um segundo método, chamado método da contra positiva (ou contraposição). Como dito em [75], o método da contra positiva se baseia na equivalência semântica³ (ver Capítulo 7) da expressão “Se α , então β ” com a expressão “Se não β , então não α ”. Formalmente o método de demonstração por contra positiva é como se segue.

Definição 2.3 — Prova por Contra Positiva (PCP). Dado uma asserção da forma: “se α , então β ”. A metodologia de prova por contra positiva para tal asserção consiste em demonstrar usando PD a asserção “se não β , então não α ”, em seguida concluir (ou enunciar) que a veracidade de “se α , então β ” segue da veracidade de “se não β , então não α ”.

Observação 2.5 Como pode ser visto no Capítulo 6 a asserção não α representa a negação da asserção α .

³De forma mais rigorosa o que de fato sustenta o método de demonstração por contra positiva é a corretude e a completude da lógica de primeira ordem, e não simplesmente uma questão semântica.

Agora em termos do diagrama de blocos o método PCP apresenta o seguinte raciocínio de construção do diagrama: (1) abrir um bloco com a primeira linha em braco; (2) realizar em um bloco (interno) a demonstração de que “se não β , então não α ” e (3) após a conclusão deste segundo bloco, o primeiro bloco é fechado, e sua conclusão consiste na informação “se α , então β ” e a justificativa de tal informação é simplesmente a conclusão PCP das linhas $i-j$, onde $i-j$ diz respeito ao intervalo contendo as linhas do bloco e da conclusão da demonstração de “se não β , então não α ”.



Nota 2.2 Vale salientar que a linha em branco no início é apenas um fator estético adotado neste manuscrito, para tornar a leitura do diagrama da demonstração mais agradável. A depender da situação do diagrama seu uso pode ser desconsiderado’.

■ **Exemplo 2.6** Demonstração da asserção: Se $n! > (n+1)$, então $n > 2$.

1.		
2.	Suponha que $n \leq 2$,	— Hipótese
3.	assim $n = 0, n = 1$ ou $n = 2$	— Direto da linha 2
4.	logo $n! = 1$ ou $n! = 2$,	— Da linha 3 e da DEF de fatorial
5.	então $n! \leq (n+1)$ com $n \leq 2$.	— Direto das linhas 3 e 4
6.	Portanto , Se $n \leq 2$, então $n! \leq (n+1)$.	— Conclusão da PD (linhas 2-5)
7.	Logo por contra positiva , Se $n! > (n+1)$, então $n > 2$.	— Conclusão da PCP (linhas 2-6)

■ **Exemplo 2.7** Demonstração da asserção: Se $n \neq 0$, então $n + c \neq c$.

1.		
2.	Suponha que $n + c = c$,	— Hipótese
3.	assim $n + c - c = c - c$	
4.	logo , $n + 0 = 0$,	
5.	então $n = 0$.	
6.	Portanto , Se $n + c = c$, então $n = 0$.	— Conclusão da PD (linhas 2-5)
7.	Logo por contra positiva , Se $n \neq 0$, então $n + c \neq c$.	— Conclusão da PCP (linhas 2-6)

■ **Exemplo 2.8** Demonstração da asserção: Dado três números $x, y, z \in \mathbb{R}$ com $x > y$. Se $xz \leq yz$, então $z \leq 0$.

1.	$x, y, z \in \mathbb{R}$,	— Premissa
2.	$x > y$,	— Premissa
3.		
4.	Suponha que $z > 0$,	— Hipótese
5.	então $xz > yz$,	— Das linhas 2 e 4 e da monotonicidade da multiplicação em \mathbb{R}
6.	Portanto , Se $z > 0$, então $xz > yz$.	— Conclusão da PD (linhas 2-5)
7.	Logo por contra positiva , Se $xz \leq yz$, então $z \leq 0$.	— Conclusão da PCP (linhas 3-6)

■ **Exemplo 2.9** Demonstração da asserção: Se n^2 é par, então n é par.

- | | | |
|----|--|---------------------------------|
| 1. | | |
| 2. | Suponha que n não é par, | — Hipótese |
| 3. | logo $n = 2k + 1$ com $k \in \mathbb{Z}$, | — DEF de paridade |
| 4. | assim $n^2 = 4k^2 + 4k + 1$ com $k \in \mathbb{Z}$, | |
| 5. | dessa forma $n^2 = 2j + 1$ com $j = 2k^2 + 2k$, | — Reescrita da linha 4 |
| 6. | então n^2 não é par | — DEF de paridade |
| 7. | Portanto , Se n não é par, então n^2 não é par. | — Conclusão da PD (linhas 2-6) |
| 8. | Logo por contra positiva , Se n^2 é par, então n é par. | — Conclusão da PCP (linhas 2-6) |

2.3 Demonstração por Absurdo

O método de demonstração por redução ao absurdo⁴ (ou por contradição) tem por objetivo provar que a asserção α junto com as premissas (se houverem) é verdadeira a partir da prova de que a suposição de que a asserção “não α ” seja verdadeira junto das mesmas premissas (mencionadas anteriormente) gera um absurdo (ou contradição). O fato deste absurdo seja gerado, permite concluir que suposição de que a asserção “não α ” seja verdadeira é ridícula, ou seja, “não α ” tem que ser falsa e, portanto, a asserção α tem que ser verdadeira. Esse argumento é garantido pelos princípios da não contradição e do terceiro excluído, ambos mencionados na Seção 6.6 do Capítulo 6.

Definição 2.4 — Prova por Redução ao Absurdo (RAA). A metodologia para uma demonstração por redução ao absurdo de uma asserção α , consiste em supor que não α é uma hipótese verdadeira, então deduzir um absurdo (ou contradição). Em seguida concluir que dado que a partir de não α foi produzido um absurdo pode-se afirmar que α é verdadeiro.

Em termos do diagrama de blocos o método RAA consiste nos seguintes passos: (1) abrir um bloco cuja primeira linha é vazia; (2) abrir um bloco interno em que na primeira linha deste bloco o termo de inicialização do bloco (já listados anteriormente) é seguida da expressão “por absurdo” e da asserção não α ; (3) em seguida nas próximas n linhas irão acontecer as deduções necessárias até que na linha $n + 2$ seja deduzido o absurdo (ou uma contradição) e o bloco é fechado, inserido na linha $n + 3$ a informação de que “Se não α , então \perp ” e é fechado o bloco externo e (4) na linha $n + 4$ será adicionada a conclusão do bloco externo, contendo algo como “Portanto, α é verdadeiro”.

Observação 2.6 Como explicado na Parte III deste manuscrito, o símbolo \perp é usado para denotar o absurdo.

■ **Exemplo 2.10** Demonstração da asserção: $\sqrt{2} \notin \mathbb{Q}$.

⁴Reductio ad absurdum em latim.

1.		
2.	Assuma por absurdo que $\sqrt{2} \in \mathbb{Q}$,	— Hipótese
3.	logo existem $a, b \in \mathbb{Z}$ tal que $\sqrt{2} = \frac{a}{b}$ sendo $b \neq 0, mdc(a, b) = 1$	
4.	logo $a^2 = 2b^2$, ou seja, a^2 é par,	
5.	nessa forma $a = 2i$ com $i \in \mathbb{Z}$,	— Pela linha 4 e o Exemplo 2.9
6.	logo $b^2 = 2i^2$ com $i \in \mathbb{Z}$,	
7.	nessa forma $b = 2j$ com $j \in \mathbb{Z}$,	— Pela linha 6 e o Exemplo 2.9
8.	assim $mdc(a, b) \geq 2$,	— Direto das linhas 5 e 7
9.	mas $mdc(a, b) = 1$ e $mdc(a, b) \geq 2$ é um absurdo.	— Direto das linhas 3 e 8
10.	Portanto , Se $\sqrt{2} \in \mathbb{Q}$, então \perp .	— Conclusão da PD (linhas 2-10)
11.	Consequentemente , $\sqrt{2} \notin \mathbb{Q}$.	— Conclusão da RAA (linhas 2-11)



Nota 2.3 O termo *mdc* que aparece na demonstração no Exemplo 2.10 é a abreviação para o conceito de máximo divisor comum.

Observação 2.7 Note que internamente a prova por redução ao absurdo de uma asserção α , deve ser demonstrado a asserção: não $\alpha \Rightarrow \perp$.

■ **Exemplo 2.11** Demonstração da asserção: Não existe solução inteira positiva não nula para a equação diofantina⁵ $x^2 - y^2 = 1$.

1.		
2.	Assuma por absurdo que $\exists x, y \in \mathbb{Z}_+^*$ tal que $x^2 - y^2 = 1$,	— Hipótese
3.	logo $x, y \in \mathbb{Z}_+^*$ tem-se que $\min(x, y) = 1$ e tem-se que $(x - y)(x + y) = 1$,	
4.	assim $x - y = 1$ e $x + y = 1$ ou $x - y = -1$ e $x + y = -1$,	— Por $x, y \in \mathbb{Z}_+^*$
5.	mas se $x - y = 1$ e $x + y = 1$ pode-se assumir $x = 1$ e $y = 0$,	
6.	assim $\min(x, y) \neq 1$,	— Direto da linha 5
7.	mas se $x - y = -1$ e $x + y = -1$ pode-se assumir $x = -1$ e $y = 0$,	
8.	assim $\min(x, y) \neq 1$,	— Direto da linha 7
9.	mas $\min(x, y) = 1$ e $\min(x, y) \neq 1$ é um absurdo.	— Direto das linhas 3, 6 e 8.
10.	Portanto , Se $\exists x, y \in \mathbb{Z}_+^*$ tal que $x^2 - y^2 = 1$, então \perp .	— Conclusão da PD (linhas 2-10)
11.	Consequentemente , não $\exists x, y \in \mathbb{Z}_+^*$ tal que $x^2 - y^2 = 1$.	— Conclusão da RAA (linhas 2-10)

Observação 2.8 Equações diofantinas tem papel central para computação, assim vale mencionar aqui que um importante resultado sobre essas equações que possui forte impacto na lógica, computabilidade e teoria dos números foi demonstrado pela combinação dos trabalhos de Julia Robinson (1919–1985) e Yuri Matiyasevich(1947–.) [71]. Tal resultado é a prova do problema de número dez da famosa lista de Hilbert^a, de forma sucinta a prova do resultado diz que não existe um algoritmo (ou

⁵Equações diofantinas são equações polinomiais, que permite a duas ou mais variáveis assumirem apenas valores inteiros.

método) universal para determinar se uma equação diofantina tem raízes inteiras.

^aA lista de Hilbert é um lista inicialmente composta por 10 problemas e depois expandida para 23, que foi apresentada pelo matemático alemão David Hilbert (1862-1943) como uma forma de guia a atenção dos matemáticos no século XX.

■ **Exemplo 2.12** Demonstração da asserção: Não existe um programa P jogador de xadrez que sempre vença.

1.	
2.	Assuma por absurdo que P é um programa jogador de xadrez que sempre vence, — Hipótese
3.	logo pode-se instanciar duas execuções de P denotadas por P_1 e P_2 ,
4.	mas se P_1 joga contra P_2 e vence,
5.	tem-se que P não é um programa que sempre vence, — Direto da linha 4
6.	além disso se ocorre o contrário e P_2 joga contra P_1 e vence,
7.	tem-se que P não é um programa que sempre vence, — Direto da linha 6
8.	mas P perder é um absurdo, já que por hipótese P é um programa que sempre vence. — Direto das linhas 3, 5 e 7.
9.	Portanto , Se P é um programa de jogar xadrez que sempre vence, então \perp . — Conclusão da PD (linhas 2-8)
10.	Consequentemente , Não existe um programa P jogador de xadrez que sempre vença.. — Conclusão da RAA (linhas 2-9)



Nota 2.4 O Exemplo 2.12 apresenta a demonstração de uma clássica asserção diretamente ligada a computação, tal exemplo foi apresentado para mostrar ao leitor iniciante que argumentos válidos não necessariamente usam a notação matemática.

■ **Exemplo 2.13** Demonstração da asserção: Se $3n + 2$ é ímpar, então n é ímpar.

1.	
2.	Suponha por absurdo que $3n + 2$ é ímpar e n é par, — Hipótese
3.	logo $n = 2k, k \in \mathbb{Z}$, — DEF de paridade
4.	dessa forma $3n + 2 = 2(3k + 1), k \in \mathbb{Z}$,
5.	assim $3n + 2$ é par, — Da linha 4 e da DEF de paridade
6.	mas $3n + 2$ ser ímpar e $3n + 2$ ser par, é um absurdo. — Das linhas 2 e 5
7.	Portanto , se $3n + 2$ é ímpar e n é par, então \perp . — Conclusão da PD (linhas 2-7)
8.	Consequentemente , se $3n + 2$ é ímpar, então n é ímpar. — Conclusão da RAA (linhas 2-8)

2.4 Demonstrando Generalizações

Antes de falar sobre o método usado para demonstrar generalizações deve-se primeiro reforçar ao leitor o que são generalizações. Uma generalização é qualquer asserção que contenha em sua formação expressões das formas:

(a) Para todo _____.

(b) Para cada _____.

(c) Para qualquer _____.

■ **Exemplo 2.14** As seguintes asserções são generalizações.

- (a) Todos os cachorros tem quatro patas.
- (b) Todos os números inteiros possuem um inverso aditivo.
- (c) Todos os times de futebol pernambucanos são times brasileiros.

Nos termos da lógica uma asserção é uma generalização sempre que o quantificador universal é o quantificador mais externo a da asserção.

Observação 2.9 Neste texto sempre que possível será usado a escrita da lógica de primeira ordem nas asserções universais e existenciais, para detalhes ver Capítulos 6 e 8.

Agora que o leitor está a par do que é uma generalização, pode-se prosseguir o texto deste manuscrito apresentando formalmente o método de demonstração para generalizações.

Definição 2.5 — Prova de Generalizações (PG). Para provar uma asserção da forma, “ $(\forall x)[P(x)]$ ”, em que $P(x)$ é uma asserção acerca da variável x . Deve-se assumir que a variável x assume como valor um objeto qualquer no universo do discurso de que trata a generalização, em seguida, provar que a asserção $P(x)$ é verdadeira, usando as propriedades disponível de forma genérica para os objetos do universo do discurso.

Em termos do diagrama, a prova de uma generalização começa inserido na primeira linha de um bloco a informação de que x é um objeto genérico (ou qualquer) do discurso, em seguida deve ser provado $P(x)$ é verdadeiro, caso seja necessário deve ser aberto um novos blocos para as subprovas, após demonstrar que $P(x)$ é verdadeiro para um x genérico do discurso, o bloco externo (aberto para a prova da generalização) é fechado e pode-se apresentar a conclusão de que todo objeto x do discurso $P(x)$ é verdadeiro. Note que esse raciocínio de demonstração garante (com explicado em [108]) que a asserção P é universal sobre o universo do discurso, ou seja, garante a universalidade da asserção P .

■ **Exemplo 2.15** Demonstração da asserção: $(\forall x \in \{4n \mid n \in \mathbb{N}\})[x \text{ é par}]$.

1.	Assuma que x é um elemento qualquer de $\{4n \mid n \in \mathbb{N}\}$	— Hipótese
2.	dessa forma $x = 4n, n \in \mathbb{N}$,	— DEF do discurso
3.	logo $x = (2 \cdot 2)n, n \in \mathbb{N}$,	— Reescrita da linha 2
4.	dessa forma $x = 2(2n), n \in \mathbb{N}$,	— ASS da multiplicação
5.	assim $x = 2k, k \in \mathbb{N}$,	
6.	então x é par.	— DEF de paridade
7.	Portanto , quando $x \in \{4n \mid n \in \mathbb{N}\}$, tem-se então que x é par.	— Conclusão das linhas 2-6
8.	Consequentemente , $(\forall x \in \{4n \mid n \in \mathbb{N}\})[x \text{ é par}]$.	— Conclusão da PG (linhas 1-7)

■ **Exemplo 2.16** Demonstração da asserção: $(\forall X, Y \subseteq \mathbb{U})[\text{se } X \neq \emptyset, \text{ então } (X \cup Y) \neq \emptyset]$.

1.	Considere dois conjuntos quaisquer $X, Y \subseteq \mathbb{U}$	— Hipótese
2.	Suponha que $X \neq \emptyset$,	— Hipótese
3.	logo existe pelo menos um $x \in X$,	
4.	desde que $x \in X$ tem-se que $x \in (X \cup Y)$,	
5.	então $(X \cup Y) \neq \emptyset$.	
6.	Portanto , Se $X \neq \emptyset$, então $X \cup Y \neq \emptyset$.	— Conclusão das PD (linhas 2-5)
7.	Consequentemente , $(\forall X, Y \subseteq \mathbb{U})[\text{se } X \neq \emptyset, \text{ então } (X \cup Y) \neq \emptyset]$.	— Conclusão da PG (linhas 1-6)

Um erro que muitos iniciantes frequentemente cometem ao tentar provar enunciados de generalização é utilizar uma (ou mais) propriedade(s) de um elemento genérico x para provar $P(x)$, entretanto esta(s) propriedade(s) usada(s) não é (são) compartilhada(s) por todos os elementos de \mathbb{U} , isto é, apenas um subconjunto de \mathbb{U} apresenta a(s) propriedade(s) usadas, para mais detalhes sobre este tipo de erro podem ser consultados em [108].

■ **Exemplo 2.17** Demonstração da asserção: $(\forall n \in \mathbb{Z})[\text{se } n > 2, \text{ então } n^2 > n + n]$.

1.	Assuma que n é um número inteiro,	— Hipótese
2.	Suponha que $n > 2$,	— Hipótese
3.	logo $n \cdot n > 2n$,	— Monotonicidade da multiplicação em \mathbb{Z}
4.	então $n^2 > n + n$.	— Reescrita da linha 3
5.	Dessa forma , se $n > 2$, então $x^2 > n + n$.	— Conclusão das PD (linhas 2-4)
6.	Portanto , $(\forall n \in \mathbb{Z})[\text{se } n > 2, \text{ então } x^2 > n + n]$.	— Conclusão da PG (linhas 1-5)

■ **Exemplo 2.18** Demonstração da asserção: $(\forall n \in \mathbb{Z})[3(n^2 + 2n + 3) - 2n^2 \text{ é um quadrado perfeito}]$.

1.	Assuma que x é um número inteiro,	— Hipótese
2.	Desde que $3(n^2 + 2n + 3) - 2n^2 = 3n^2 + 6n + 9 - 2n^2$,	
3.	mas $3n^2 + 6n + 9 - 2n^2 = n^2 + 6n + 9$,	
4.	assim $3(n^2 + 2n + 3) - 2n^2 = n^2 + 6n + 9$	— Direto das linhas 2 e 3
5.	mas $n^2 + 6n + 9 = (n + 3)^2$,	
6.	logo $3(n^2 + 2n + 3) - 2n^2 = (n + 3)^2$,	
7.	Dessa forma , $3(n^2 + 2n + 3) - 2n^2$ é um quadrado perfeito.	— Conclusão das linhas 2-6
8.	Portanto , $(\forall n \in \mathbb{Z})[3(n^2 + 2n + 3) - 2n^2 \text{ é um quadrado perfeito}]$.	— Conclusão da PG (linhas 1-7)

2.5 Demonstrando Existência e Unicidade

Antes de falar sobre o método de demonstração existencial deve-se primeiro reforçar ao leitor o que é um enunciado existencial. Um enunciado de uma sentença do tipo existencial é qualquer asserção que inicia usando as expressões das forma seguir:

(a) Existe um(a) _____.

(b) Há um(a) _____.

Agora sobre a metodologia para demonstrar (provar) a existência de um objeto com um determinada propriedade, ou seja, provar que um certo objeto x satisfaz uma propriedade P , é especificada pela definição a seguir.

Definição 2.6 — Prova de existência (PE). Para provar uma asserção da forma “ $(\exists x)[P(x)]$ ”, em que $P(x)$ é uma asserção sobre a variável x . Deve-se exibir um elemento específico “ a ” pertencente ao universo do discurso, e mostrar que a asserção $P(x)$ é verdadeira quando x é instanciado como sendo exatamente o elemento a , ou seja, deve-se mostrar que $P(a)$ é verdadeira.

Em relação ao diagrama de bloco, uma demonstração de existência, isto é, uma prova de uma asserção $(\exists x)[P(x)]$, irá se comportar de forma muito semelhante a uma demonstração de generalidade, as únicas mudanças significativas é que tal método inicia seu bloco com a declaração de que será atribuído um objeto **específico** em vez de considerar a variável genérica a x , ou seja, é realizado uma instanciação de um elemento. Além disso, a conclusão do bloco externo deve ser exatamente a $(\exists x)[P(x)]$, ou seja, a conclusão deverá ser a asserção existencial.

■ **Exemplo 2.19** Demonstração da asserção: $(\exists m, n \in \mathbb{I})[m^n \in \mathbb{Q}]$.

1.	Deixe ser $a = \sqrt{2}$ e $b = \sqrt{2}$	— Instanciação existencial
2.	logo $a, b \in \mathbb{I}$,	— Pelo Exemplo 2.10
3.	Se $\sqrt{2}^{\sqrt{2}} \in \mathbb{Q}$,	
4.	então não há mais nada a ser demonstrado.	
5.	Consequentemente , se $\sqrt{2}^{\sqrt{2}} \in \mathbb{Q}$, então $a^b \in \mathbb{Q}$.	— PD das linhas 3-4
6.	Se $\sqrt{2}^{\sqrt{2}} \notin \mathbb{Q}$,	
7.	logo $\sqrt{2}^{\sqrt{2}} \in \mathbb{I}$,	
8.	assim fazendo $c = a^b$ tem-se que $c \in \mathbb{I}$,	
9.	então $c^b = (\sqrt{2}^{\sqrt{2}})^{\sqrt{2}} = 2$.	
10.	Consequentemente , se $\sqrt{2}^{\sqrt{2}} \notin \mathbb{Q}$, então $c^b \in \mathbb{Q}$ com $c \in \mathbb{I}, c = a^b$.	— PD das linhas 6-9
11.	Portanto , $(\exists m, n \in \mathbb{I})[m^n \in \mathbb{Q}]$.	— Conclusão da PE (linhas 1-10)

■ **Exemplo 2.20** Demonstração da asserção: $(\exists n \in \mathbb{N})[n = n^2]$.

1.	Deixe ser $n = 1$	— Instanciação existencial
2.	logo $n \cdot 1 = 1 \cdot 1$,	— Monotonicidade da multiplicação
3.	assim $n = 1^2$,	— Reescrita da linha 2
4.	logo $1 = 1^2$,	— Das linhas 1 e 3
5.	então $n = n^2$,	— Direto das linha 1, 3 e 4
6.	Portanto , $(\exists n \in \mathbb{N})[n = n^2]$.	— Conclusão da PE (linhas 1-5)

■ **Exemplo 2.21** Demonstração da asserção: $(\exists X \subseteq \mathbb{U})[(\forall Y \subseteq \mathbb{U})[X \cup Y = Y]]$.

1.	Deixe ser $X = \emptyset$,	— Instanciação existencial
2.	Assuma que $Y \subseteq \mathbb{U}$	— Hipótese
3.	logo $y \in Y$ tem-se que $y \in (X \cup Y)$	— DEF de união
4.	assim $Y \subseteq (X \cup Y)$,	
5.		
6.	Suponha por absurdo que $(X \cup Y) \not\subseteq Y$	— Hipótese
7.	assim tem-se que existe $z \in (X \cup Y)$ e $z \notin Y$,	
8.	dessa forma $z \in X$,	— Direto da linha 7
9.	desde que $X = \emptyset$ é um absurdo que $z \in X$,	— Da linha 1 e da DEF de conjunto vazio
10.	Portanto , se $(X \cup Y) \not\subseteq Y$, então \perp .	— Conclusão da PD (linhas 6-9)
11.	Consequentemente , $(X \cup Y) \subseteq Y$.	— Conclusão RAA (linhas 6-10)
12.	Dessa forma , $(X \cup Y) = Y$.	— Direto das linhas 4 e 11
13.	Portanto , $(\exists X \subseteq \mathbb{U})[(\forall Y \subseteq \mathbb{U})[X \cup Y = Y]]$.	— Conclusão da PE (linhas 1-12)

Observação 2.10 O leitor que leu com atenção o Capítulo 1, ou que tenha domínio sobre a teoria dos conjuntos sabe que $(\emptyset \cup X) = X$, para qualquer conjunto X , assim poderia escrever uma prova bem mais curta (fica como exercício) do que a demonstração mostrada no Exercício 2.21.

Agora vale ressaltar uma importante questão, a prova de existência não garante que um único elemento do discurso satisfaça uma determinada propriedade, note que no Exemplo 2.20 poderia ser substituído 1 pelo número natural 0 sem haver qualquer perca para a demonstração. De fato, o que a prova de existência garante é que **pelo menos um** elemento dentro do discurso satisfaz a propriedade que está sendo avaliada. Uma demonstração que garante que **um e apenas um** elemento em todo discurso satisfaz uma certa propriedade é chamada de demonstração de unicidade.

Antes de falar sobre o método de demonstração de unicidade deve-se primeiro reforçar ao leitor o que é um enunciado existencial de unicidade. Basicamente tal tipod e enunciado consiste de um enunciado de existência que adiciona os termos “único” ou “apenas um” na uma sentença do tipo existencial ficando da formas:

(a) Existe apenas um(a) _____.

(b) Há apenas um(a) _____.

ou ainda,

(a) Existe um(a) único(a) _____.

(b) Há um(a) único(a) _____.



Nota 2.5 Deste ponto em diante sempre que possível será substituído a escrita “se P , então Q ” pela notação da lógica simbólica $P \Rightarrow Q$.

Definição 2.7 — Prova de unicidade (PU). Uma prova de unicidade consiste em provar uma asserção da forma “ $(\exists x)[P(x) \wedge (\forall y)[P(y) \Rightarrow x = y]]$ ”, em que P é uma asserção sobre os elementos do discurso. Para tal primeiro deve-se demonstrar que a asserção “ $(\exists x)[P(x)]$ ” é verdadeira, e depois prova que a generalização $(\forall y)[P(y) \Rightarrow x = y]$ também é verdadeira.

Observação 2.11 Pela noção de contrapositiva pode-se substituir na Definição 2.7 a generalização $(\forall y)[P(y) \Rightarrow x = y]$ pela asserção da forma $(\forall y)[x \neq y \Rightarrow \neg P(y)]$.

Com respeito ao diagrama de blocos, uma demonstração de unicidade apresenta um diagrama similar ao de uma prova de existência, entretanto, internamente ao bloco da demonstração irá existir uma subprova para a asserção $(\forall y)[P(y) \Rightarrow x = y]$, sendo está subprova responsável por mostrar a unicidade. Por fim após fechar o bloco mais externo deve-se enunciar a conclusão.

■ **Exemplo 2.22** Demonstração da asserção: $(\exists!x \in \mathbb{N})[x + x = x \wedge (\forall y \in \mathbb{N})[y + y = y \Rightarrow x = y]]$.

1.	Deixe ser $x = 0$,	— Instanciação existencial
2.	logo $x + 0 = 0 + 0$,	
3.	assim $x + 0 = 0$,	
4.	 dessa forma $x + x = x$.	— Da linha 1 e da reescrita da linha 3
5.	Suponha que $y \in \mathbb{N}$,	— Hipótese
6.	Assuma que $y + y = y$,	— Hipótese
7.	logo $y = y - y$,	
8.	assim $y = 0$,	
9.	então $y = x$,	— Reescrita da linha 8
10.	Consequentemente , $y + y = y \Rightarrow x = y$.	— Conclusão da PD (linhas 6-9)
11.	Portanto , $(\forall y \in \mathbb{N})[y + y = y \Rightarrow x = y]$,	— Conclusão da PG (linhas 5-10)
12.	logo $x + x = x \wedge (\forall y \in \mathbb{N})[y + y = y \Rightarrow x = y]$.	— Direto das linhas 4 e 11
13.	Portanto , $(\exists!x \in \mathbb{N})[x + x = x \wedge (\forall y \in \mathbb{N})[y + y = y \Rightarrow x = y]]$.	— Conclusão da PU (linhas 2-13)

Observação 2.12 Obviamente uma aserção de unicidade pode muito bem ser escrita usando a abreviação $(\exists!x)[P(x)]$, para mais detalhes sobre isto veja o Capítulo 6 da Parte III deste manuscrito.

■ **Exemplo 2.23** Demonstração da asserção: $(\forall x \in \mathbb{Z})[(\exists!y \in \mathbb{Z})[x + y = 0]]$.

1.	Assuma que $x \in \mathbb{Z}$,	— Hipótese
2.	Deixe ser $y = -x$,	— Instanciação existencial
3.	logo $x + y = x + (-x)$,	
4.	mas $x + (-x) = 0$,	
5.	então $x + y = 0$.	
6.	$(\exists y \in \mathbb{Z})[x + y = 0]$.	— Conclusão da PE (linhas 2-5)
7.	Assuma que $z \in \mathbb{Z}$,	— Hipótese
8.		
9.	Suponha por absurdo que $x + z = 0$ e $z \neq y$,	— Hipótese
10.	desde que $x + y = 0$ tem-se que $x + z = x + y$,	
11.	mas assim $z = y$, o que contradiz a hipótese e, portanto, é um absurdo.	
12.	Consequentemente , se $x + z = 0$ e $z \neq -x$, então \perp .	— Conclusão da PD (linhas 8-10)
13.	Portanto , se $x + z = 0$, então $x = y$.	— Conclusão da RAA (linhas 8-11)
14.	Dessa forma $(\forall z \in \mathbb{Z})[x + z = 0 \Rightarrow z = y]$.	— Conclusão da PG (linhas 7-13)
15.	Portanto , $(\forall x \in \mathbb{Z})[(\exists! y \in \mathbb{Z})[x + y = 0]]$.	— Conclusão da PU (linhas 2-13)

2.6 Demonstração Guiada por Casos

Para realizar uma demonstração guiada por casos (ou simplesmente demonstração por casos) a estratégia empregada consiste em demonstrar cobrindo todos os casos possíveis que as premissas α_i em um enunciado podem assumir, formalmente esta metodologia de demonstração é definida como se segue.

Definição 2.8 — Prova por Casos (PPC). Uma prova por caso, consiste em provar um enunciado da forma: Se α_1 ou \dots ou α_n , então β . Para isso é realizado os seguintes passos:

- Supor α_1 (e apenas ela) verdadeira, e demonstrar β .
- ⋮
- Supor α_n (e apenas ela) verdadeira, e demonstrar β .

A justificativa da validade da metodologia da prova por casos é que um enunciado que tenha a forma $(\alpha_1 \vee \dots \vee \alpha_n) \Rightarrow \beta$ será verdadeiro quando a conjunção da forma $(\alpha_1 \Rightarrow \beta) \wedge \dots \wedge (\alpha_n \Rightarrow \beta)$ for verdadeira, e para isso deve-se provar a validade de $(\alpha_i \Rightarrow \beta)$ para todo $1 \leq i \leq n$. Dessa forma o leitor pode notar facilmente que uma prova por casos nada mais é do que provar uma série de n implicações (se for necessário releia a Seção 2.2).

Observação 2.13 A justificativa descrita acima pode ser facilmente constatada utilizando o conceito de equivalência lógica, para detalhes veja a Seção 7.8 que trata dos sistemas semânticos para da lógica proposicional.

Com respeito ao diagrama de blocos uma prova por casos consiste de um diagrama que possui em

seu interior n provas da forma $\alpha_i \Rightarrow \beta$ com $1 \leq i \leq n$, após todas as sub-provas serem apresentadas a última linha no diagrama mais externo irá expressar uma sentença da forma $(\alpha_1 \Rightarrow \beta) \wedge \cdots \wedge (\alpha_n \Rightarrow \beta)$, então o diagrama será fechado e será escrita a conclusão do diagrama. O exemplo a seguir ilustram esse procedimento.

■ **Exemplo 2.24** Demonstração da asserção: Dado $n \in \mathbb{N}$. Se $n \leq 2$, então $n! \leq n + 1$.

1.	$n \in \mathbb{N}$	— Premissa
2.	Assuma que $n = 0$,	— Hipótese
3.	desde que $0! = 1$	
4.	assim $0! \leq 1$	
5.	mas $1 = n + 1$	
6.	então $n! \leq n + 1$	
7.	Portanto , se $n = 0$, então $n! \leq n + 1$	— Conclusão da PD (linhas 2-5)
8.	Assuma que $n = 1$,	— Hipótese
9.	desde que $n! = 1$	
10.	assim $n! < 2$	
11.	mas $2 = n + 1$	
12.	então $n! < n + 1$	
13.	Portanto , se $n = 1$, então $n! < n + 1$	— Conclusão da PD (linhas 7-11)
14.	Assuma que $n = 2$,	— Hipótese
15.	desde que $n! = 2$	
16.	logo $n! < 3$	
17.	mas $3 = n + 1$	
18.	então $n! < n + 1$	
19.	Portanto , se $n = 2$, então $n! < n + 1$	— Conclusão da PD (linhas 7-11)
20.	Consequentemente , Dado $n \in \mathbb{N}$. Se $n \leq 2$, então $n! \leq n + 1$.	— Conclusão da PPC (linhas 1-19)

■ **Exemplo 2.25** Demonstração da asserção: Se $x \in \mathbb{Z}$, então x^2 tem a mesma paridade de x .

1.		
2.	Assuma que $x = 2i$ com $i \in \mathbb{Z}$,	— Hipótese
3.	logo $x^2 = 2(2i^2)$, com $i \in \mathbb{Z}$	
4.	então x é par	— DEF de paridade
5.	Portanto , se x é par, então x^2 é par	— Conclusão da PD (linhas 2-4)
6.	Assuma que $x = 2i + 1$ com $i \in \mathbb{Z}$,	— Hipótese
7.	logo $x^2 = 2(2i^2) + 1$, com $i \in \mathbb{Z}$	
8.	assim $x^2 = 2j + 1$, com $j = (2i^2)$, $i \in \mathbb{Z}$	— Reescrita
9.	então x é par	— DEF de paridade
10.	Portanto , se x é ímpar, então x^2 é par	— Conclusão da PD (linhas 6-9)
11.	Consequentemente , se $x \in \mathbb{Z}$, então x^2 tem a mesma paridade que x .	— Conclusão da PPC (linhas 1-10)

Observação 2.14 Note que os casos que guiam a prova do Exemplo 2.25 são o caso do x ser par e o caso do x ser ímpar.

2.7 Outras Formas de Representação de Provas

Durante este capítulo foram apresentadas diversas metodologias para se realizar demonstrações, e para representar as provas (demonstrações) usando tais metodologias foi empregado o uso de representação por diagrama de blocos. Este manuscrito utilizou-se dessa representação por ela ser mais amigável ao leitor iniciante na tarefa de provar teoremas.

Existem diversas outras formas de representar a demonstração de um teorema, por exemplo, o professor Thanos Tsouanas em seu livro [106], usa o conceito de tabuleiro do “jogo” da demonstração para representar as demonstrações. Por fim, vale destacar a representação das demonstrações por meio de texto formal, que consiste basicamente em descrever a prova usando um texto utilizando o máximo de formalismo matemático possível, o exemplo a seguir ilustra a representação em texto formal.

■ **Exemplo 2.26** A representação por texto formal da demonstração da asserção: “Se n é par, então n^2 é par”, pode ser da seguinte forma.

Demonstração. Suponha que n é par, logo $n = 2k$ para algum $k \in \mathbb{Z}$, dessa forma tem-se que $n^2 = n \cdot n = 2k \cdot 2k = 4k^2 = 2(2k^2)$, mas desde que a multiplicação e potenciação são fechadas em \mathbb{Z} tem-se que existe $r \in \mathbb{Z}$ tal que $r = 2k^2$ e, portanto, $n^2 = 2r$, consequentemente, n^2 é par. \square

A representação por texto formal é em geral a maneira utilizada de fato no meio acadêmico, para mais exemplos dessa representação veja [27, 28, 33, 76, 87] e com texto em inglês é sugerido a leitura de [89].

Observação 2.15 A partir deste ponto do manuscrito será adotado a escrita de demonstração em texto formal, ficando assim a representação por bloco “confinada” neste capítulo.

2.8 Demonstração de Suficiência e Necessidade

Escrever futuramente sobre as noções de provas de suficiência e de necessidade.

2.9 Refutação por Contraexemplo

Escrever futuramente sobre contraexemplos⁶...!

⁶A grafia contra-exemplos era usando no AO de 1945 e não é mais usada!

2.10 Questionário

■ **Exercício 2.1** Demonstre as seguintes asserções.

- (a). Dado $a, b \in \mathbb{R}$. Se $a < b < 0$, então $a^2 > b^2$.
- (b). Dado $a, b \in \mathbb{R}$. Se $0 < a < b$, então $\frac{1}{b} < \frac{1}{a}$.
- (c). Dado $a \in \mathbb{R}$. Se $a^3 > a$, então $a^5 > a$.
- (d). Sejam $(A - B) \subseteq (C \cap D)$ e $x \in A$. Se $x \notin D$, então $x \in B$.
- (e). Sejam $a, b \in \mathbb{R}$. Se $a < b$, então $\frac{a+b}{2} < b$.
- (f). Dado $x \in \mathbb{R}$ e $x \neq 0$. Se $\frac{\sqrt[3]{x}+5}{x^2+6} = \frac{1}{x}$, então $x \neq 8$.
- (g). Sendo $a, b, c, d \in \mathbb{R}$ com $0 < a < b$ e $d > 0$. Se $ac \geq bd$, então $c > d$.
- (h). Dado $x, y \in \mathbb{R}$ e $3x + 2y \leq 5$. Se $x > 1$, então $y < 1$.
- (i). Sejam $x, y \in \mathbb{R}$. Se $x^2 + y = -3$ e $2x - y = 2$, então $x = -1$.
- (j). Se $n \in \mathbb{Z}$ e $4 \leq n \leq 12$, então n é a soma de dois números primos.
- (k). Dado $n \in \mathbb{N}$. Se $n \leq 3$, então $n! \leq 2^n$.
- (l). Dado $n \in \mathbb{N}$. Se $2 \leq n \leq 4$, então $n^2 \geq 2^n$.
- (m). Se n é um inteiro par, então $n^2 - 1$ é ímpar.
- (n). Seja $n_0 \in \mathbb{N}$ e $n_1 = n_0 + 1$. Tem-se que $n_0 n_1$ é par.
- (o). Se $n \in \mathbb{Z}$, então $n^2 + n$ é par.
- (p). Se $n \in \mathbb{Z}$ e n é par, então n^2 é divisível por 4.
- (q). Para todo $n \in \mathbb{Z}$ o número $3(n^2 + 2n + 3) - 2n^2$ é um quadrado perfeito.
- (r). Dado $n \in \mathbb{Z}$. Se $x > 0$, então $x + 1 > 0$.
- (s). Se n é ímpar, então n é a diferença de dois quadrados.
- (t). Se $3n + 5 = 6k + 8$ com $k \in \mathbb{Z}$, então n é ímpar.
- (u). Se n é par, então $3n + 2 = 6k + 2$ com $k \in \mathbb{Z}$.
- (v). Se $x^2 + 2x - 3 = 0$, então $x \neq 2$.

- (w). Dado $n, n_0, n_1 \in \mathbb{Z}$. Se n_0 e n_1 são ambos divisíveis por n , então $n_0 + n_1$ é também divisível por n .
- (x). Dado $x, y \in \mathbb{Z}$. Se xy não é divisível por n tal que $n \in \mathbb{Z}$, então $x + y$ é divisível por n .
- (y). Dado $m, n, p \in \mathbb{Z}$. Se m é divisível por n e n é divisível por p , então m é divisível por p .
- (z). Se x é ímpar, então $x^2 - x$ é par.

■ **Exercício 2.2** Prove que se A e $(B - C)$ são disjuntos, então $(A \cap B) \subseteq C$.

■ **Exercício 2.3** Prove que se $A \subseteq (B - C)$, então A e C são disjuntos.

■ **Exercício 2.4** Dado $x \in \mathbb{R}$ prove que:

- (a). Se $x \neq 1$, então existe $y \in \mathbb{R}$ tal que $\frac{y+1}{y-2} = x$.
- (b). Se existe um $y \in \mathbb{R}$ tal que $\frac{y+1}{y-2} = x$, então $x \neq 1$.

■ **Exercício 2.5** Dado um conjunto A e \mathcal{G} uma família demonstre que:

- (a). Se $A \in \mathcal{G}$, então $A \subseteq \mathcal{G}_\cup$.
- (b). Se $A \in \mathcal{G}$, então $\mathcal{G}_\cap \subseteq A$.

■ **Exercício 2.6** Demonstre que: se B é um conjunto, \mathcal{G} uma família não vazia e para todo $A \in \mathcal{G}$ tem-se que $B \subseteq A$, então $B \subseteq \mathcal{G}_\cap$.

■ **Exercício 2.7** Demonstre que: se $\emptyset \in \mathcal{G}$, então $\mathcal{G}_\cap = \emptyset$.

■ **Exercício 2.8** Considere que \mathbb{P} e $\overline{\mathbb{P}}$ representam respectivamente o conjunto dos números inteiros pares e ímpares, assim demonstre as seguintes asserções.

- (a). Para todo $x, y \in \overline{\mathbb{P}}$ tem-se que $x - y \in \mathbb{P}$.
- (b). Para todo $x, y \in \mathbb{P}$ e todo $z \in \overline{\mathbb{P}}$ tem-se que $(x + y) + z \in \overline{\mathbb{P}}$.
- (c). A soma de três elementos consecutivos de $\overline{\mathbb{P}}$ é um número múltiplo de 3.

■ **Exercício 2.9** Prove que $\sqrt{3} \notin \mathbb{Q}$.

■ **Exercício 2.10** Demonstre que: para todo $n \in \mathbb{Z}$, se $5n$ é ímpar, então n é ímpar.

■ **Exercício 2.11** Demonstre que $x^2 = 4y + 3$ não tem solução inteira.

■ **Exercício 2.12** Prove que todo número primo maior que 3 é igual a $6k + 1$ ou igual a $6k - 1$.

■ **Exercício 2.13** Considerando o conjunto dos números inteiros demonstre as seguintes asserções.

- (a). Para todo x, y, z se x divide y e x divide z , então x divide $y + z$.
- (b). Para todo x, y, z se xy divide yz e $z \neq 0$, então x divide y .

■ **Exercício 2.14** Considerando o conjunto \mathbb{R} como universo do discurso demonstre as asserções a seguir:

- (a). $(\forall x)[(\exists! y)[x^2 y = x - y]]$.
- (b). $(\exists! x)[(\forall y)[xy + x - 4 = 4y]]$.
- (c). $(\forall x)[x \neq 0 \wedge x \neq 1 \Rightarrow (\exists! y)[\frac{y}{x} = y - x]]$.
- (d). $(\forall x)[x \neq 0 \Rightarrow (\exists! y)[(\forall z)[zy = \frac{z}{x}]]]$

■ **Exercício 2.15** Seja \mathbb{U} um conjunto qualquer, demonstre as seguintes asserções:

- (a). $(\exists! A \in \wp(\mathbb{U}))[(\forall B \in \wp(\mathbb{U}))[A \cup B = B]]$.
- (b). $(\exists! A \in \wp(\mathbb{U}))[(\forall B \in \wp(\mathbb{U}))[A \cap B = B]]$.

Capítulo 3

Relações

“O cliente pode ter um carro pintado com a cor que desejar, contanto que esta seja preto”.

Henry Ford

3.1 Noções Básicas de Relações

A ideia de relação é um conceito frequentemente utilizado, seja no cotidiano das pessoas, seja na matemática [11]. Uma subárea da matemática de extrema importância para a Ciência da Computação, especificamente na área de banco de dados, é a álgebra relacional, que de forma resumida é o estudo das relações entre objetos de um mesmo espaço (conjunto).

Como comentado em [35], no cotidiano do mundo “real” existem diversos tipos de relacionamentos entre as entidades, por exemplo, imagine que duas pessoas, um homem jovem e um(a) garotinho(a) compartilham um ancestral comum, tal como um avô, assim pode-se dizer que os dois apresentam uma relação de parentesco, ou ainda que existe uma relação familiar entre os dois.

No que diz respeito ao universo matemático a noção de relação entre os objetos é algo onipresente em todos os campos da matemática. Um exemplo clássico de relacionamento que se pode estabelecer entre dois números x e y , é a ideia de dobro, isto é, x e y apresentam um relacionamento de dobro entre si no caso de $y = 2x$ ou $x = 2y$.

Note que de forma subliminar os exemplos anteriores caracterizam as relações de parentesco e dobro através da associação de elementos que juntos apresentavam uma certa propriedade, e nesse sentido uma relação nada mais é do que um conjunto definido sobre uma certa propriedade entre elementos de um espaço. A formalização das relações como sendo um conjunto será construída nas próximas seções.

3.2 Pares Ordenados e Produto Cartesiano

Da mesma forma que [2], neste manuscrito será considerada a definição apresentada a seguir de par ordenado, sendo que tal definição foi apresentada pela primeira vez pelo grande matemático e lógico polonês Kazimierz Kuratowski (1896–1980).

Definição 3.1 — Par ordenado. Sejam x e y elementos em um universo do discurso. O par ordenado entre x e y , denotado por (x, y) , corresponde a seguinte igualdade.

$$(x, y) = \{x, \{x, y\}\}$$

Dado qualquer par ordenado (x, y) o elemento x é chamado de primeira componente do par ordenado, e o y é chamado de segunda componente do par ordenado. Além disso, como explicado em [64, 63] a propriedade fundamental dos pares ordenados diz que, dois pares ordenados (x_1, y_1) e (x_2, y_2) serão iguais¹, isto é, $(x_1, y_1) = (x_2, y_2)$ se, e somente se, $x_1 = x_2$ e $y_1 = y_2$.

Observação 3.1 O leitor deve ficar atento na distinção que existe entre o par ordenado (x, y) e o conjunto $\{x, y\}$. Apesar de ambos terem os mesmos elementos básicos, são objetos matemáticos distintos.

De posse do conceito de par ordenado é possível definir uma nova operação sobre conjuntos, tal operação recebe o nome de produto Cartesiano² e será de vital importância para em seguida apresentar as ideias ligadas ao conceito de relações.

Definição 3.2 — Produto Cartesiano. Sejam A e B dois conjuntos quaisquer. O produto Cartesiano de A e B , denotado por $A \times B$, corresponde ao conjunto de todos os pares ordenados onde a primeira componente é um elemento de A e a segunda componente é um elemento de B , em notação formal tem-se que:

$$A \times B = \{(x, y) \mid x \in A, y \in B\}$$

■ **Exemplo 3.1** Dado os seguintes dois conjuntos $\{a, b, c\}$ e $\{-1, 1\}$ tem-se os seguintes produtos Cartesianos:

$$(a) \quad \{a, b, c\} \times \{-1, 1\} = \{(a, 1), (a, -1), (b, -1), (b, 1), (c, -1), (c, 1)\}.$$

$$(b) \quad \{-1, 1\} \times \{a, b, c\} = \{(1, a), (1, b), (1, c), (-1, a), (-1, c), (-1, b)\}.$$

$$(c) \quad \{a, b, c\} \times \{a, b, c\} = \{(a, a), (a, b), (a, c), (b, a), (b, b), (b, c), (c, a), (c, b), (c, c)\}.$$

¹Este resultado segue da definição de igualdade de conjuntos (ver a Definição 1.9), provar tal igualdade é um exercício interessante ao leitor.

²O nome produto Cartesiano provém do matemático francês René Descartes (1596–1650), que foi o primeiro a estudar tal operação conjuntista [63].

$$(d) \{-1, 1\} \times \{-1, 1\} = \{(1, 1), (1, -1), (-1, 1), (-1, -1)\}.$$

Um caso particular do produto Cartesiano é o chamado Cartesiano quadrado apresentado a seguir.

Definição 3.3 — Cartesiano quadrado. Seja A um conjunto qualquer. O produto Cartesiano quadrado de A , denotado por $A \times A$, corresponde ao produto Cartesiano de A consigo mesmo, em notação formal tem-se que:

$$A \times A = \{(x, y) \mid x, y \in A\}$$

■ **Exemplo 3.2** Os itens (c) e (d) do Exemplo 3.1 são produtos Cartesianos quadrados.

Teorema 3.1 — Produto Cartesiano - absorção. Dado dois conjuntos A e B tem-se que, $A \times B = \emptyset$ se, e somente se, $A = \emptyset$ ou $B = \emptyset$.

Demonstração. (\Rightarrow) Por contrapositiva assumamos que $A \neq \emptyset$ e $B \neq \emptyset$, assim tem-se que existem $x \in A$ e $y \in B$, consequentemente, pela definição de produto cartesiano existe $(x, y) \in A \times B$, assim tem-se que, $A \times B \neq \emptyset$, e portanto, a afirmação: Se $A \times B = \emptyset$, então $A = \emptyset$ ou $B = \emptyset$ é verdadeira.

(\Leftarrow) Suponha que $A = \emptyset$ ou $B = \emptyset$, assim por vacuidade é claro que $A \times B = \emptyset$. □

Teorema 3.2 — Produto Cartesiano - igualdade. Dado dois conjuntos A e B tem-se que, $A \times B = B \times A$ se, e somente se, $A = \emptyset$ ou $B = \emptyset$ ou $A = B$.

Demonstração. A prova deste enunciado irá ficar como exercício ao leitor. □

O produto Cartesiano enquanto operação tem a propriedade de preservar a relação de inclusão à direita e à esquerda como pode ser visto a seguir.

Teorema 3.3 — Produto Cartesiano - monotonicidade à direita. Dado três conjuntos A, B e C tem-se que, $A \subset B$ se, e somente se, $A \times C \subset B \times C$.

Demonstração. (\Rightarrow) Suponha que $A \subset B$, logo por definição tem-se que todo $x \in A$ é tal que $x \in B$, e assim é óbvio que para todo $(x, y) \in A \times C$ tem-se que $(x, y) \in B \times C$, e portanto, pela definição de subconjunto tem-se que $A \times C \subseteq B \times C$, mas por hipótese tem-se que existe $x' \in B$ tal que $x' \notin A$, logo existe $(x', y) \in B \times C$ tal que $(x', y) \notin A \times C$, consequentemente, $A \times C \subset B \times C$.

(\Leftarrow) Assumamos que $A \times C \subset B \times C$, logo tem-se que para todo $(x, y) \in A \times C$ tem-se que $(x, y) \in B \times C$, mas note que por definição $(x, y) \in A \times C$ se, e somente se, $x \in A$ e de forma similar tem-se que $(x, y) \in B \times C$ se, e somente se, $x \in B$, dessa forma tem-se que $A \subset B$, além disso, por hipótese existe um $(x', y) \in B \times C$ tal que $(x', y) \notin A \times C$, portanto, é claro que existe $x' \in B$ tal que $x' \notin A$, consequentemente, $A \subset B$. □

Teorema 3.4 — Produto Cartesiano - monotonicidade à esquerda. Dado três conjuntos A, B e C tem-se que, $A \subset B$ se, e somente se, $C \times A \subset C \times B$.

Demonstração. Similar a demonstração do Teorema 3.3. □

O próximo resultado mostra que a operação de produto Cartesiano se distribui sobre as operações de união, interseção e diferença.

Teorema 3.5 — Leis de Distributividade do Cartesiano. Dado três conjuntos A, B e C tem-se que:

- (i) $A \times (B \cap C) = (A \times B) \cap (A \times C)$.
- (ii) $(A \cap B) \times C = (A \times C) \cap (B \times C)$.
- (iii) $A \times (B \cup C) = (A \times B) \cup (A \times C)$.
- (iv) $(A \cup B) \times C = (A \times C) \cup (B \times C)$.
- (v) $A \times (B - C) = (A \times B) - (A \times C)$.
- (vi) $(A - B) \times C = (A \times C) - (B \times C)$.
- (vii) $A \times (B \ominus C) = (A \times B) \ominus (A \times C)$.
- (vii) $(A \ominus B) \times C = (A \times C) \ominus (B \times C)$.

Demonstração. Sejam A, B e C conjuntos tem-se que:

(i)

$$\begin{aligned}
 A \times (B \cap C) &= \{(x, y) \mid x \in A, y \in (B \cap C)\} \\
 &= \{(x, y) \mid x \in (A \cap A), y \in (B \cap C)\} \\
 &= \{(x, y) \mid x \in A, x \in A, y \in B, y \in C\} \\
 &= \{(x, y) \mid x \in A, y \in B, x \in A, y \in C\} \\
 &= \{(x, y) \mid x \in A, y \in B\} \cap \{(x, y) \mid x \in A, y \in C\} \\
 &= (A \times B) \cap (A \times C)
 \end{aligned}$$

(ii) Similar ao item anterior.

(iii)

$$\begin{aligned}
A \times (B \cup C) &= \{(x, y) \mid x \in A, y \in (B \cup C)\} \\
&= \{(x, y) \mid x \in (A \cup A), y \in (B \cup C)\} \\
&= \{(x, y) \mid x \in A \text{ ou } x \in A, y \in B \text{ ou } y \in C\} \\
&= \{(x, y) \mid x \in A, y \in B \text{ ou } x \in A, y \in C\} \\
&= \{(x, y) \mid x \in A, y \in B\} \cup \{(x, y) \mid x \in A, y \in C\} \\
&= (A \times B) \cup (A \times C)
\end{aligned}$$

(iv) Similar ao item anterior.

(v)

$$\begin{aligned}
A \times (B - C) &= \{(x, y) \mid x \in A, y \in (B - C)\} \\
&= \{(x, y) \mid x \in A \cap A, y \in (B - C)\} \\
&= \{(x, y) \mid x \in A, x \in A, y \in B, y \notin C\} \\
&= \{(x, y) \mid (x, y) \in A \times B, (x, y) \notin (A \times C)\} \\
&= (A \times B) - (A \times C)
\end{aligned}$$

(vi) Similar ao item anterior.

(vii)

$$\begin{aligned}
A \times (B \ominus C) &\stackrel{\text{Cor. 1.1}}{=} A \times ((B \cup C) - (B \cap C)) \\
&\stackrel{\text{Teo. 3.5(v)}}{=} (A \times (B \cup C)) - (A \times (B \cap C)) \\
&\stackrel{\text{Teo. 3.5(iii)}}{=} ((A \times B) \cup (A \times C)) - (A \times (B \cap C)) \\
&\stackrel{\text{Teo. 3.5(i)}}{=} ((A \times B) \cup (A \times C)) - ((A \times B) \cap (A \times C)) \\
&\stackrel{\text{Cor. 1.1}}{=} (A \times B) \ominus (A \times C)
\end{aligned}$$

(viii) Similar ao item anterior.

□

O conceito do produto Cartesiano pode, como explicado em [63, 64], ser estendido a poder operar com mais de dois conjuntos, sendo essa extensão realizada de forma natural apenas aumentando um

número de componentes nos elementos do conjunto resultante ao conjunto do produto, ou seja, os elementos deixam de ser simples pares ordenados para serem tuplas ordenadas. A seguir este conceito é formalizado.

Definição 3.4 — Produto Cartesiano n -ário. Dado $n \geq 2$ e sejam A_1, A_2, \dots, A_n conjuntos quaisquer, o produto Cartesiano n -ário, denotado por $A_1 \times \dots \times A_n$, corresponde ao conjunto formado por todas as tuplas da forma (a_1, \dots, a_n) tal que para todo $1 \leq i \leq n$ tem-se que $a_i \in A_i$.

Em um produto Cartesiano n -ário da forma $A_1 \times \dots \times A_n$ cada A_i com $1 \leq i \leq n$ é chamado de i -ésimo fator do produto. Outra forma comum de denotar o produto Cartesiano n -ário muito encontrada na literatura é usando o símbolo do produtório, ou seja, $\prod_{i=1}^n A_i$.

■ **Exemplo 3.3** Dado os conjuntos $\{-1, 1\}$, $\{a, b\}$ e $\{0, 1\}$ tem-se os seguintes produtos Cartesianos n -ários:

$$\begin{aligned} \{-1, 1\} \times \{a, b\} \times \{0, 1\} &= \{(-1, a, 0), (-1, a, 1), (-1, b, 0), (-1, b, 1), \\ &\quad (1, a, 0), (1, a, 1), (1, b, 0), (1, b, 1)\} \end{aligned}$$

$$\begin{aligned} \{-1, 1\} \times \{-1, 1\} \times \{a, b\} \times \{a, b\} &= \{(-1, 1, a, a), (-1, 1, a, b), \\ &\quad (-1, 1, b, a), (-1, 1, b, b), \\ &\quad (-1, -1, a, a), (-1, -1, a, b), \\ &\quad (-1, -1, b, a), (-1, -1, b, b), \\ &\quad (1, -1, a, a), (1, -1, a, b), \\ &\quad (1, -1, b, a), (1, -1, b, b), \\ &\quad (1, 1, a, a), (1, 1, a, b), \\ &\quad (1, 1, b, a), (1, 1, b, b)\} \end{aligned}$$



Nota 3.1 — Açúcar Sintático. No caso de $A_i = A_j$ para todo $1 \leq i, j \leq n$ e $n \geq 2$ é comum usar um açúcar sintático^a (*syntactic sugar* em inglês) para representar o produto Cartesiano n -ário, em vez de usar, $A_1 \times \dots \times A_n$ ou mesmo $\prod_{i=1}^n A_i$, em geral é usado a notação A^n .

^aAçúcar sintático é uma expressão criada em 1964 por Peter J. Landin (1930-2009) em seus seminais trabalhos [55, 56]. De forma direta um açúcar sintático diz respeito a uma sintaxe dentro da linguagem formal que tem por finalidade tornar suas construções mais fáceis de serem lidas e expressas, ou seja, um açúcar sintático é uma ferramenta para tornar o uso da linguagem “mais doce” (ou amigável) para o uso dos seres humanos.

■ **Exemplo 3.4** Dado o conjunto $\{0, 1\}$ tem-se que

$$\begin{aligned} \{0, 1\}^5 = & \{(0, 0, 0, 0, 0), (0, 0, 0, 0, 1), (0, 0, 0, 1, 0), (0, 0, 0, 1, 1), (0, 0, 1, 0, 0), (0, 0, 1, 0, 1), \\ & (0, 0, 1, 1, 0), (0, 0, 1, 1, 1), (0, 1, 0, 0, 0), (0, 1, 0, 0, 1), (0, 1, 0, 1, 0), (0, 1, 0, 1, 1), \\ & (0, 1, 1, 0, 0), (0, 1, 1, 0, 1), (0, 1, 1, 1, 0), (0, 1, 1, 1, 1), (1, 0, 0, 0, 0), (1, 0, 0, 0, 1), \\ & (1, 0, 0, 1, 0), (1, 0, 0, 1, 1), (1, 0, 1, 0, 0), (1, 0, 1, 0, 1), (1, 0, 1, 1, 0), (1, 0, 1, 1, 1), \\ & (1, 1, 0, 0, 0), (1, 1, 0, 0, 1), (1, 1, 0, 1, 0), (1, 1, 0, 1, 1), (1, 1, 1, 0, 0), (1, 1, 1, 0, 1), \\ & (1, 1, 1, 1, 0), (1, 1, 1, 1, 1)\} \end{aligned}$$

■ **Exemplo 3.5** São produtos Cartesianos n -ários:

$$(a) \{a, b, c\}^2 = \{(a, a), (c, b), (a, c), (a, b), (c, c), (b, a), (b, b), (b, c), (c, a)\}.$$

$$(b) \{0, 1\}^2 = \{(1, 0), (1, 1), (0, 1), (0, 1)\}.$$

$$(c) \{1\}^9 = \{(1, 1, 1, 1, 1, 1, 1, 1, 1)\}.$$

Observação 3.2 Sempre que for possível durante este manuscrito será utilizado a notação açucarada do produto cartesiano n -ário.


Quando os conjuntos A_1, A_2, \dots, A_n são todos conjuntos finitos, uma estratégia muito utilizada para se obter e também representar o mecanismo de construção das tuplas (a_1, a_2, \dots, a_n) pertencentes ao produto Cartesiano n -ário $A_1 \times A_2 \times \dots \times A_n$ é usando a noção de diagrama de árvore [63, 64].

Observação 3.3 De forma contrária ao que acontecer nos diagramas de árvores na área de estrutura de dados [103], linguagens formais [15, 49, 61] e compiladores [4, 25], os diagramas de árvore na teoria dos conjuntos são construídos de forma horizontal no sentido da esquerda para à direita.

Em um diagrama de árvore o número de níveis na árvore é igual ao número de conjuntos envolvidos no Cartesiano mais 2, ou seja, para cada produto Cartesiano n -ário, o número de níveis na árvore que gera/representa tal cartesiano é igual a $n + 2$. O diagrama é construindo por níveis da seguinte forma: Seguido a sequência dos conjuntos no Cartesiano $A_1 \times \dots \times A_n$, para todo $1 \leq i \leq n$ cada nível i do diagrama de árvore vai ser preenchido pelos elementos do conjunto A_i , com o conjunto A_i sendo repetido exatamente 2^{i-1} vezes a cada nível i , no nível inicial da árvore (nível 0) é colocado o símbolo de inicio da árvore, neste manuscrito será usado o $*$ como símbolo inicial, e no último nível da árvore (ou nível EPC) estão os elementos do produto Cartesiano em si.

formalmente o conceito de relação binária como se segue.

Definição 3.5 — Relação binária. Seja A e B dois conjuntos, uma relação R de A em B é qualquer subconjunto de $A \times B$, isto é, $R \subseteq (A \times B)$.

 **Nota 3.2 — Açúcar sintático.** Dado R uma relação binária de A em B a sintaxe da teoria dos conjuntos e de pares ordenados permite que seja escrito que $(x, y) \in R$, entretanto, está escrita é geralmente substituída por $x R y$. E no caso de $(x, y) \notin R$ é escrito simplesmente $x \not R y$.

A semântica das palavras $x R y$ e $x \not R y$ podem ser interpretadas respectivamente como: “ x está R -relacionado (está relacionado por R) com y ” e “ x não está R -relacionado (não está relacionado por R) com y ”. Em algumas obras como [21], é possível ver a sintaxe $x \underline{R} y$ para designar que $(x, y) \in R$, neste manuscrito o autor irá optar sempre que possível pelo açúcar sintática descrito na Nota 3.2, e quando não for possível (ou conveniente) será usado a sintaxe padrão da teoria dos conjuntos e dos pares ordenados.

Observação 3.5 Quando $R \subseteq A \times A$ é dito simplesmente que R é uma relação sobre A , em vez de dizer que R é uma relação de A em A .

Este manuscrito irá continuar definido dois objetos fundamentais no estudo das relações binárias, que são respectivamente o domínio e a imagem.

Definição 3.6 — Domínio e Imagem. Seja R uma relação de A em B , o domínio de R , denotado por $Dom(R)$, corresponde ao conjunto de todos os elementos de A que são a primeira coordenada de $x R y$, ou seja,

$$Dom(R) = \{x \in A \mid x R y\}$$

e a imagem de R , denotada por $Ima(R)$, corresponde ao conjunto de todos os elementos de B que são a segunda coordenada de $x R y$, ou seja,

$$Ima(R) = \{y \in B \mid x R y\}$$

■ **Exemplo 3.7** Seja $R = \{(a, 1), (b, -1), (c, 1), (b, 1), (c, -1)\}$ uma relação tem-se que $Dom(R) = \{a, b, c\}$ e $Ima(R) = \{1, -1\}$.

■ **Exemplo 3.8** Dado a relação $Q = \{(x, y) \in \mathbb{N}^2 \mid x^2 = y\}$ tem-se que $Dom(Q) = \{x \in \mathbb{N} \mid (\exists y \in \mathbb{N})[\sqrt{y} = x]\}$ e $Ima(Q) = \{y \in \mathbb{N} \mid (\exists x \in \mathbb{N})[x^2 = y]\}$

■ **Exemplo 3.9** Uma relação binária R famosa é aquela usada para representar o conjunto das frações positivas, tal relação é definida como $F = \{(x, y) \mid x \in \mathbb{N}, y \in (\mathbb{N} - \{0\})\}$, note que a fração $\frac{1}{12}$ por exemplo corresponde ao elemento $1 F 12$.

Dada qualquer relação R sempre é possível obter uma nova relação a partir de R , essa nova relação

recebe o nome de relação inversa ou oposta.

Definição 3.7 — Relação inversa. Seja R uma relação. A relação inversa (ou oposta) de R , denotada por R^{-1} , corresponde ao seguinte conjunto:

$$R^{-1} = \{(y, x) \mid x R y\}$$

■ **Exemplo 3.10** Considere a relação R do Exemplo 3.7, tem-se que a relação inversa de R corresponde ao conjunto $R^{-1} = \{(1, a), (-1, b), (1, c), (-1, c), (1, b)\}$.

■ **Exemplo 3.11** Dado a relação $P = \{(a, b) \in \mathbb{N} \times \mathbb{N} \mid a = b^2\}$ tem-se a inversa de P é exatamente a relação $R = \{(b, a) \in \mathbb{N} \times \mathbb{N} \mid b = \sqrt{a}\}$, isto é, $R = P^{-1}$.

Observação 3.6 Um fato básico para qualquer relação R é que $(R^{-1})^{-1} = R$. Em outras palavras, tal igualdade descreve que a reversa de uma relação, vista como uma operação é sempre involutiva, assim como a negação e o complemento.

Proposição 3.1 Se $R \subseteq A \times B$, então $R^{-1} \subseteq B \times A$.

Demonstração. Suponha que $R \subseteq A \times B$, logo todo $(x, y) \in R$ é tal que $(y, x) \in R^{-1}$, mas de $R \subseteq A \times B$ tem-se que $(x, y) \in A \times B$, consequentemente, por definição $x \in A$ e $y \in B$ e, portanto, $(y, x) \in B \times A$, consequentemente, $R^{-1} \subseteq B \times A$. \square

Observação 3.7 É fácil ver que para qualquer Cartesiano $A \times B$ tem-se que $(A \times B)^{-1} = B \times A$.

O leitor atento pode notar que o resultado da Proposição 3.1 pode ser estendido para relacionar diretamente duas relações, e isso é feito como se segue.

Teorema 3.6 Se R e S são relações tais que $R \subseteq S$, então $R^{-1} \subseteq S^{-1}$.

Demonstração. Similar ao raciocínio da demonstração da Proposição 3.1. \square

Uma vez que relações são conjuntos pode-se falar sobre as operações sobre relações, aqui não serão tratadas as operações triviais de união, interseção, complemento e diferença. Para essas operações é recomendável que o leitor retorne para revisar o texto apresentado na Seção 1.2 que trata exatamente de tais operações.

Uma operação natural que surge para as relações é a noção de composição entre duas relações R_1 e R_2 , a ideia da composição é gerar uma terceira relação a partir das relações iniciais. A seguir este manuscrito apresenta formalmente o conceito de composição.

Definição 3.8 — Composição de relações. Seja R_1 uma relação de A em B e seja R_2 uma relação de B em C , a composição de R_1 e R_2 , denotada por $R_1 \bullet R_2$, corresponde ao seguinte conjunto:

$$R_1 \bullet R_2 = \{(x, z) \mid (\exists y \in B)[x R_1 y \text{ e } y R_2 z]\}$$

Proposição 3.2 Seja R_1 uma relação de A em B e seja R_2 uma relação de B em C , então tem-se que:

(i) $Dom(R_1 \bullet R_2) \subseteq Dom(R_1)$.

(ii) $Ima(R_1 \bullet R_2) \subseteq Ima(R_2)$.

Demonstração. Trivial pela própria Definição 3.8. □

■ **Exemplo 3.12** Sejam $R = A \times B$ e $Q = B \times C$ tem-se que $R \bullet Q = A \times C$.

■ **Exemplo 3.13** Sejam $R_1 = \{(a, b), (i, b), (o, c), (o, e)\}$ e $R_2 = \{(b, 1), (b, -1), (c, 3), (d, 4)\}$ tem-se então que a composição de R_1 e R_2 é exatamente igual a relação $R = \{(a, 1), (a, -1), (i, 1), (i, -1), (o, 3)\}$.

Teorema 3.7 — Monotonicidade da Composição de Relações. Seja R_1 e R_2 relações de A em B . Se $R_1 \subseteq R_2$, então para toda relação R_3 de B em C tem-se que $(R_1 \bullet R_3) \subseteq (R_2 \bullet R_3)$

Demonstração. Suponha que R_1 e R_2 são ambas relações de A em B e que $R_1 \subseteq R_2$, agora note que para qualquer relação R_3 de B em C tem-se por definição que $(x, z) \in (R_1 \bullet R_3)$ se, e somente se, $(\exists y \in B)[x R_1 y \text{ e } y R_3 z]$, mas uma vez que, $R_1 \subseteq R_2$ é claro que $(\exists y \in B)[x R_2 y \text{ e } y R_3 z]$, e assim $(x, z) \in (R_2 \bullet R_3)$, portanto, $(R_1 \bullet R_3) \subseteq (R_2 \bullet R_3)$, concluindo assim a prova. □

Corolário 3.1 Se R_1, R_2, S_1, S_2 são relações tais que $R_1 \subseteq R_2$ e $S_1 \subseteq S_2$, então $(R_1 \bullet S_1) \subseteq (R_2 \bullet S_2)$.

Demonstração. Suponha que R_1, R_2, S_1, S_2 são relações tais que $R_1 \subseteq R_2$ e $S_1 \subseteq S_2$, assim pelo Teorema 3.7 tem-se que $(R_1 \bullet S_1) \subseteq (R_2 \bullet S_1)$. Agora note que por definição $(x, z) \in (R_2 \bullet S_1)$ se, e somente se, $(\exists y \in Dom(S_1))[x R_2 y \text{ e } y S_1 z]$, mas uma vez que $S_1 \subseteq S_2$ tem-se que $Dom(S_1) \subseteq Dom(S_2)$ e $Ima(S_1) \subseteq Ima(S_2)$ e assim é claro que $(\exists y \in Dom(S_2))[x R_2 y \text{ e } y S_2 z]$, logo $(x, z) \in (R_2 \bullet S_2)$, consequentemente pela definição de subconjunto tem-se que $(R_2 \bullet S_1) \subseteq (R_2 \bullet S_2)$. E portanto, $(R_1 \bullet S_1) \subseteq (R_2 \bullet S_2)$. □

Os próximos resultados estabelece propriedades algébricas importantes para a operação de composição de relações.

Teorema 3.8 Seja R_1 uma relação de A em B e seja R_2 uma relação de B em C tem-se que $(R_1 \bullet R_2)^{-1} = R_2^{-1} \bullet R_1^{-1}$.

Demonstração. Dado R_1 uma relação de A em B e seja R_2 uma relação de B em C logo,

$$\begin{aligned}
 (x, z) \in (R_1 \bullet R_2)^{-1} &\iff (z, x) \in (R_1 \bullet R_2) \\
 &\stackrel{\text{Def. 3.8}}{\iff} (\exists y \in B)[(z, y) \in R_1 \text{ e } (y, x) \in R_2] \\
 &\iff (\exists y \in B)[(y, z) \in R_1^{-1} \text{ e } (x, y) \in R_2^{-1}] \\
 &\iff (\exists y \in B)[(x, y) \in R_2^{-1} \text{ e } (y, z) \in R_1^{-1}] \\
 &\iff (x, z) \in R_2^{-1} \bullet R_1^{-1}
 \end{aligned}$$

e assim pela Definição 1.9 tem-se que $(R_1 \bullet R_2)^{-1} = R_2^{-1} \bullet R_1^{-1}$ o que completa a prova. \square

Teorema 3.9 Seja R_1 uma relação de A em B e seja R_2 uma relação de B em C e R_3 uma relação de C em D tem-se que $(R_1 \bullet R_2) \bullet R_3 = R_1 \bullet (R_2 \bullet R_3)$.

Demonstração. Dado três relações R_1 de A em B , R_2 de B em C e R_3 de C em D tem-se por definição que, $(x, z) \in (R_1 \bullet R_2) \bullet R_3$ se, e somente se, existe $w \in C$ tal que $(x, w) \in (R_1 \bullet R_2)$ e $(w, z) \in R_3$, mas isso só é possível se, e somente se, $\exists y \in B$ tal que $(x, y) \in R_1$ e $(y, w) \in R_2$. Mas assim pela Definição 3.8 tem-se que $(y, z) \in R_2 \bullet R_3$, o que irá implicar que $(x, z) \in R_1 \bullet (R_2 \bullet R_3)$ e, portanto, tem-se que $(x, z) \in (R_1 \bullet R_2) \bullet R_3 \iff (x, z) \in R_1 \bullet (R_2 \bullet R_3)$, logo pela Definição 1.9 tem-se que $(R_1 \bullet R_2) \bullet R_3 = R_1 \bullet (R_2 \bullet R_3)$. \square

Teorema 3.10 Dado duas relações R_1 e R_2 e A, B e C conjuntos. Se $R_1 \subset A \times B$ e $R_2 \subset B \times C$, então $R_1 \bullet R_2 \subset A \times C$.

Demonstração. Suponha que $R_1 \subset A \times B$ e $R_2 \subset B \times C$ logo tem-se que se $(x, y) \in R_1 \bullet R_2$ logo por definição existe $z \in B$ tal que $(x, z) \in R_1$ e $(z, y) \in R_2$, mas assim é claro que $(x, z) \in A \times B$ e $(z, y) \in B \times C$ e, portanto, $(x, y) \in A \times C$. Consequentemente, $R_1 \bullet R_2 \subset A \times C$. \square

Teorema 3.11 Seja A, B e C conjuntos. Então tem-se que:

- (1) Se $A \cap B \neq \emptyset$, então $(A \times B) \bullet (A \times B) = A \times B$.
- (2) Se $A \cap B = \emptyset$, então $(A \times B) \bullet (A \times B) = \emptyset$.
- (3) Se $B \neq \emptyset$, então $(B \times C) \bullet (A \times B) = A \times C$.

Demonstração. Aqui será demonstrado só o fato (1) ficando o (2) e (3) como exercício ao leitor. Dado A, B e C conjuntos, assuma que $A \cap B \neq \emptyset$, agora note que para todo $(x, y) \in (A \times B) \bullet (A \times B)$ tem-se que pelo fato de A e B não serem disjuntos sempre existe um $\exists z \in A \cap B$ tal que $(x, z) \in (A \times B)$ e $(z, y) \in (A \times B)$, portanto, $(x, y) \in A \times B$, logo pela Definição 1.9 tem-se que $(A \times B) \bullet (A \times B) = A \times B$. \square

Teorema 3.12 Dado duas relações R_1 e R_2 tem-se que:

$$(1) (R_1 \cup R_2)^{-1} = R_1^{-1} \cup R_2^{-1}.$$

$$(2) (R_1 \cap R_2)^{-1} = R_1^{-1} \cap R_2^{-1}.$$

Demonstração. Sejam R_1 e R_2 duas relações logo,

(1) Note que $(x, y) \in (R_1 \cup R_2)^{-1} \iff (y, x) \in (R_1 \cup R_2) \iff (y, x) \in R_1 \text{ ou } (y, x) \in R_2 \iff (x, y) \in R_1^{-1} \text{ ou } (x, y) \in R_2^{-1} \iff (x, y) \in R_1^{-1} \cup R_2^{-1}$, logo pela Definição 1.9 tem-se que $(R_1 \cup R_2)^{-1} = R_1^{-1} \cup R_2^{-1}$.

(2) A demonstração é similar ao item anterior.

□

3.4 Tipos ou Propriedades das Relações Binárias

Deste ponto em diante todas as relações consideradas até o final deste capítulo serão relações binárias sobre um conjunto não vazio A genérico, ou seja, tem-se que se R for uma relação, então $R \subseteq A \times A$. Dito isto, agora serão apresentados os “tipos”, ou na visão de [2], as propriedades que as relações binárias sobre um conjunto podem ser possuir.

Definição 3.9 — Tipo Identidade. Uma relação R é dita ser uma relação de identidade (ou relação idêntica [2]) sempre que R é igual ao conjunto $\{(x, x) \mid x \in A\}$.

■ **Exemplo 3.14** Seja $A = \{1, 2, 3, 4\}$ a relação $M = \{(3, 3), (1, 1), (2, 2)\}$ é uma relação de identidade, já a relação $Q = \{(1, 1), (2, 2), (3, 4)\}$ não é uma relações de identidade.

■ **Exemplo 3.15** Dado o conjunto \mathbb{N} , a relação $R = \{(x, y) \in \mathbb{N}^2 \mid x - y = 0\}$ é uma relação de identidade, já a relação $S = \{(x, y) \in \mathbb{N}^2 \mid x - y > 0\}$ não é uma relação de identidade pois $(5, 4) \in S$.

Observação 3.8 Dado que a relação de identidade possui exatamente todos os pares da forma (x, x) , é comum chamar esta relação de identidade do conjunto A , ou simplesmente identidade de A , que costuma também ser denotado por Id_A .

Teorema 3.13 — Neutralidade da relação de identidade. Se R é uma relação sobre A , então as seguintes igualdade são verdadeiras:

$$(i) R \bullet Id_A = R.$$

$$(ii) Id_A \bullet R = R.$$

Demonstração. (i) Suponha que R é uma relação sobre A , assim tem-se que:

$$\begin{aligned}(x, y) \in R \bullet Id_A &\iff (\exists y \in A)[x R y \text{ e } y Id_A y] \\ &\iff (x, y) \in R\end{aligned}$$

Portanto, $R \bullet Id_A = R$. (ii) Similar a demonstração do item anterior. \square

Proposição 3.3 Se A é um conjunto não vazio, então $Id_A^{-1} = Id_A$.

Demonstração. Trivial pelas Definições 3.7 e 3.9. \square

Observação 3.9 É fácil notar que para qualquer que seja o conjunto não vazio A tem-se que a relação identidade Id_A é única.

Definição 3.10 — Tipo Reflexivo. Uma relação R é dita ser reflexiva quando para todo $x \in A$ tem-se que $x R x$.

Um leitor atento pode perceber que a relação identidade de um conjunto é sempre reflexiva, porém, o oposto não é verdadeiro como exposto no exemplo a seguir.

■ **Exemplo 3.16** Dado o conjunto $A = \{a, b, c\}$ tem-se que:

(a) $K = \{(a, a), (b, c), (b, b), (c, c), (a, c), (c, a)\}$ é uma relação reflexiva, mas não é a identidade do conjunto A .

(a) $M = \{(a, a), (b, b), (c, c)\}$ é uma relação reflexiva e é também a relação identidade do conjunto A .

Como dito em [2], uma relação R não será reflexiva quando existir pelo menos um $x \in A$ tal que $x \not R x$.

■ **Exemplo 3.17** Dado o conjunto $L = \{0, 0.5, 1\}$ tem-se que o conjunto Q formado pelos elementos $(0, 0)$, $(0, 0.5)$ e $(1, 1)$ não é uma relação reflexiva, pois $0.5 \not R 0.5$, ou seja, $(0.5, 0.5) \notin Q$.

O próximo resultado estabelece uma caracterização para as relações serem reflexivas, isto é, tal resultado apresenta as condições suficientes e necessárias para que uma relação seja reflexiva.

Teorema 3.14 — Caracterização das Relações Reflexivas. Uma relação R é reflexiva se, e somente se, $Id_A \subset R$.

Demonstração. (\Rightarrow) Suponha que R seja reflexiva, logo por definição para todo $x \in A$ tem-se que $x R x$, e portanto, pela Definição 3.9 é claro que $Id_A \subset R$.

(\Leftarrow) Assuma que $Id_A \subset R$, agora uma vez que para todo $x \in A$ tem-se que $(x, x) \in Id_A$, pela Definição 1.7 segue que $(x, x) \in R$, isto é, tem-se que $x R x$, e portanto, R é reflexiva. \square

Corolário 3.2 Uma relação R é reflexiva se, e somente se, R^{-1} é reflexiva.

Demonstração. A demonstração é simples e fica como exercício ao leitor. \square

Teorema 3.15 — Fecho Algébrico das Relações Reflexivas. Se R_1 e R_2 são relações reflexivas sobre o mesmo conjunto, então $R_1 \cup R_2$ e $R_1 \cap R_2$ são também relações reflexivas.

Demonstração. Assuma que R_1 e R_2 são relações reflexivas sobre um conjunto A , assim pelo Teorema 3.14 tem-se que $Id_A \subset R_1$ e $Id_A \subset R_2$, agora pelo Teorema 1.3 tem-se a seguinte relação de inclusão:

$$R_1 \subseteq R_1 \cup R_2$$

logo, tem-se que $Id_A \subset R_1 \subseteq R_1 \cup R_2$, consequentemente pelo Teorema 1.3 tem-se que $R_1 \cup R_2$ é uma relação reflexiva. Agora suponha por absurdo que $Id_A \not\subseteq (R_1 \cap R_2)$, logo existe $(x, x) \in Id_A$ tal que $(x, x) \notin (R_1 \cap R_2)$, consequentemente pela Definição 1.14 tem-se que $(x, x) \notin R_1$ e $(x, x) \notin R_2$, o que contradiz a hipótese de que R_1 e R_2 sejam relações reflexivas, isto é, contradiz a hipótese de $Id_A \subset R_1$ e $Id_A \subset R_2$, e portanto, $Id_A \subset (R_1 \cap R_2)$, logo pelo Teorema 3.14 tem-se que $R_1 \cap R_2$ é também uma relação reflexiva. \square

Teorema 3.16 Seja R_1 uma relação reflexiva sobre um conjunto A e seja R_2 um relação qualquer sobre o conjunto A , tem-se $R_1 \cup R_2$ é uma relação reflexiva.

Demonstração. A demonstração é trivial e ficará como exercício ao leitor. \square

Teorema 3.17 Se R é uma relação reflexiva, então $R \bullet R^{-1}$ e $R^{-1} \bullet R$ são também relações reflexivas.

Demonstração. Assuma que R é uma relação reflexiva sobre um conjunto A , assim pelo Corolário 3.2 tem-se que R^{-1} é uma relação reflexiva. Assim pelo Teorema 3.14 tem-se que $Id_A \subseteq R$ e $Id_A \subseteq R^{-1}$, consequentemente, pelo Corolário 3.1 tem-se que $(Id_A \bullet Id_A) \subseteq (R \bullet R^{-1})$ e $(Id_A \bullet Id_A) \subseteq (R^{-1} \bullet R)$, mas pela neutralidade da relação identidade (Teorema 3.13) tem-se que $Id_A \bullet Id_A = Id_A$, assim tem-se que $Id_A \subseteq (R \bullet R^{-1})$ e $Id_A \subseteq (R^{-1} \bullet R)$, e portanto, $R \bullet R^{-1}$ e $R^{-1} \bullet R$ são relações reflexivas. \square

Teorema 3.18 Se R é uma relação reflexiva, então as seguintes afirmações são verdadeiras.

- (i) $R \subset R \bullet R$.
- (ii) $R \bullet R$ é reflexiva.

Demonstração. A demonstração é simples e fica como exercício ao leitor. \square

Um terceiro tipo de relações binárias é o tipo irreflexivo, de um certo ponto de vista, tal tipo de relação pode ser visto como sendo o contraponto do tipo reflexivo.

Definição 3.11 — Tipo Irreflexivo. Uma relação R é dita ser irreflexiva quando para todo $x \in A$ tem-se que $x \not R x$.

■ **Exemplo 3.18** Seja P o conjunto de todas as pessoas, e seja R a relação “ser vó”, tem-se que R é irreflexiva pois é claro que ninguém pode ser vó de si próprio, portanto, para todo $x \in P$ tem-se que $x \not R x$.

■ **Exemplo 3.19** Seja $\mathbb{N}_1 = \{x \in \mathbb{N} \mid x > 0\}$ tem-se que a relação R definida sobre \mathbb{N}_1 como sendo $x \not R y \iff y = 2x$ é irreflexiva.

Seguindo com a tipagem das relações binárias, a seguir este manuscrito irá apresentar os tipos: simétrico, assimétrico e anti-simétricos.

Definição 3.12 — Tipo Simétrico. Uma relação R é dita ser simétrica quando para todo $x, y \in A$ se $x R y$, então $y R x$.

■ **Exemplo 3.20** Dado o conjunto $A = \{-3, -2, -1, 0, 1, 2, 3, 4\}$ o conjunto $\{(x, y) \in A^2 \mid x + y \geq 6\}$ é claramente uma relação simétrica sobre A .

■ **Exemplo 3.21** Sendo $B = \{1, 2, 3, 4\}$ o conjunto $\{(1, 1), (1, 3), (4, 2), (2, 4), (2, 2), (3, 1)\}$ é claramente uma relação simétrica sobre B .

Pela Definição 3.12 é fácil notar que uma relação R não será simétrica sempre que existir pelo menos um par (x, y) tal que $y R x$ mas $y \not R x$. O próximo resultado estabelece uma caracterização para as relações simétricas.

Teorema 3.19 — Caracterização das Relações Simétricas. Uma relação R será simétrica se, e somente se, $R = R^{-1}$.

Demonstração. (\Rightarrow) Suponha que R é simétrica, logo

$$\begin{aligned} (x, y) \in R &\iff (y, x) \in R \\ &\iff (x, y) \in R^{-1} \end{aligned}$$

portanto, pela Definição 1.9 tem-se que $R = R^{-1}$. (\Leftarrow) É trivial e fica como exercício ao leitor. □

Corolário 3.3 Se R é simétrica, então $R \bullet R^{-1} = R^{-1} \bullet R$.

Demonstração. Direto do Teorema 3.19. □

Agora será mostrado que união e interseção são operações fechadas sobre o conjunto de todas as relações binárias simétricas.

Teorema 3.20 Se R e S são relações simétricas, então $R \cup S$ e $R \cap S$ também são simétricas.

Demonstração. Trivial. □

Teorema 3.21 Se R é uma relação qualquer, então $R \bullet R^{-1}$ e $R^{-1} \bullet R$ são ambas simétricas.

Demonstração. Suponha que R é uma relação, assim tem-se que

$$\begin{aligned} (R \bullet R^{-1})^{-1} &\stackrel{\text{Teo.3.8}}{=} (R^{-1})^{-1} \bullet R^{-1} \\ &\stackrel{\text{Obs.3.6}}{=} R \bullet R^{-1} \end{aligned}$$

e

$$\begin{aligned} (R^{-1} \bullet R)^{-1} &\stackrel{\text{Teo.3.8}}{=} R^{-1} \bullet (R^{-1})^{-1} \\ &\stackrel{\text{Obs.3.6}}{=} R^{-1} \bullet R \end{aligned}$$

assim pelo Teorema 3.19 tem-se que $R \bullet R^{-1}$ e $R^{-1} \bullet R$ são ambas simétricas. □

Teorema 3.22 Se R é uma relação qualquer, então $R \cup R^{-1}$ e $R \cap R^{-1}$ são ambas simétricas.

Demonstração. Suponha que R é uma relação, assim tem-se que

$$\begin{aligned} (R \cup R^{-1})^{-1} &\stackrel{\text{Teo.3.12}}{=} R^{-1} \cup (R^{-1})^{-1} \\ &= (R^{-1})^{-1} \cup R^{-1} \\ &\stackrel{\text{Obs.3.6}}{=} R \cup R^{-1} \end{aligned}$$

e

$$\begin{aligned} (R \cap R^{-1})^{-1} &\stackrel{\text{Teo.3.12}}{=} R^{-1} \cap (R^{-1})^{-1} \\ &= (R^{-1})^{-1} \cap R^{-1} \\ &\stackrel{\text{Obs.3.6}}{=} R \cap R^{-1} \end{aligned}$$

assim pelo Teorema 3.19 tem-se que $R \cup R^{-1}$ e $R \cap R^{-1}$ são ambas simétricas. □

Definição 3.13 — Tipo Assimétrico. Uma relação R é dita ser assimétrica quando para todo $x, y \in A$ se $x R y$, então $y \not R x$.

■ **Exemplo 3.22** Considere que P é a relação de paternidade definida sobre o conjunto dos seres humanos, isto é, $x P y$ significa que x é pai de y , obviamente esta relação é assimétrica pois dado que um indivíduo x é pai de um certo y é impossível que y seja pai de x , ou seja, sempre que $x R y$ será verdade que $y \not R x$.

■ **Exemplo 3.23** A relação $R = \{(x, y) \in \mathbb{N} \mid x - y \leq 0\}$ é uma relação assimétrica

O leitor deve ficar atento ao fato de que uma relação R será dita não ser assimétrica se existir pelo menos um par (x, y) tal que $x R y$ e também que $x R y$.

■ **Exemplo 3.24** Considere $K = \{1, 2, 3, 4\}$ e T a relação binária definida sobre o conjunto K tal que $T = \{(1, 2), (1, 3), (4, 1), (1, 4), (2, 3)\}$. Tem-se claramente que T não é assimétrica pois $4 T 1$ e $1 T 4$.

Teorema 3.23 Se R é uma relação assimétrica sobre A , então R é uma relação irreflexiva sobre A .

Demonstração. Suponha por absurdo que R é uma relação assimétrica sobre A e R não é irreflexiva sobre A , logo por R não ser irreflexiva existe $x \in A$ tal que $x R x$, mas isso não satisfaz a Definição 3.13 e, portanto, isso contradiz a hipótese de que R é uma relação assimétrica sobre A , consequentemente se R é assimétrica, então R tem que ser irreflexiva. \square

Definição 3.14 — Tipo Anti-simétrico. Uma relação R é dita ser anti-simétrica quando para todo $x, y \in A$ se $x R y$ e $y R x$, então $x = y$.

■ **Exemplo 3.25** Considerando $A = \{1, 2, 3, 4\}$ e $R = \{(1, 1), (2, 3), (4, 4), (4, 3)\}$ tem-se que R é claramente anti-simétrica.

■ **Exemplo 3.26** Dado um conjunto A qualquer a relação de subconjunto \subseteq sobre $\wp(A)$ é uma relação que é anti-simétrica, pois para todo $A, B \in \wp(A)$ quando $A \subseteq B$ e $B \subseteq A$ tem-se por definição que $A = B$.

O leitor deve ter notado que uma relação R sobre um conjunto A não será anti-simétrica se existir pelo menos $x, y \in A$ tais que $x R y$ e $y R x$, mas $x \neq y$.

■ **Exemplo 3.27** Considere que $A = \{1, 2, 3, 4\}$ e $R = (1, 1), (3, 2), (2, 3), (3, 4)$ obviamente R não é anti-simétrica pois $3 R 2$ e $2 R 3$ mas claramente 2 e 3 são elementos distintos de A .

Teorema 3.24 — Caracterização das Relações Anti-simétricas. Uma relação R é anti-simétrica sobre A se, e somente se, $R \cap R^{-1} \subset Id_A$.

Demonstração. (\Rightarrow) Suponha por absurdo que R é anti-simétrica sobre A e que $R \cap R^{-1} \not\subset Id_A$, logo existe $(x, y) \in R \cap R^{-1}$ tal que $(x, y) \notin Id_A$, mas pelo fato de que $(x, y) \in R \cap R^{-1}$ tem-se que $(x, y) \in R$ e $(x, y) \in R^{-1}$ e assim $(y, x) \in R$ e como R é anti-simétrica tem-se que $x = y$, logo $(x, y) \in Id_A$ o que é um absurdo, portanto, se R é anti-simétrica sobre A , então tem-se que $R \cap R^{-1} \subset Id_A$. (\Leftarrow) Suponha

que $R \cap R^{-1} \subset Id_A$, assim seja $x, y \in A$ tal que $x R y$ e $y R x$, ou seja, $(x, y) \in R$ e $(x, y) \in R^{-1}$, logo $(x, y) \in R \cap R^{-1}$ e assim tem-se que $x = y$ e assim R é anti-simétrica. \square

Corolário 3.4 Uma relação R é anti-simétrica se, e somente se, R^{-1} for anti-simétrica.

Demonstração. Note que,

$$\begin{array}{ll}
 R \text{ é uma relação anti-simétrica} & \xLeftrightarrow{\text{Teo.3.24}} R \cap R^{-1} \subset Id_A \\
 & \xLeftrightarrow{\text{Teo.3.6}} (R \cap R^{-1})^{-1} \subset Id_A^{-1} \\
 & \xLeftrightarrow{\text{Prop.3.3}} (R \cap R^{-1})^{-1} \subset Id_A \\
 & \xLeftrightarrow{\text{Teo.3.12(2)}} R^{-1} \cap (R^{-1})^{-1} \subset Id_A \\
 & \xLeftrightarrow{\text{Teo.3.24}} R^{-1} \text{ é uma relação anti-simétrica}
 \end{array}$$

E isto conclui a prova. \square

Teorema 3.25 Se R e S são relações anti-simétricas, então $R \cap S$ também é anti-simétrica.

Demonstração. Suponha que R e S são relações anti-simétricas, logo pela Teorema 3.24 tem-se que $R \cap R^{-1} \subset Id_A$ e $S \cap S^{-1} \subset Id_A$, agora note que,

$$\begin{aligned}
 (R \cap S) \cap (R \cap S)^{-1} & \xLeftrightarrow{\text{Teo.3.12(2)}} (R \cap S) \cap (R^{-1} \cap S^{-1}) \\
 & \xLeftrightarrow{\text{Tab.1.1(p2) e (p3)}} (R \cap R^{-1}) \cap (S \cap S^{-1}) \\
 & \xLeftrightarrow{\text{Hip.}} Id_A \cap Id_A \\
 & = Id_A
 \end{aligned}$$

Portanto, $(R \cap S) \cap (R \cap S)^{-1} \subset Id_A$ e assim pelo Teorema 3.24 tem-se que $R \cap S$ é anti-simétrica. \square

Continuando a apresentação dos tipos (propriedades) das relações binárias, agora será introduzida o tipo transitivo.

Definição 3.15 — Tipo Transitivo. Uma relação R é dita ser transitiva quando para todo $x, y, z \in A$ se $x R y$ e $y R z$, então $x R z$.

■ **Exemplo 3.28** Dado um conjunto não vazio A a relação \subseteq definida sobre $\mathcal{P}(A)$ é um clássico exemplo de relação transitiva.

■ **Exemplo 3.29** A relação $R = \{(x, y) \in \mathbb{R}^2 \mid (\exists k \in \mathbb{R})[x = ky]\}$ é uma relação transitiva³.

³A prova disso ficará como exercício ao leitor.a prova disso ficará como exercício ao leitor

■ **Exemplo 3.30** A relação “ser ancestral de”, definida sobre o conjunto de todos os seres humanos (vivos e mortos) é uma relação transitiva.

Como muito explicado em [2] uma relação não será transitiva sempre que existirem $x, y, z \in A$ tais que $x R y$ e $y R z$ mas $x \not R z$.

■ **Exemplo 3.31** Seja $P = \{1, 2, 3, 4\}$ a relação $R_1 = \{(1, 1), (1, 2), (2, 4), (3, 2), (1, 4)\}$ é transitiva, já a relação $R_2 = \{(1, 1), (3, 1), (1, 2), (2, 3), (2, 4), (3, 3), (4, 1)\}$ não é transitiva pois $(3, 1), (1, 2) \in R_2$ mas $(3, 2) \notin R_2$.

Teorema 3.26 — Caracterização das Relações Transitivas. Uma relação R é transitiva sobre A se, e somente se, $R \bullet R \subset R$.

Demonstração. (\Rightarrow) Suponha que R seja transitiva, assim para todo $(x, y), (y, z) \in R$ é tal que $(x, z) \in R$, mas note que $(x, y), (y, z) \in R$ implica que $(x, z) \in R \bullet R$ e, portanto, $R \bullet R \subseteq R$. (\Leftarrow) Assuma que $R \bullet R \subset R$, logo todo $(x, z) \in R \bullet R$ é tal que $(x, z) \in R$, mas note que $(x, z) \in R \bullet R$ implica que existe $y \in A$ tal que $(x, y), (y, z) \in R$ e assim por definição R é transitiva. \square

Corolário 3.5 Uma relação R é transitiva se, e somente se, R^{-1} é também transitiva.

Demonstração. Note que,

$$\begin{aligned}
 R \text{ é uma relação transitiva} & \stackrel{\text{Teo. 3.26}}{\iff} R \bullet R \subset R \\
 & \stackrel{\text{Obs. 3.7}}{\iff} (R \bullet R)^{-1} \subset R^{-1} \\
 & \stackrel{\text{Teo. 3.8}}{\iff} R^{-1} \bullet R^{-1} \subset R^{-1} \\
 & \stackrel{\text{Teo. 3.26}}{\iff} R^{-1} \text{ é uma relação transitiva}
 \end{aligned}$$

E isto conclui a prova. \square

Teorema 3.27 Se R é transitiva, então $R \bullet R$ é transitiva.

Demonstração. Direto do Teorema 3.26. \square

Por fim será agora apresentado o último tipo das relação binárias, sendo este último tipo a contra parte do tipo transitivo.

Definição 3.16 — Tipo Intransitivo. Uma relação R é dita ser intransitiva quando para todo $x, y, z \in A$ se $x R y$ e $y R z$, então $x \not R z$.

■ **Exemplo 3.32** A relação “ x é mãe de y ” definida sobre o conjunto de todas as pessoas (vivas e mortas) é uma relação intransitiva, pois se “Maria é mãe de Julia” e “Julia é mãe de Rebeca” tem-se que “Maria não pode ser mãe de Rebeca”.

O leitor atento pode notar que uma relação R sobre um conjunto não vazio A qualquer, será dita não ser intransitiva quando existir pelo menos três elementos $x, y, z \in A$ tal que $x R y$ e $y R z$ e $x R z$.

3.5 Fecho das Relações Binárias

Como explicado no Capítulo 1 um conjunto é definido por uma propriedade, assim dado que uma relação é também um conjunto é natural que ela seja descrita por uma ou mais propriedades, e como dito em [40], as vezes dado uma relação R pode-se questionar “o que seria necessário para que R também tenha a propriedade P ?”, a resposta para essa pergunta é clara, **nada pode ser feito**, pois uma vez que, R não possui a propriedade P se for adicionado uma restrição sobre o elementos de R para que eles também passem a satisfazer P significa que foi na verdade criada uma nova relação R' , essa ação costuma ser chamada de fechamento da relação e essa nova relação R' é chamada de fecho de R definido formalmente a seguir.

Definição 3.17 — Fecho de uma relação. Se R uma relação sobre A e P uma propriedade, a relação \hat{R} definida sobre A é dita ser o fecho de R com respeito a P sempre que:

1. \hat{R} possui a propriedade P .
2. $R \subseteq \hat{R}$.
3. Para toda relação R^* que satisfaça as condições (1) e (2) tem-se que $\hat{R} \subseteq R^*$.

Observação 3.10 Em capítulos futuros a noção de fecho será visto como um operador, isto é, como uma função.

Para fins de interesse de estudantes de computação e também para fins didáticos neste manuscrito serão estudados os métodos de construção dos fechos para propriedades específicas, a saber, as propriedades de reflexividade, transitividade e simetria.

Um ponto importante a se destacar ao leitor iniciante é que se R já possui uma certa propriedade P , o fecho de R com respeito a P é exatamente o próprio R .

Definição 3.18 Seja R uma relação binária sobre A o fecho reflexivo de R , denotado por $ref(R)$, corresponde a seguinte relação:

$$ref(R) = R \cup Id_A$$

■ **Exemplo 3.33** Seja $A = \{a, b, c\}$ e $R = \{(a, a), (a, b), (a, c), (c, a), (b, c)\}$ uma relação sobre A , o fecho reflexivo desta relação é dado por:

$$\begin{aligned} \text{ref}(R) &= R \cup \text{Id}_A \\ &= \{(a, a), (a, b), (a, c), (c, a), (b, c)\} \cup \{(a, a), (b, b), (c, c)\} \\ &= \{(a, a), (a, b), (a, c), (c, a), (b, c), (b, b), (c, c)\} \end{aligned}$$

■ **Exemplo 3.34** Seja $B = \{0, 1\}$ e $K = \{(0, 0), (1, 0), (0, 1)\}$ uma relação sobre B , o fecho reflexivo desta relação é dado por:

$$\begin{aligned} \text{ref}(K) &= K \cup \text{Id}_B \\ &= \{(0, 0), (1, 0), (0, 1)\} \cup \{(0, 0), (1, 1)\} \\ &= \{(0, 0), (1, 0), (0, 1), (1, 1)\} \end{aligned}$$

■ **Exemplo 3.35** Dado o conjunto \mathbb{N} e a relação “menor que” definida nos número naturais corresponde ao conjunto $\{(x, y) \in \mathbb{N}^2 \mid (\exists z \in \mathbb{N}_*)[x = y + z]\}$ e o fecho reflexivo para tal relação é exatamente a relação “menor ou igual que” e corresponde exatamente ao conjunto $\{(x, y) \in \mathbb{N}^2 \mid (\exists z \in \mathbb{N})[x = y + z]\}$.

Definição 3.19 Seja R uma relação binária sobre A o fecho simétrico de R , denotado por $\text{sim}(R)$, corresponde a seguinte relação:

$$\text{sim}(R) = R \cup \{(y, x) \mid (x, y) \in R\}$$

Observação 3.11 O fecho simétrico também pode ser chamado de fecho dual, no sentido de descrever uma dualidade de propriedades que descrever o conjunto.

■ **Exemplo 3.36** Seja $A = \{a, b, c\}$ e $R = \{(a, a), (a, b), (a, c), (c, a), (b, c)\}$ uma relação sobre A , o fecho reflexivo desta relação é dado por:

$$\begin{aligned} \text{sim}(R) &= R \cup \{(y, x) \mid (x, y) \in R\} \\ &= \{(a, a), (a, b), (a, c), (c, a), (b, c)\} \cup \{(a, a), (b, a), (c, a), (a, c), (c, b)\} \\ &= \{(a, a), (a, b), (a, c), (c, a), (b, c), (b, a), (c, b)\} \end{aligned}$$

■ **Exemplo 3.37** Se X é o conjunto dos humanos (vivos ou mortos) e M é a relação “pai de”, então $\text{sim}(M) = \{(x, y) \in X^2 \mid x \text{ é pai de } y \text{ ou } x \text{ é filho de } y\}$.

■ **Exemplo 3.38** Dado o conjunto \mathbb{R} e a relação $D = \{(x, y) \in \mathbb{R}^2 \mid (\exists z \in \mathbb{R})[x = yz]\}$ o fecho reflexivo

de D é dado por

$$\begin{aligned} \text{sim}(D) &= D \cup \{(y, x) \mid (x, y) \in R\} \\ &= \{(x, y) \in \mathbb{R}^2 \mid (\exists z \in \mathbb{R})[x = yz]\} \cup \{(y, x) \in \mathbb{R}^2 \mid (\exists z \in \mathbb{R})[y = \frac{x}{z}]\} \\ &= \{(x, y) \in \mathbb{R}^2 \mid (\exists z \in \mathbb{R})[x = yz \vee y = \frac{x}{z}]\} \end{aligned}$$

Definição 3.20 Seja R uma relação binária sobre A o fecho transitivo de R , denotado por R^+ , corresponde a seguinte relação:

$$R^+ = \bigcup_{i=1}^{\infty} R_i$$

em que, $R^1 = R$ e $R^{i+1} = R \bullet R^i$.

O fecho transitivo é especialmente importante para diversas áreas que são usualmente de interesse dos cientistas da computação tais como a teoria dos grafos, a lógica e teoria da complexidade. Particularmente é sabido que diversos bancos de dados são construídos desde a década de 70 de forma que seja sempre possível realizar a implementação de fechos transitivos [86].

Observação 3.12 Como dito em [21] sempre que a relação R é definida sobre um conjunto finito A a Definição 3.20 pode ser reescrita como:

$$R^+ = \bigcup_{i=1}^n R_i$$

em que n é o número de elemento em A .

■ **Exemplo 3.39** Dado o conjunto $B = \{0, 1\}$ e a relação $T = \{(1, 0), (0, 1)\}$ definida sobre B tem-se que o fecho transitivo de T é dado por,

$$\begin{aligned} T^+ &= \bigcup_{i=1}^2 T_i \\ &= T_1 \cup T_2 \\ &= T \cup (T \bullet T_1) \\ &= T \cup (T \bullet T) \\ &= \{(1, 0), (0, 1)\} \cup \{(1, 0), (0, 1), (1, 1), (0, 0)\} \\ &= \{(1, 0), (0, 1), (1, 1), (0, 0)\} \end{aligned}$$

■ **Exemplo 3.40** Seja $A = \{1, 2, 3\}$ e $S = \{(1, 2), (2, 2), (3, 1), (2, 3)\}$ uma relação sobre A o fecho transitivo de R é exatamente o conjunto:

$$\begin{aligned} S^+ &= \bigcup_{i=1}^3 S_i \\ &= S_1 \cup S_2 \cup S_3 \\ &= S \cup (S \bullet S_1) \cup (S \bullet S_2) \\ &= \{(1, 2), (2, 2), (3, 1), (2, 3), (1, 3), (1, 1), (3, 3)\} \end{aligned}$$

■ **Exemplo 3.41** Para qualquer conjunto A a relação binária \subseteq definida sobre $\mathcal{P}(A)$ é um clássico exemplo de relação cujo fecho transitivo é igual a ela mesmo.

3.6 Relações e Grafos

Escrever depois...

3.7 Questionário

■ **Exercício 3.1** Dado os conjuntos $A = \{1, 2, 3\}$, $B = \{x, y\}$, $C = \{q, v\}$ e $D = \{-1, 0, 1\}$ construa os conjuntos a seguir.

- (a). $A \times B$.
- (b). $A \times C$.
- (c). $B \times D$.
- (d). $C \times A$.
- (e). $A \times A$.
- (f). $A \times (B \times D)$.
- (g). $((A \times B) \cap (D \times A)) \times C$.
- (h). $C \times ((B \times A) \cap (A \times D))$.
- (i). $B^3 \times D^2$.
- (j). $A^2 \times (C^3 \times D)$.
- (k). $A \times B \times C$.

(l). $A \times (B \times C)$.

(m). $A^2 \times (B \times C)$.

(n). $(D^3 \times C) \times B$.

(o). $A^2 \times A^2$.

■ **Exercício 3.2** Considerando $X = \{1, 2\}$, $Y = \{a, b\}$ e $Z = \{b, c\}$ determine:

(a). $X \times (Y \cup Z)$.

(b). $X \times (Y - Z)$.

(c). $(X \times Y) \times Z$.

(d). $(X \times Z) \times Y$.

(e). $X \times (Y \ominus Z)$.

(f). $X \times (Y \cap Z)$.

(g). $(X \times Y) \times (X \times Z)$.

(h). $(X \times Y) \cap (X \times Z)$.

(i). $X^2 \times Y^2$

(j). $Z^2 \times Y^2$

■ **Exercício 3.3** Sendo $(x + y, 1), (3, x - y) \in \mathbb{N}^2$ dado que $(x + y, 1) = (3, x - y)$, determine o valor de x e y .

■ **Exercício 3.4** Dado os conjuntos $A = \{P, R\}$, $B = \{N, E\}$ e $C = \{1, 2\}$ construa os diagramas de árvore para os seguintes produtos Cartesianos.

(a). A^2 .

(b). $B \times C$.

(c). $A \times (B \times C)$.

(d). $B \times (A \times C)$.

(e). $C^2 \times B$.

(f). $C^3 \times C$.

- (g). $A \times C^2$.
- (h). $B \times (C \cup A)$.
- (i). $(A \cup C)^2 \times A$.
- (j). $A^2 \times B \times C^2$.

■ **Exercício 3.5** Determine o domínio e a imagem das seguintes relações binárias.

- (a). $R_1 = \{(x, y) \in \mathbb{N} \times \mathbb{N} \mid 0 \leq x, y \leq 6, y = 2x\}$ ⁴.
- (b). $R_2 = \{(x, y) \in \mathbb{N} \times \mathbb{N} \mid 0 \leq x \leq 5, 1 \leq y \leq 4, (x - y) \in \mathbb{N}\}$.
- (c). $R_3 = \{(a, b), (e, c), (a, c), (o, c), (i, (d, f)), ((o, u), b), ((o, u), c), (u, b)\}$.
- (d). $R_4 = \{(x_0, x_1) \mid x_0 \text{ é uma consoante e } x_1 \text{ é uma vogal}\}$.
- (e). $R_5 = \{(x, y) \in \mathbb{N} \times \mathbb{N} \mid x + x = x, x + y = y\}$.

■ **Exercício 3.6** Determine o domínio, imagem e a relação inversa das seguintes relações binárias.

- (a). Relação igualdade sobre o conjunto \mathbb{N}_1 com $\mathbb{N}_1 = \{x \in \mathbb{N} \mid x > 0\}$.
- (b). Relação “maior que” sobre o conjunto $\{1, 2, 3, 4, 5, 6\}$.
- (c). Relação \subseteq sobre um conjunto $\mathcal{P}(A)$ com A sendo um conjunto arbitrário.

■ **Exercício 3.7** Dado os pares ordenados: $(6, 4), (2, 3), (3, -3), (9, 3), (3, 3), (4, 5), (3, 1), (7, 4), (2, 4), (1, 8), (-1, 1), (0, 5), (9, 3), (8, 11), (-1, 2), (4, -6), (1, 2), (10, -5), (0, 0), (4, 0), (9, 3), (5, 0), (0, 1)$. Para cada relação definida sobre os naturais a seguir diga quais pares pertencem as relações.

- (a). $A = \{(x, y) \mid x + 2y > 15\}$.
- (b). $B = \{(x, y) \mid xy \geq 8\}$.
- (c). $C = \{(x, y) \mid (\exists k \in \mathbb{N})[x + y = 2k]\}$.
- (d). $D = \{(x, y) \mid (\exists k \in \mathbb{N})[k = \sqrt{xy}]\}$.
- (e). $E = \{(x, y) \mid (\exists k \in \mathbb{N})[x^y = 2k]\}$.
- (f). $F = \{(x, y) \mid x, y \text{ são múltiplos de } 5\}$.

⁴Na relação \leq quando é escrito que $0 \leq x, y \leq z$ significa que ambas as desigualdades $0 \leq x \leq z$ e $0 \leq y \leq z$ são verdadeiras, porém, não se tem qualquer informação acerca da tricotomia entre os elementos x e y .

- (g). $R = \{(x, y) \mid x + y < 9\}$.
- (h). $S = \{(x, y) \mid 3x + y < 15\}$.
- (i). $T = \{(x, y) \mid 3x + 2y = 10\}$.
- (j). $U = \{(x, y) \mid (\exists k \in \mathbb{N})[y = \sqrt{k}]\}$.
- (k). $V = \{(x, y) \mid x - y \in \mathbb{N}\}$.
- (l). $X = \{(x, y) \mid x, y \text{ são primos}\}$.
- (m). $Y = \{(x, y) \mid mmc(x, y) = 4\}$.
- (n). $Z = \{(x, y) \mid 2x - y = x\}$.

■ **Exercício 3.8** Para cada relação a seguir diga se ela é reflexiva, simétrica, transitiva ou que não tem nenhuma destas propriedades (ou tipos).

- (a). $R_1 = \{(a, a), (a, b), (a, d), (b, b), (b, a), (c, d), (d, d)\}$.
- (b). $R_2 = \{(a, a), (b, a), (b, b), (b, c), (c, c), (c, d)\}$.
- (c). $R_3 = \{(d, d), (c, d), (d, c)\}$.
- (d). $R_4 = \{(b, c), (c, b), (b, d), (d, b)\}$.
- (e). $R_5 = \{(a, a), (a, b), (a, c), (b, c)\}$.
- (f). $R_6 = \{(a, b), (a, c)\}$.
- (g). $R_7 = \{(a, d), (c, d)\}$.
- (h). $R_8 = \{(a, a), (b, b)\}$.

■ **Exercício 3.9** Para cada relação do Exercício 3.7 diga se ela é reflexiva, simétrica, transitiva ou que não tem nenhuma destas propriedades (ou tipos).

■ **Exercício 3.10** Considere o universo do discurso $\mathbb{U} = \{a, b, c\}$ verifique se as seguintes relações definidas sobre \mathbb{U} são reflexivas, simétricas, antissimétricas ou transitivas.

- (a). $S_1 = \{(a, c), (c, c), (c, a), (b, b), (b, c), (1, 1), (1, 2)\}$.
- (b). $S_2 = \{(a, a), (c, c), (b, b), (b, c)\}$.
- (c). $S_3 = \{(a, a), (a, b), (b, c), (c, a), (a, c)\}$.

(d). $S_4 = \{(a, a), (a, b), (b, c), (a, c)\}$.

■ **Exercício 3.11** Considere o universo do discurso $\mathbb{U} = \{a, b, c, d, e, f\}$ verifique se as seguintes relações definidas sobre \mathbb{U} são reflexivas, simétricas, antissimétricas ou transitivas.

(a). $T = \{(a, a), (b, b), (c, c), (e, e), (f, f), (a, b), (b, c), (c, e), (e, f)\}$.

(b). $U = \{(a, b), (b, a), (c, e), (e, c), (e, f), (f, e)\}$.

(c). $V = \{(a, b), (b, c), (a, c), (c, a), (c, b), (b, a), (a, a), (b, b), (c, c)\}$.

(d). $X = \{(a, a), (b, b), (c, c), (e, e), (f, f), (e, f), (f, e)\}$.

(e). $Y = \emptyset$.

■ **Exercício 3.12** Esboce a relação inversa de cada uma das relações dos Exercícios 3.10 e 3.11.

■ **Exercício 3.13** Considere o universo do discurso $\mathbb{U} = \{1, 2, 0, 4\}$ e construa sobre \mathbb{U} :

(a). Uma relação que seja reflexiva e transitiva, mas não seja simétrica.

(b). Uma relação que seja reflexiva e simétrica, mas não seja transitiva.

(c). Uma relação que seja simétrica e transitiva, mas não seja reflexiva.

(d). Uma relação que não seja reflexiva e nem transitiva, e também não seja simétrica.

■ **Exercício 3.14** Esboce os fechos reflexivo, simétrico e transitivo de cada relação no Exercício 3.10.

■ **Exercício 3.15** Esboce os fechos reflexivo, simétrico e transitivo de cada relação no Exercício 3.11.

Capítulo 4

Equivalência e Ordem

“Matemática não é difícil, matemática tem muita lógica e o que é lógico não pode ser difícil”.

João Lucas Marques Barbosa

“O grande inimigo do conhecimento não é a ignorância, é a ilusão de ter conhecimento”.

Stephen Hawking

4.1 Introdução

No Capítulo 3 anterior este manuscrito apresentou ao leitor a ideia de relações entre conjuntos, em especial foram tratadas as relações binários sobre um conjunto dado. Agora nesta seção será apresentada de forma mais profunda as relações de equivalência. Como dito em [2, 21], as relações de equivalência ao lado das relações de ordem (estudadas na Seção 4.3) são de importante central para toda a matemática, além disso, as relações de equivalência também desempenho importantes papéis nas área de mineração de dados [59, 60], aprendizado de máquina [8] e processamento de sinais [58] e imagens [3, 81]. E por sua vez, as relações de ordem também aparecem em diversas áreas de caráter prática tais como processamento de imagens [36, 29], criptografia [41], otimização [53] e etc.

4.2 Relações de Equivalência e Espaço Quociente

Mas o que seria uma relação de equivalência? Bem, uma resposta satisfatória para essa pergunta é que uma relação de equivalência pode ser entendida como sendo uma forma de parear os elementos de um conjunto que apresentam similaridade entre si com respeito a uma ou mais propriedades específicas, isto é, uma relação de equivalência junta os elementos em pares pela similaridade deles. A seguir será apresentado de forma precisa o conceito de relação de equivalência.

Definição 4.1 — Relação de Equivalência. Seja A um conjunto uma relação binária \equiv sobre A é dita ser uma relação de equivalência sempre que \equiv for reflexiva, simétrica e transitiva.

Observação 4.1 Como dito em [21] além da notação \equiv outros símbolos também são comumente encontrados na literatura para representar relações de equivalência, entre, tais símbolos destacam-se \approx e \sim . Neste manuscrito tais símbolos apareceram mais adiante representando outros conceitos, assim neste manuscrito será usado sempre usado o símbolo \equiv , a menos que seu uso gere confusão e nesse caso será usado notações como R_i para denotar relação de equivalência, em que i poderá ser um número natural ou o rótulo de um conjunto.

■ **Exemplo 4.1** Dado um conjunto C qualquer, a relação de igualdade ($=$) definida em C é obviamente uma relação de equivalência.

■ **Exemplo 4.2** Dado o conjunto $A = \{a, b, c, d\}$ a relação definida por $a \equiv a, a \equiv b, b \equiv a, b \equiv b, c \equiv c$ e $d \equiv d$, é claramente uma relação de equivalência.

Os Exemplos 4.1 e 4.2 permite o leitor perceber uma importante verdade matemática, tal verdade como expressa em [21] pode ser escrita como: “objetos iguais são equivalentes, mas objetos equivalentes nem sempre são iguais”.

■ **Exemplo 4.3** Dado um plano P a relação de paralelismo definido sobre o conjunto de retas de P é uma relação de equivalência, outro exemplo clássico da geometria é a semelhança entre triângulos neste mesmo plano P .

■ **Exemplo 4.4** A relação $R = \{(x, y) \in \mathbb{Z}^2 \mid x, y \text{ possuem o mesmo resto da divisão por } 5\}$ é uma relação de equivalência.

■ **Exemplo 4.5** A relação $I = \{(x, y) \in PERS^2 \mid x, y \text{ possuem a mesma idade}\}$ é uma relação de equivalência sobre o conjunto de todas as pessoas ($PERS$).

■ **Exemplo 4.6** Dado o conjunto de todos os times de futebol do Brasil a relação T definida como $x T y \iff x, y$ nunca foram rebaixados para a segunda divisão, é uma relação de equivalência.

Definição 4.2 — Classes de Equivalência. Seja \equiv uma relação de equivalência sobre um conjunto A , para todo $x \in A$ é definida a classe de equivalência de x , denotado por $[x]$, como sendo o conjunto de todos os elementos equivalentes a x , simbolicamente tem-se que:

$$[x] = \{y \in A \mid y \equiv x\}$$

Obviamente toda classe de equivalência $[x]$ é um subconjunto do conjunto base¹. Além disso, obviamente tem-se que $[x] = \emptyset$ se, e somente se, o conjunto base for vazio.

■ **Exemplo 4.7** Seja $A = \{0, 1, 2, 3\}$ e $0 \equiv 0, 1 \equiv 1, 2 \equiv 2, 3 \equiv 3, 0 \equiv 2, 1 \equiv 3, 2 \equiv 0, 3 \equiv 1$ tem-se que: $[0] = \{0, 2\}, [1] = \{1, 3\}, [2] = \{0, 2\}$ e $[3] = \{1, 3\}$.

■ **Exemplo 4.8** A relação $a \equiv a, b \equiv b, c \equiv c, a \equiv b, b \equiv a$ definida sobre o conjunto $\{a, b, c\}$ é uma relação de equivalência e existem as seguintes classes de equivalência $[a] = [b] = \{a, b\}$ e $[c] = \{c\}$.

Teorema 4.1 Seja \equiv uma relação de equivalência sobre um conjunto A não vazio e sejam $a, b \in A$ tem-se que $a \equiv b$ se, e somente se, $[a] = [b]$.

Demonstração. (\Rightarrow) Suponha que $a \equiv b$, assim dado qualquer $x \in [a]$ tem-se que $x \equiv a$, agora pela transitiva de \equiv é claro que $x \equiv b$ e, portanto, $x \in [b]$, logo $[a] \subseteq [b]$ e com raciocínio similar pode-se concluir que $[b] \subseteq [a]$ e assim pela Definição 1.9 tem-se que $[a] = [b]$. (\Leftarrow) Suponha que $[a] = [b]$, por \equiv ser reflexiva é claro que $a \equiv a$ e assim $a \in [a]$, mas como $[a] = [b]$ tem-se que $a \in [b]$, e portanto, por definição $a \equiv b$. \square

Teorema 4.2 Seja \equiv uma relação de equivalência sobre um conjunto A não vazio e sejam $a, b \in A$ tem-se que $a \not\equiv b$ se, e somente se, $[a] \cap [b] = \emptyset$.

Demonstração. (\Rightarrow) Suponha por absurdo que $a \not\equiv b$ e $[a] \cap [b] \neq \emptyset$, logo existe um $x \in A$ tal que $x \in [a] \cap [b]$, mas assim pela Definição 1.14 tem-se que $x \in [a]$ e $x \in [b]$, logo $x \equiv a$ e $y \equiv b$, mas uma vez que \equiv é simétrica tem-se que $a \equiv x$, e como \equiv é transitiva tem-se que $a \equiv b$, o que contradiz a hipótese, caracterizando um absurdo, consequentemente, se $a \not\equiv b$ tem-se então que $[a] \cap [b] = \emptyset$. (\Leftarrow) Suponha que $[a] \cap [b] = \emptyset$, como $a \in [a]$ e pela hipótese $a \notin [a] \cap [b]$ tem-se que $a \notin [b]$ e, portanto, $a \not\equiv b$. \square

Definição 4.3 — Espaço Quociente. Seja A um conjunto e \equiv uma relação de equivalência sobre A , o espaço quociente de A com respeito (ou módulo) \equiv , denotado por $A_{/\equiv}$, é o conjunto de todas as classes de equivalência do conjunto A , na linguagem na teoria dos conjuntos tem-se que:

$$A_{/\equiv} = \{[x] \mid x \in A\}$$

¹Conjunto base aqui diz respeito ao conjunto sobre o qual a relação de equivalência está definida.

■ **Exemplo 4.9** Seja $A = \{1, 2, 3\}$ e $R = \{(a, a), (b, b), (c, c), (a, b), (b, a)\}$ claramente R é uma relação de equivalência e além disso $[a] = [b] = \{a, b\}$ e $[c] = \{c\}$ assim $A/R = \{[a], [c]\}$.

■ **Exemplo 4.10** Dado que a relação $P = \{(x, y) \in \mathbb{Z}_+ \mid x, y \text{ tem o mesmo resto da divisão por } 2\}$ é uma relação de equivalência sobre \mathbb{Z}_+ (a prova fica como exercício ao leitor) tem-se claramente que,

$$[0] = \{0, 2, 4, 6, 8, \dots\}$$

e

$$[1] = \{1, 3, 5, 7, 9, \dots\}$$

ou seja, $[0]$ é o conjunto dos pares positivos e $[1]$ é o conjunto dos ímpares positivos, assim claramente tem-se que $\mathbb{Z}_{+/p} = \{[0], [1]\}$.

Uma fato importante sobre o espaço quociente de uma relação de equivalência \equiv é que sempre que o conjunto base $A \neq \emptyset$ tem-se que $A/_\equiv \neq \emptyset$, e mais do que isso, como mostrado a seguir o espaço quociente é sempre uma partição sobre o conjunto base A .

Teorema 4.3 Seja \equiv uma relação de equivalência sobre um conjunto não vazio A , então $A/_\equiv$ é uma partição de A .

Demonstração. Primeiramente note que como \equiv é uma relação reflexiva tem-se para todo $x \in A$ que $x \in [x]$ e assim claramente $[x] \in A/_\equiv$ e $[x] \neq \emptyset$, satisfazendo assim a condição (1) da Definição 1.22. Por outro lado, os Teoremas 4.1 e 4.2 mostram que dado $[x], [y] \in A/_\equiv$ sempre que $[x] \neq [y]$ tem-se que $[x] \cap [y] = \emptyset$ e, portanto, a condição (2) da Definição 1.22 é satisfeita, desde que as condições (1) e (2) são satisfeita pelo elementos de $A/_\equiv$ tem-se que $A/_\equiv$ é uma partição de A . \square

Um interessante uso das relações de equivalência é a construção do conjunto dos números racionais (\mathbb{Q}) a partir dos números inteiros (\mathbb{Z}) como dito em [21].

4.3 Relações de Ordem

Em algumas situação é interessante que seja possível definir uma hierarquia entre os elementos de um determinado conjunto, de fato, como dito em [2] diversos campos das ciências empíricas, tais como a área da biologia comparada, são dependentes de construções hierárquicas. Dentro da própria ciência da computação diversas áreas (estrutura de dados, classificação de dado e etc.) também utilizam de ordens de hierarquia. Assim é conveniente apresentar o estudo das relações de ordem e das estruturas existentes envolta de tais relações, para isso apresenta-se primeiro as ideias de pré-ordem e ordem estrita.

Definição 4.4 — Ordem Estrita. Seja A um conjunto uma relação \sqsubset sobre A , é dita ser uma relação de ordem parcial estrita, ou simplesmente ordem estrita, sempre que \sqsubset for irreflexiva e transitiva.

- **Exemplo 4.11** A relação $\{(x, y) \in \mathbb{N}^2 \mid (\exists k \in \mathbb{N})[k \geq 1 \wedge y = x + k]\}$ é uma ordem estrita.
- **Exemplo 4.12** Dado um conjunto A qualquer a relação de ser subconjunto próprio (\subset) é um ordem estrita sobre o conjunto $\mathcal{P}(A)$.
- **Exemplo 4.13** Dado o conjunto $\{1, 2, 3\}$ a relação $\{(1, 2), (2, 3), (2, 2), (1, 3)\}$ não é uma ordem estrita pois $(2, 2)$ é um par pertencente a relação.

Definição 4.5 — Pré-ordem. Seja A um conjunto uma relação \sqsubseteq sobre A , é dita ser uma relação de pré-ordem sempre que \sqsubseteq for reflexiva e transitiva.

- **Exemplo 4.14** Dado o conjunto $\{a, b, c\}$ a relação $\{(a, a), (b, b), (c, c), (b, c), (c, a), (b, a)\}$ é uma relação de pré-ordem.
- **Exemplo 4.15** A relação $\{(x, y) \in \mathbb{N}^2 \mid (\exists x \in \mathbb{N})[y = xk]\}$ é uma pré-ordem.
- **Exemplo 4.16** A relação $\{(x, y) \in \mathbb{N}^2 \mid x + y \neq x \text{ e } x + y \neq y\}$ não é uma pré-ordem pois não é reflexiva, basta notar que $(0, 0)$ não pertence a tal relação.

Aumentando as restrições sobre uma pré-ordem, isto é, adicionando mais propriedades a serem exigidas, é construída a noção de ordem parcial, tal conceito é formalizado a seguir.

Definição 4.6 — Ordem Parcial. Seja A um conjunto uma relação \sqsubseteq sobre A , é dita ser uma relação de ordem parcial sempre que \sqsubseteq for reflexiva, antissimétrica e transitiva.

Como dito em [2] se \sqsubseteq é uma ordem parcial sobre um conjunto A , então tem-se que \sqsubseteq organizar (ou ordena) o conjunto A em uma determinada hierarquia (ou ordem), obviamente um mesmo conjunto pode apresentar diferentes ordenações, ou seja, podem existir diversas ordens parciais sobre A .

Observação 4.2 De forma geral quando $x \sqsubseteq y$ pode ser interpretado como x é anterior ou igual a y , entretanto, para o caso específico das relações de ordem parcial \leq e \subseteq suas semânticas são as aquelas que o leitor já conhece, isto é, $x \leq y$ significa x é menor ou igual a y e $X \subseteq Y$ significa que X é subconjunto de Y .

Além das duas famosas ordens parciais mencionadas na Observação 4.2 a seguir serão apresentados mais algumas ordens parciais.

- **Exemplo 4.17** As seguintes relações são exemplos de ordens parciais:

- (a) $\{(x, y) \in \mathbb{N}^2 \mid (\exists k \in \mathbb{N})[y = x + k]\}$.
- (b) $\{(x, y) \in \mathbb{N}^2 \mid (\exists k \in \mathbb{N})[y = xk]\}$.

- (c) Dado um conjunto A de todas as pessoas da terra a relação $x R y$ se, e somente se, x tem a mesma altura ou é mais alto que y , é uma ordem parcial sobre A .

Observação 4.3 O leitor atento pode notar que a ordem no item (b) do Exemplo 4.17 foi anteriormente usado como exemplo de uma pré-ordem, isso ocorre por que toda ordem parcial é uma pré-ordem.

■ **Exemplo 4.18** Seja $A = \{a, b, c, d\}$ a relação $\{(a, a), (b, b), (c, c), (d, d), (a, b), (b, c), (a, c), (a, d)\}$ é uma ordem parcial sobre A .

■ **Exemplo 4.19** Dado o $\{0, 1\}$ a relação $\{(0, 0), (0, 1), (1, 1)\}$ é uma ordem parcial.

■ **Exemplo 4.20** Dado um conjunto $\{0, 1, 2, 3\}$ a relação $\{(0, 0), (0, 1), (2, 2), (1, 2), (0, 2), (3, 3)\}$ não é um ordem parcial pois o par $(1, 1)$ não pertence a relação e por definição uma ordem parcial deve ser reflexiva.

A partir da ideia de ordem parcial é possível definir o conceito de comparabilidade como se segue.

Definição 4.7 — Comparabilidade. Seja A um conjunto não vazio, \sqsubseteq uma ordem parcial sobre A e seja $x, y \in A$, é dito que x e y são comparáveis sempre que $x \sqsubseteq y$ ou $y \sqsubseteq x$.

Como dito em [2, 21] tem-se que a noção de comparabilidade está ligada a ordem em questão, assim pode haver um conjunto A e uma ordem parcial \sqsubseteq_1 tal que dois elementos x e y são comparáveis, entretanto, pode haver outra ordem parcial \sqsubseteq_2 sobre o mesmo conjunto tal que os elementos x e y não pode ser comparáveis².

■ **Exemplo 4.21** Considere o conjunto $A = \{1, 2, 3\}$ tem-se que \subseteq será obviamente um ordem parcial sobre $\mathcal{P}(A)$, agora note que $\{1, 2\} \subseteq A$, portanto, $\{1, 2\}$ e A são comparáveis, por outro lado, $\{1, 2\} \not\subseteq \{1, 3\}$ e $\{1, 3\} \not\subseteq \{1, 2\}$, logo $\{1, 2\}$ e $\{1, 3\}$ são incomparáveis.

■ **Exemplo 4.22** Dado o conjunto \mathbb{R} e a ordem parcial \leq sobre R tem-se que todo par de números reais (x, y) é sempre comparável.

Por fim é apresentado a ideia de ordem parcial, pela definição a seguir é fácil para o leitor perceber que toda ordem total é uma ordem parcial, porém, o oposto não é verdade.

Definição 4.8 — Ordem total. Uma ordem parcial \sqsubseteq sobre um conjunto A é dita ser total quando para todo par de elementos $x, y \in A$ é comparável.

■ **Exemplo 4.23** São exemplos de ordem totais:

(a) A ordem usual “menor igual” (\leq) sobre o conjunto \mathbb{R} .

(b) A relação $\{(a_0, a_0), (a_0, a_1), (a_0, a_2), (a_1, a_1), (a_1, a_2), (a_2, a_2)\}$ sobre o conjunto $\{a_0, a_1, a_2\}$.

²Em algumas obras é usado a escrita $x \not\sqsubseteq y$ para esboça que x e y são incomparáveis por uma ordem parcial \sqsubseteq .

Nesta seção ao apresentar a ideia de relações de ordem isso era feito pelo estudo da relação em si ficando seu conjunto base em segundo plano e com pouco interesse, agora será considerada simultaneamente os dois conceitos juntos, ou seja, será agora apresentado o conceito de conjunto parcialmente ordenado.

4.4 Posets e Diagramas de Hasse

Como dito em [84], os conjuntos parcialmente ordenados ou *posets* (em inglês) têm uma longa história que remonta ao início do século XIX, onde as propriedades da ordenação dos subconjuntos de um conjunto foram investigadas. Embora o matemático Felix Hausdorff³ (1868-1942) não tenha sido a pessoa que introduziu a ideia de conjunto parcialmente ordenado, foi ele que fez o primeiro estudo sério de uma teoria geral dos posets em seu trabalho [46].

Definição 4.9 — Poset. Um conjunto parcialmente ordenado ou *poset* é uma estrutura $\langle A, \sqsubseteq \rangle$ onde A é um conjunto não vazio e \sqsubseteq é uma ordem parcial sobre A .

■ **Exemplo 4.24** São exemplos de conjuntos parcialmente ordenados:

- (a) $\langle \mathcal{P}(A), \subseteq \rangle$.
- (b) $\langle C, : \rangle$ onde $C = \{1, 2, 3, 4, 5, 6, 12\}$ e $x : y$ se, e somente se, x é um divisor de y .
- (c) $\langle \mathbb{Z}, \geq \rangle$.
- (d) $\langle B, \subseteq \rangle$ onde $B = \{R_1, R_2, \dots\}$ é o conjunto de todas as relações binárias sobre um conjunto A .

Agora como dito em [78], muitas vezes é conveniente utilizar uma representação gráfica para os *posets* que possa evidenciar as relações hierárquicas existentes entre os elementos do conjunto base. Essa representação como dito em [2], é chamada de diagrama de Hasse⁴. Vale salientar que tal representação não é para qualquer *poset* apenas os *posets* finitos podem ser representados por tais diagramas de forma completa.

O diagrama de Hasse é um grafo orientado acíclico construído utilizando a relação “ x cobre y ” sempre que $x \sqsubseteq y$, o diagrama é construído para um *poset* $\langle A, \sqsubseteq \rangle$ com A finito usando as seguintes regras:

- (r_1) Para todo $x, y \in A$ se $x \sqsubseteq y$ e não existe um z tal que $x \sqsubseteq z$ e $z \sqsubseteq y$ com $x \neq y$, então o ponto de x aparece inferior no diagrama ao ponto de y .
- (r_2) Para todo $x, y \in A$ se x e y satisfazem (r_1), então os pontos de x e y são ligados por segmento de reta.

³Famoso por seus trabalhos em topologia.

⁴Em homenagem ao matemático alemão Helmut Hasse (1898-1979) que introduziu tais diagramas.

(r_3) Todos os elementos $x \in A$ devem aparecer no diagrama como um ponto (ou nó).

■ **Exemplo 4.25** Considere o poset $\langle \{a, 2, 1, b\}, R \rangle$ onde $R = \{(1, 1), (a, a), (1, a), (b, b), (1, b), (2, 2), (b, 2), (a, 2), (1, 2)\}$. Agora note que como $(1, a)$ satisfaz a regra (r_1), assim o ponto de 1 aparece inferior no diagrama ao ponto de a e o mesmo iria valer para os casos $(1, b)$, $(a, 2)$ e $(b, 2)$, logo pode-se estabelecer a seguinte distribuição espacial dos pontos:

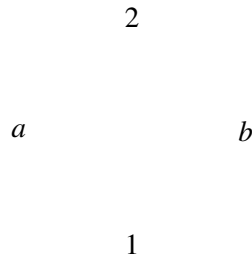


Figura 4.1: Distribuição espacial dos pontos para o diagrama de Hasse do poset $\langle \{a, 2, 1, b\}, R \rangle$.

Agora executando a regra (r_2) são ligados os pontos afim de ilustrar as relações entre os elementos, ficando a figura como se segue:

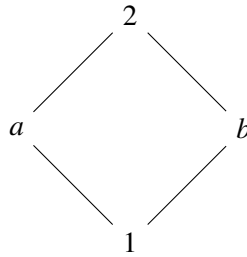


Figura 4.2: Diagrama de Hasse do poset $\langle \{a, 2, 1, b\}, R \rangle$.

Como todos os elementos de $\{a, 2, 1, b\}$ já estão no diagrama então não há nada mais a fazer.

■ **Exemplo 4.26** O poset $\langle \{0, 1, c, d\}, Lip \rangle$ onde $Lip = \{(x, y) \mid x \leq y \text{ ou } (x \in \{0, 1\}, y \in \{a, b\})\}$ pode ser representado pelo diagrama a seguir.

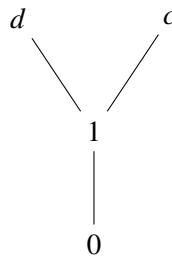


Figura 4.3: Diagrama de Hasse do poset $\langle \{0, 1, c, d\}, Lip \rangle$

Observação 4.4 Vale salientar que a representação por diagrama não é única, em termos de distribuição espacial (desenho do grafo).

■ **Exemplo 4.27** O poset $\langle \mathcal{P}(\{a, b, c\}), \subseteq \rangle$ pode ser representado pelo dois diagramas a seguir.

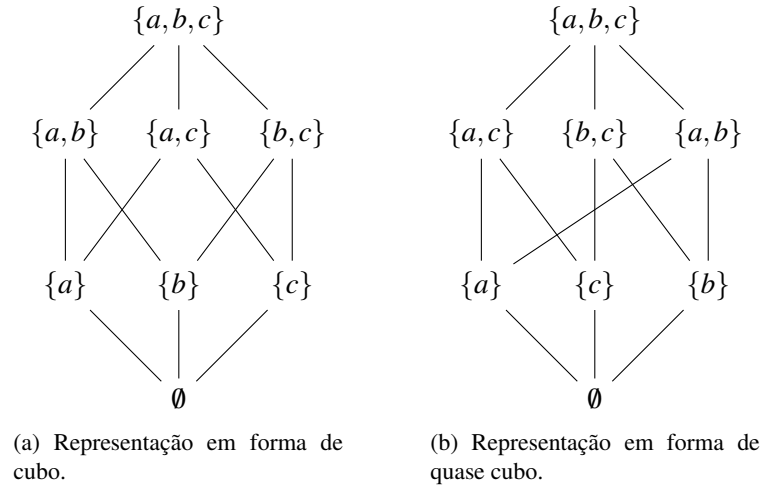


Figura 4.4: Diagramas de Hasse para o poset $\langle \mathcal{P}(\{a, b, c\}), \subseteq \rangle$.

■ **Exemplo 4.28** O poset $\langle \{5, 6, 7, 8, 9\}, \leq \rangle$ é representado pelo diagrama a seguir.

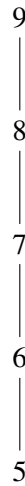


Figura 4.5: Diagrama de Hasse do poset $\langle \{5, 6, 7, 8, 9\}, \leq \rangle$



Nota 4.1 Posets cuja ordem é total também são chamados de cadeias (ver [2, 78]) e nesse caso o diagrama será uma linha reta com todos os elementos sobre essa linha, exatamente como no Exemplo 4.28. O leitor sem saber já usava esse conceito em seus estudos matemáticos a proferir frases como “a reta real” ou “a reta dos números reais”.

Agora obviamente dado um diagrama de Hasse sempre é possível recuperar a estrutura do poset do mesmo, isto é, recuperar o conjunto base e a relação de ordem parcial que define o poset, a seguir são

apresentados alguns exemplos disto.

■ **Exemplo 4.29** Dado o diagrama de Hasse da Figura 4.6 a seguir, dado que 6 está ligado e abaixo de 3 e 2 tem-se que $(6, 3), (6, 2) \in R$ onde R é uma relação, o mesmo vale para os pares $(3, 5), (2, 4), (5, 1)$ e $(4, 1)$. Desse modo tal figura representa o *poset* $\langle A, R \rangle$ em que $A = \{1, 2, 3, 4, 5, 6\}$ e R corresponde a relação $\{(6, 3), (6, 2), (3, 5), (2, 4), (5, 1), (4, 1), (6, 5), (6, 4), (6, 1), (2, 1), (5, 1)\} \cup Id_A$.

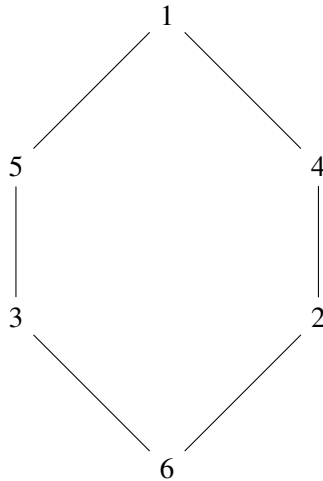


Figura 4.6: Um diagrama de Hasse.

■ **Exemplo 4.30** Dado o diagrama de Hasse da Figura 4.7 a seguir representa o *poset* $\langle B, R_B \rangle$ em que $B = \{0, 1, 2, 3, 4, 5, 6\}$ e $R_B = Id_B \cup \{(5, 1), (6, 1), (1, 2), (5, 2), (6, 2), (2, 3), (2, 4), (1, 3), (1, 4)\}$.

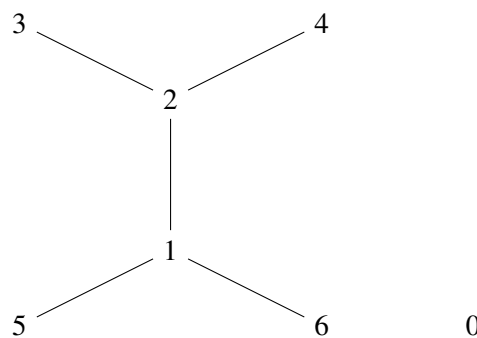


Figura 4.7: Um diagrama de Hasse.

Como dito em [21] um *poset* é uma estrutura relacional ordenado, ou ainda, uma álgebra⁵ relacional⁶, e como uma álgebra pode-se estudar: os elementos destacados, suas sub-álgebras (sub-estruturas) ou ainda os mecanismo de extensão da mesma. Na próxima seção este manuscrito irá realizar um estudo sobre os pontos.

⁵O conceito de álgebra será melhor formalizado em capítulos futuros deste manuscrito, por hora o leitor pode pensar em uma álgebra como uma estrutura.

⁶Uma álgebra relacional é uma estrutura criada na interseção da lógica de primeira ordem e da álgebra booleana (ou álgebra de conjuntos) finita, tal tipo de álgebra é de suma importância para a área de banco de dados.

4.5 Elementos Notáveis de um *Poset*

Como muito bem aprofundado em [21, 78, 84], existem diversos conceitos e aplicações interessantes ligados a ideia de *posets*, assim para ajudar que leitor tenha uma formação sólida na área de teoria dos *posets* é interessante apresentar alguns destes conceitos-chaves da teoria, nesta seção serão trabalhados os elementos notáveis existentes nos *posets* e suas propriedades.

Definição 4.10 — Máximo e Mínimo de um subconjunto. Seja $\langle A, \sqsubseteq \rangle$ um *poset* e $X \subseteq A$ o máximo de X , denotado por $\max(X)$, é um elemento $x \in X$ tal que para todo $y \in X$ tem-se que $y \sqsubseteq x$. O mínimo de X , denotado por $\min(X)$, é um elemento $x \in X$ tal que para todo $y \in X$ tem-se que $x \sqsubseteq y$.

■ **Exemplo 4.31** Considere o *poset* da Figura 4.2 para o conjunto $X = \{1, 2, a\}$ tem-se que $\min(X) = 1$ e $\max(X) = 2$.

■ **Exemplo 4.32** Considere o *poset* da Figura 4.3 para o conjunto $X = \{0, 1\}$ tem-se que $\min(X) = 0$ e $\max(X) = 1$.

■ **Exemplo 4.33** Considere o *poset* da Figura 4.4a para o conjunto $X = \{\{a, b\}, \{a\}, \{b\}, \emptyset\}$ tem-se que $\min(X) = \emptyset$ e $\max(X) = \{a, b\}$.

Agora é conveniente ressaltar que tanto máximo quanto o mínimo podem vir a não existir, isto é, dado um *poset* $\langle A, \sqsubseteq \rangle$ um subconjunto X de A pode-se de tal forma que ele não contenha máximo e(ou) mínimo.

■ **Exemplo 4.34** Considere o *poset* ilustrado pela Figura 4.6 o subconjunto $\{3, 2, 5, 4\}$ deste *poset* não apresenta máximo e nem mínimo.

■ **Exemplo 4.35** Considere o *poset* ilustrado pela Figura 4.7 o subconjunto $\{1, 5, 6\}$ deste *poset* possui máximo mas não possui mínimo.

Teorema 4.4 — Unicidade do máximo. Seja $\langle A, \sqsubseteq \rangle$ um *poset* e $X \subseteq A$. Se existe $x \in X$ tal que $\max(X) = x$, então x é único.

Demonstração. Dado $\langle A, \sqsubseteq \rangle$ um *poset* e $X \subseteq A$. Suponha por absurdo que existem $x, x' \in X$ tal que $\max(X) = x$ e $\max(X) = x'$ com $x \neq x'$, logo por definição para todo $a \in X$ tem-se que $a \sqsubseteq x$ e $a \sqsubseteq x'$, mas desde que $x, x' \in X$ tem-se que $x \sqsubseteq x'$ e $x' \sqsubseteq x$ e, desde que, \sqsubseteq é antissimétrica tem-se que $x = x'$ o que contradiz a hipótese e, portanto, se existe o máximo de X ele é único. \square

Teorema 4.5 — Unicidade do mínimo. Seja $\langle A, \sqsubseteq \rangle$ um *poset* e $X \subseteq A$. Se existe $x \in X$ tal que $\min(X) = x$, então x é único.

Demonstração. Similar a demonstração do Teorema 4.4. \square

Teorema 4.6 Seja $\langle A, \sqsubseteq \rangle$ um *poset* e $X, Y \subseteq A$ com $X \subseteq Y$ tem-se que:

- (i) Se $\max(X) = a$ e $\max(Y) = b$, então $a \sqsubseteq b$.
- (ii) Se $\min(X) = a$ e $\min(Y) = b$, então $b \sqsubseteq a$.

Demonstração. A demonstração é simples e fica como exercício ao leitor. □

Além da ideia de máximo e mínimo sobre os *posets* também definido a ideia de maximais e minimais.

Definição 4.11 — Elementos maximais. Seja $\langle A, \sqsubseteq \rangle$ um *poset* um elemento $x \in A$ é dito ser um maximal se para todo $y \in A$ tem-se que se $x \sqsubseteq y$, então $x = y$.

■ **Exemplo 4.36** Considere o *poset* esboçado pela Figura 4.8 a seguir. Para tal *poset* tem-se que o elemento g é um maximal e também é o máximo do conjunto.

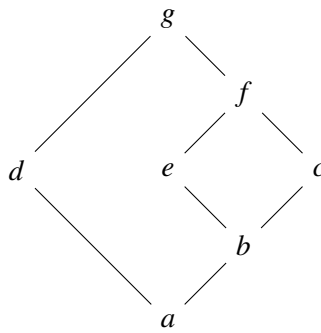


Figura 4.8: Um *poset* representado por um diagrama de Hasse.

■ **Exemplo 4.37** Considere o *poset* esboçado pela Figura 4.6. Para tal *poset* tem-se que o elemento 1 é um maximal e também é o máximo do conjunto.

Em algumas situações elementos maximais coincidem com o máximo, entretanto, como mostrado a seguir ser maximal não é garantia de ser o máximo do conjunto.

■ **Exemplo 4.38** Considere o *poset* esboçado pela Figura 4.9 a seguir. Para tal *poset* tem-se que os elementos 4 e 5 satisfazem a Definição 4.11 logo ambos são maximais do conjunto, porém, ambos são incomparáveis, portanto, nenhum é o máximo do conjunto.

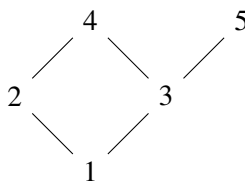


Figura 4.9: Um *poset* representado por um diagrama de Hasse.

De forma dual ao conceito de maximal pode-se definir a ideia de elemento minimal, e isto é feito como se segue.

Definição 4.12 — Elementos minimais. Seja $\langle A, \sqsubseteq \rangle$ um *poset* um elemento $x \in A$ é dito ser um minimal se para todo $y \in A$ tem-se que se $y \sqsubseteq x$, então $x = y$.

■ **Exemplo 4.39** Considerando o *poset* da Figura 4.8 tem-se que o elemento a é minimal de tal conjunto e também apresenta a característica de ser o mínimo.

■ **Exemplo 4.40** Considerando o *poset* da Figura 4.9 tem-se que o elemento 1 é minimal de tal conjunto e também apresenta a característica de ser o mínimo.

■ **Exemplo 4.41** Considerando o *poset* da Figura 4.7 tem-se que os elementos 5 e 6 são ambos minimais, porém, como são ambos incomparáveis tem-se que nenhum dos dois é o mínimo do conjunto base do *poset*.

■ **Exemplo 4.42** Considere o *poset* da Figura 4.10 a seguir, tem-se que g e f são elementos maximais e c, a e b são elementos minimais, note que tal *poset* não possui máximo e mínimo.

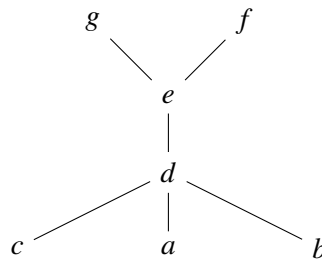


Figura 4.10: Um *poset* representado por um diagrama de Hasse.

Definição 4.13 — Majorante. Seja $\langle A, \sqsubseteq \rangle$ um *poset* e $X \subseteq A$, um elemento $x \in A$ é dito ser um majorante (ou cota superior) de X sempre que para todo $y \in X$ tem-se que $y \sqsubseteq x$.

■ **Exemplo 4.43** Considere o *poset* esboçado na Figura 4.10 e o subconjunto $X = \{e, d, c\}$ tem-se que os elementos e, f e g são todos majorantes de X . Já os subconjuntos $Y = \{g, f, e\}$ e $Z = \{g, f, a\}$ não possuem majorantes.

■ **Exemplo 4.44** Considere o *poset* esboçado na Figura 4.9 e o subconjunto $X = \{2, 3, 5\}$ não possui majorantes, já o conjunto $Y = \{1, 2, 3, 4\}$ possui o elemento 4 como majorante.

■ **Exemplo 4.45** Considere o *poset* esboçado na Figura 4.8 e o subconjunto $X = \{d, e, f\}$ tem-se que o elemento g é o majorante de X .

Como mencionado em [2] se x é um majorante (ou cota superior) de um conjunto X , é dito que X é um conjunto majorado pelo elemento x ou que o conjunto X possui uma cota superior x .

Definição 4.14 — Conjunto dos majorantes. Seja $\langle A, \sqsubseteq \rangle$ um *poset* e $X \subseteq A$, o conjunto de todos os majorantes de X é denotado por X_λ .

Como dito em [2, 21, 78] a maior parte dos conceitos na teoria dos *posets* tem natureza dual, assim é natural que seja imediatamente apresentadas as definições de minorantes e conjunto dos minorantes.

Definição 4.15 — Minorante. Seja $\langle A, \sqsubseteq \rangle$ um *poset* e $X \subseteq A$, um elemento $x \in A$ é dito ser um minorante (ou cota inferior) de X sempre que para todo $y \in X$ tem-se que $x \sqsubseteq y$.

■ **Exemplo 4.46** Considere o *poset* esboçado na Figura 4.10 e o subconjunto $X = \{e, d, c\}$ tem-se que o elemento c é o minorante de X . Já os subconjuntos $Y = \{g, f, e\}$ tem como minorantes os elementos e, d, c, a e b , por fim, para o conjunto $Z = \{g, f, a\}$ o único minorante é o elemento a .

■ **Exemplo 4.47** Considere o *poset* esboçado na Figura 4.9 e o subconjunto $X = \{2, 3, 5\}$ tem como minorante o elemento 1, já o conjunto $Y = \{3, 4, 5\}$ possui como minorantes os elementos 1 e 3.

■ **Exemplo 4.48** Considere o *poset* esboçado na Figura 4.8 e o subconjunto $X = \{d, e, f\}$ tem-se que o elemento a é o único minorante deste conjunto, já para o conjunto $Y = \{g, f\}$ tem-se os elementos f, e, c, b e a como minorantes de Y .

Definição 4.16 — Conjunto dos minorantes. Seja $\langle A, \sqsubseteq \rangle$ um *poset* e $X \subseteq A$, o conjunto de todos os minorantes de X é denotado por X_γ .

Teorema 4.7 Seja $\langle A, \sqsubseteq \rangle$ um *poset* e $X \subset A$. Se $a \in A$ é um majorante (minorante) de X e $a \in X$, então $a = \max(X)$ ($a = \min(X)$).

Demonstração. Trivial. □

O próximo resultado estabelece que o conjunto de majorantes e minorantes respeita a monotonicidade da relação inclusão.

Teorema 4.8 Seja $\langle A, \sqsubseteq \rangle$ um *poset* e $X, Y \subseteq A$. Se $X \subseteq Y$, então:

- (i) $Y_\lambda \subseteq X_\lambda$.
- (ii) $Y_\gamma \subseteq X_\gamma$.

Demonstração. A demonstração é simples e ficará como exercício ao leitor. □

Definição 4.17 — Conjunto limitado. Seja $\langle A, \sqsubseteq \rangle$ um *poset* e $X \subseteq A$, o conjunto X é dito ser limitado em A sempre que X possui majorantes e minorantes.

■ **Exemplo 4.49** Considere o poset $\langle \mathbb{N}, \leq \rangle$ o conjunto $X_{k_1, k_2} = \{x \in \mathbb{N} \mid k_1 < x < k_2\}$ é um conjunto que sempre possuirá majorantes e minorantes (a saber k_1 e k_2) logo ele é um conjunto limitado em $\langle \mathbb{N}, \leq \rangle$.

■ **Exemplo 4.50** Considere o poset $\langle \wp(A), \subseteq \rangle$ onde A é um conjunto qualquer, um conjunto $X \in \wp(A)$ sempre irá possuir minorante e majorante, que como dito em [2] são gerados pela interseção e união respectivamente. Assim qualquer $X \in \wp(A)$ sempre será um conjunto limitado em $\langle \wp(A), \subseteq \rangle$.

■ **Exemplo 4.51** Considere o poset esboçado na Figura 4.10 e o subconjunto $X = \{e, d, c, a, b\}$ tem-se que tal conjunto possui os majorantes e, g e f , mas não possui minorantes, assim X não é limitado.

Teorema 4.9 Seja $\langle A, \subseteq \rangle$ um poset e $X \subseteq A$. Se $X \neq \emptyset$, então $X \subseteq (X_\gamma)_\lambda$.

Demonstração. Dado $\langle A, \subseteq \rangle$ um poset e $X \subseteq A$, suponha que $X \neq \emptyset$, assim para todo $x \in X$ e cada $y \in X_\gamma$ tem-se que $y \subseteq x$, logo $x \in (X_\gamma)_\lambda$, portanto, $X \subseteq (X_\gamma)_\lambda$. \square

Teorema 4.10 Seja $\langle A, \subseteq \rangle$ um poset e $X \subseteq A$. Se $X \neq \emptyset$, então $X \subseteq (X_\lambda)_\gamma$.

Demonstração. Similar a prova do Teorema 4.9. \square

Teorema 4.11 Seja $\langle A, \subseteq \rangle$ um poset e $X \subseteq A$. Se $X \neq \emptyset$, então:

(i) $X_\lambda = ((X_\lambda)_\gamma)_\lambda$.

(ii) $X_\gamma = ((X_\gamma)_\lambda)_\gamma$.

Demonstração. Dado $\langle A, \subseteq \rangle$ um poset e $X \subseteq A$. Suponha que $X \neq \emptyset$, assim tem-se que:

(i) Pelo Teorema 4.10 segue que $X_\lambda \subseteq ((X_\gamma)_\lambda)_\gamma$, por outro lado, pelo Teorema 4.9 pode-se concluir que $((X_\lambda)_\gamma)_\lambda \subseteq X_\lambda$, portanto, pela Definição 1.9 tem-se que $X_\lambda = ((X_\lambda)_\gamma)_\lambda$.

(ii) Similar ao item anterior.

Completando assim a prova. \square

Teorema 4.12 Seja $\langle A, \subseteq \rangle$ um poset e $X, Y \subseteq A$. Se $X, Y \neq \emptyset$, então:

(i) $(X \cup Y)_\gamma = X_\gamma \cap Y_\gamma$.

(ii) $(X \cup Y)_\lambda = X_\lambda \cap Y_\lambda$.

Demonstração. Dado $\langle A, \subseteq \rangle$ um poset e $X \subseteq A$. Suponha que $X \neq \emptyset$, assim tem-se que:

(i) Desde que $X \subseteq X \cup Y$ e $Y \subseteq X \cup Y$ tem-se pelo Teorema 4.8 que $(X \cup Y)_\gamma \subseteq X_\gamma$ e $(X \cup Y)_\gamma \subseteq Y_\gamma$, consequentemente, $(X \cup Y)_\gamma \subseteq X_\gamma \cap Y_\gamma$. Por outro lado, para todo $x \in X_\gamma \cap Y_\gamma$ tem-se para todo

$y \in X$ que $x \leq y$ e para todo $z \in Y$ que $x \leq z$, assim $x \in (X \cup Y)_\gamma$ e assim $(X \cup Y)_\gamma \subseteq X_\gamma \cap Y_\gamma$ e, portanto, pela Definição 1.9 tem-se que $(X \cup Y)_\gamma = X_\gamma \cap Y_\gamma$.

(ii) Similar ao item anterior.

O que termina a prova. □

Definição 4.18 — Supremo. Seja $\langle A, \sqsubseteq \rangle$ um *poset* e $X \subseteq A$ o supremo de X (caso exista), denotado por $\sup(X)$, é o menor dos majorantes, em notação formal tem-se que $\sup(X) = \min(X_\lambda)$.

Como dito em [2, 21], uma forma de caracterização do supremo é através de suas propriedades inerentes, e isto é feito da seguinte forma, dado um *poset* $\langle A, \sqsubseteq \rangle$ tem-se que $\sup(X) = a$ se, e somente se:

1. $a \in A$.
2. Para todo $x \in X$ tem-se que $x \sqsubseteq a$.
3. Se $a' \in A$ e para todo $x \in X$ tem-se que $x \sqsubseteq a'$, então $a \sqsubseteq a'$.

Note que a propriedade (1) diz que o supremo (caso exista) é sempre um elemento do poset, já a propriedade (2) estabelece que o supremo deve ser um majorante e a propriedade (3) estabelece que o supremo deve ser a menor cota superior, isto é, o mínimo do conjunto dos majorantes.

Definição 4.19 — Ínfimo. Seja $\langle A, \sqsubseteq \rangle$ um *poset* e $X \subseteq A$ o ínfimo de X (caso exista), denotado por $\inf(X)$, é o maior dos minorantes, em notação formal tem-se que $\inf(X) = \max(X_\gamma)$.

Dualmente ao supremo como dito em [2, 21], uma forma de caracterização do ínfimo é através de suas propriedades inerentes, e isto é feito da seguinte forma, dado um *poset* $\langle A, \sqsubseteq \rangle$ tem-se que $\inf(X) = a$ se, e somente se:

1. $a \in A$.
2. Para todo $x \in X$ tem-se que $a \sqsubseteq x$.
3. Se $a' \in A$ e para todo $x \in X$ tem-se que $a' \sqsubseteq x$, então $a' \sqsubseteq a$.

Ou seja, a propriedade (1) diz que o ínfimo (caso exista) é sempre um elemento do poset, já a propriedade (2) estabelece que o ínfimo deve ser um minorante e a propriedade (3) estabelece que o ínfimo deve ser a maior cota inferior, isto é, o máximo do conjunto dos minorantes.

Observação 4.5 Como destacado em [106], os termos *least upper bound* (menor cota superior) e *join* são sinônimos de supremo, enquanto, que *greatest lower bound* (maior cota inferior) e *meet* são sinônimos de ínfimo.

■ **Exemplo 4.52** Considere o poset representado pela Figura 4.11 a seguir, o subconjunto $X = \{2, 3, 4, 5\}$ é tal que $\sup(X) = 7$ e $\inf(X) = 2$, por lado, para o subconjunto $Y = \{1, 0, 2, 4\}$ tem-se que $\sup(Y) = 3$ mas não existe um ínfimo de tal conjunto. Além disso, pegando o conjunto inteiro do poset tem-se que o conjunto possui supremo, a saber 7 mas não possui ínfimo.

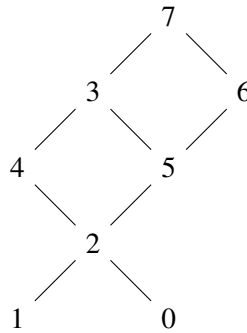


Figura 4.11: Diagrama de Hasse do poset do Exemplo 4.52.

■ **Exemplo 4.53** Considerando o poset $\langle \mathbb{Q}, \leq \rangle$ tem-se que o subconjunto $\{x \in \mathbb{Q} \mid x^2 \leq 2\}$ não possui supremo pois $\sqrt{2} \notin \mathbb{Q}$.

Teorema 4.13 Seja $\langle A, \sqsubseteq \rangle$ um poset e $X, Y \subseteq A$. Se $X \subseteq Y$ e além disso X e Y possuem supremo, então $\sup(X) \sqsubseteq \sup(Y)$.

Demonstração. Dado $\langle A, \sqsubseteq \rangle$ um poset e $X, Y \subseteq A$, assumamos $X \subseteq Y$ e que X e Y possuem supremo, assim pelo Teorema 4.8(i) tem-se que $Y_\lambda \subseteq X_\lambda$. Mas pelo Teorema 4.6(ii) tem-se que $\min(Y_\lambda) \sqsubseteq \min(X_\lambda)$, logo $\sup(X) \sqsubseteq \sup(Y)$. \square

Teorema 4.14 Seja $\langle A, \sqsubseteq \rangle$ um poset e $X, Y \subseteq A$. Se $X \subseteq Y$ e além disso X e Y possuem ínfimo, então $\inf(Y) \sqsubseteq \inf(X)$.

Demonstração. Dado $\langle A, \sqsubseteq \rangle$ um poset e $X, Y \subseteq A$, assumamos $X \subseteq Y$ e que X e Y possuem ínfimo, assim pelo Teorema 4.8(ii) tem-se que $Y_\lambda \subseteq X_\lambda$. Mas pelo Teorema 4.6(i) tem-se que $\max(Y_\gamma) \sqsubseteq \max(X_\gamma)$, logo $\inf(Y) \sqsubseteq \inf(X)$. \square

Teorema 4.15 Se $\langle A, \sqsubseteq \rangle$ é um poset, $X \subseteq A$ com $\sup(X_\gamma) = a$ e $a \in A$, então $\inf(X) = \sup(X_\gamma)$.

Demonstração. $\langle A, \sqsubseteq \rangle$ é um poset, $X \subseteq A$ com $\sup(X_\gamma) = a$ e $a \in A$, agora note que para qualquer $x \in X$ e $y \in X_\gamma$ tem-se que $y \sqsubseteq x$, e assim x é claramente um majorante de X_γ , ou seja, $x \in (X_\gamma)_\lambda$,

consequentemente, $a \sqsubseteq y$, uma vez que, $\sup(X_\gamma) = a$, assim a é um minorante de X , ou seja, $a \in X_\gamma$. Por outro lado, para qualquer $z \in X_\gamma$ tem-se que $z \sqsubseteq a$ e, portanto, $a = \inf(X)$. \square

Teorema 4.16 Se $\langle A, \sqsubseteq \rangle$ é um *poset*, $X \subseteq A$ com $\inf(X_\gamma) = a$ e $a \in A$, então $\sup(X) = \inf(X_\lambda)$.

Demonstração. Similar a demonstração do Teorema 4.15. \square

Observação 4.6 Quando um conjunto X satisfaz o Teorema 4.15 é dito que ele é inferiormente limitado, e quando ele satisfaz o Teorema 4.16 é dito que ele é superiormente limitado. Quando ele satisfaz os dois então ele satisfaz a Definição 4.17.

Teorema 4.17 Se $\langle A, \sqsubseteq \rangle$ é um *poset*, então as seguintes asserções são equivalentes:

- (1) Todo subconjunto não vazio superiormente limitado de A possui supremo.
- (2) Todo subconjunto não vazio inferiormente limitado de A possui ínfimo.

Demonstração. Dado $\langle A, \sqsubseteq \rangle$ um *poset*, tem-se que:

(1) \Rightarrow (2) Assuma que todo subconjunto não vazio superiormente limitado de A possui supremo, agora seja $X \subseteq A$ limitado inferiormente, ou seja, $X_\gamma \neq \emptyset$, agora obviamente cada $x \in X$ é tal que x é um majorante de X_γ , ou seja, $x \in (X_\gamma)_\lambda$ e, portanto, X_γ é superiormente limitado, logo pelo Teorema 4.16 existe o supremo do conjunto X_γ , consequentemente pelo Teorema 4.15 existe o ínfimo de X_γ .

(2) \Rightarrow (1) Similar ao item anterior. \square

4.6 Princípio da Boa Ordenação

Escrever depois...

4.7 Somas Ordinais, Ordem Produto e Ordem Lexicográfica

Escrever depois...

4.8 Questionário

■ **Exercício 4.1** Seja B o conjunto de todos os brasileiros e R uma relação definida sobre B como $x R y$ se, e somente se, x e y são nascidos no mesmo estado ou território do Brasil. Responda as questões a seguir.

- (a). Quantas classes de equivalência diferentes são determinadas por R ?
- (b). Escreva em notação compacta a classe de equivalência definida por Heitor Villa-Lobos.

(c). Responda se Getúlio Vargas \in [Heitor Villa-Lobos].

(d). Escreva em notação compacta a classe de equivalência definida por Leopoldo Nachbin.

■ **Exercício 4.2** Considerando o conjunto $A = \mathbb{N}^2$ demonstre que a relação $R = \{((a, b), (c, d)) \mid a + b = c + d\}$ é uma relação de equivalência sobre A^2 .

■ **Exercício 4.3** Considerando os conjuntos e as relações de equivalência a seguir sobre tais conjuntos, esboce os conjuntos quociente formado por cada relação.

(a). $A = \{a, b, c, d, e\}$ e $R = \{(a, a), (a, e), (b, b), (b, d), (c, c), (d, b), (d, d), (e, a), (e, e)\}$.

(b). $C = \{1, 2, 3, 4\}$ e $S = \{(1, 1), (2, 2), (3, 3), (4, 4), (2, 4), (4, 2)\}$.

(c). $A = \{x \in \mathbb{N} \mid x \leq 20\}$ e $K = \{(x, y) \in A^2 \mid (\exists k \in \mathbb{N})[\frac{(x-y)}{4} = k]\}$.

(d). $D = \{x \in \mathbb{Z} \mid -4 \leq x \leq 5\}$ e $T = \{(x, y) \in D^2 \mid (\exists k \in \mathbb{Z})[\frac{(x-y)}{3} = k]\}$.

(e). $E = \{(1, 3), (2, 4), (0, 3), (-4, -8), (3, 9), (-1, 5), (2, -4), (1, 5), (3, 6)\}$ e L é a relação definida sobre E da seguinte forma $(x_1, y_1) L (x_2, y_2)$ se, e somente se, $x_0 y_1 = x_1 y_0$.

(f). $A = \{-1, 0, 1\}$ e $M = \{(X, Y) \in \wp(A) \mid \text{num}(X) = \text{num}(Y)\}$ em que $\text{num}(D)$ é o número de elementos em D para qualquer $D \in \wp(A)$.

(g). $N = \{1, -2, 3\}$ e $M = \{(X, Y) \in \wp(N) \mid \sum_{x \in X} x = \sum_{y \in Y} y\}$ em que $\sum_{a \in D} a$ representa a soma de todos os elementos a pertencentes ao conjunto D para qualquer que seja o $D \in \wp(N)$.

(h). $D = \{x \in \mathbb{Z} \mid -5 \leq x \leq 5\}$ e $T = \{(x, y) \in D^2 \mid (\exists k \in \mathbb{Z})[\frac{(x^2 - y^2)}{3} = k]\}$.

(i). $F = \{x \in \mathbb{Z} \mid -4 \leq x \leq 5\}$ e $T = \{(x, y) \in F^2 \mid (\exists k \in \mathbb{Z})[\frac{(x^2 - y^2)}{3} = k]\}$.

(j). $B = \{00, 01, 02, 10, 11, 12, 20, 21, 22\}$ $R = \{(x, y) \in B^2 \mid sC(x) = sC(y)\}$ onde $sC(x)$ denotada a soma dos caracteres para qualquer $x \in B$.

■ **Exercício 4.4** Seja A o conjunto de todos os estudantes do seu campus demonstre que a relação T definida sobre A como “ $x T y$ se, e somente se, x é do mesmo curso que y ” é uma relação de equivalência.

■ **Exercício 4.5** Seja $A = \{a, b, c, d\}$ e $R = \{(a, a), (a, b), (b, a), (b, b), (c, d), (d, c), (d, d), (c, c)\}$. Prove que R é uma relação de equivalência sobre A e esboce também o espaço quociente A/R .

■ **Exercício 4.6** Sendo R uma relação de equivalência sobre X é correto afirmar que $\text{dom}(R) = X$?

■ **Exercício 4.7** Suponha que R e S são duas relações de equivalência sobre um conjunto A . Prove que $R \cap S$ é também uma relação de equivalência sobre A .

■ **Exercício 4.8** Demonstre que a relação $R = \{(x, y) \in \mathbb{Z}^2 \mid x - y \in \mathbb{Z}\}$ é uma relação de equivalência sobre o conjunto \mathbb{Z} .

■ **Exercício 4.9** Considerando para todo $n \in \mathbb{Z}$ realize o que é solicitado a seguir.

- (a). Demonstre que $R_n = \{(x, y) \in \mathbb{Z}^2 \mid \frac{x-y}{n} = k\}$ é uma relação de equivalência.
- (b). Para o caso particular de $n = 2$ esboce usando notação compacta a classe $[1]$ com respeito a relação R_n .
- (c). Para o caso particular de $n = 5$ esboce usando notação compacta a classe $[-4]$ com respeito a relação R_n .

■ **Exercício 4.10** Sejam $A = \{1, 2\}$ e $B = \{3, 4\}$ dois elementos de uma partição de $\{1, 2, 3, 4\}$ liste os pares ordenados que compõem a relação de equivalência que definiu a partição que contém A e B .

■ **Exercício 4.11** Demonstre que a relação $P = \{(x, y) \in \mathbb{Z}^2 \mid (\exists k \in \mathbb{Z})[x^2 - y^2 = 2k]\}$ é uma relação de equivalência sobre o conjunto \mathbb{Z} e usando notação compacta a classe escreva o espaço quociente gerado por tal relação.

■ **Exercício 4.12** Seja $S = \mathbb{N}^2$. Demonstre que a relação $P = \{((x, y), (w, z)) \in S^2 \mid x = z\}$ é uma relação de equivalência sobre o conjunto S e usando notação compacta a classe escreva o espaço quociente gerado por tal relação.

■ **Exercício 4.13** Seja $S = \mathbb{Z}^2$. Verifique se a relação $T = \{((x, y), (w, z)) \in S^2 \mid x - y = w - z\}$ é uma relação de equivalência sobre o conjunto S , se for então usando notação compacta a classe escreva o espaço quociente gerado por tal relação.

■ **Exercício 4.14** Seja M o conjunto de todas as matrizes simétricas, demonstre que a relação $m_1 K m_2 \iff m_1 = m_2^t$, onde m^t representada a matriz transposta de m com $m \in M$, é uma relação de equivalência.

■ **Exercício 4.15** Verifique se as seguintes estruturas são *posets*.

- (a). $\langle \mathbb{N}, \alpha \rangle$ onde $\alpha = \{(x, y) \in \mathbb{N}^2 \mid \text{o resto da divisão de } y \text{ por } x \text{ é igual a } 1\}$.
- (b). $\langle \mathbb{Z}, \alpha \rangle$ onde $\alpha = \{(x, y) \in \mathbb{Z}^2 \mid y = x + 3\}$.
- (c). $\langle \mathbb{N}, \alpha \rangle$ onde $\alpha = \{(x, y) \in \mathbb{N}^2 \mid y - x \in \mathbb{N}\}$.
- (d). $\langle \mathbb{Z}, \alpha \rangle$ onde $\alpha = \{(x, y) \in \mathbb{Z}^2 \mid 2y - x \in \mathbb{N}\}$.

■ **Exercício 4.16** Para cada *poset* a seguir desenhe o diagrama um Hasse para o mesmo.

- (a). $\langle \{1, 2, 3, 4, 5, 6, 10, 15, 20, 60\}, M \rangle$ onde $x M y$ se, e somente se, $\frac{60}{x} = y$.

- (b). $\langle \{1, 2, 3, 4, 5, 6, 7, 8, 9, 10\}, D \rangle$ onde $x D y$ se, e somente se, $\frac{y}{x} = k$ tal que $k \in \{1, 2, 3, 4, 5, 6, 7, 8, 9, 10\}$.
- (c). $\langle \{1, 2, 3\}, \{(1, 1), (1, 2), (2, 2), (3, 3), (1, 3), (2, 3)\} \rangle$.
- (d). $\langle \{a, b, c, d\}, \{(d, d), (b, b), (a, a), (c, c), (d, c), (d, a), (a, c)\} \rangle$.

■ **Exercício 4.17** Para cada um dos diagrama na Figura 4.12, construa sua estrutura de *poset*.

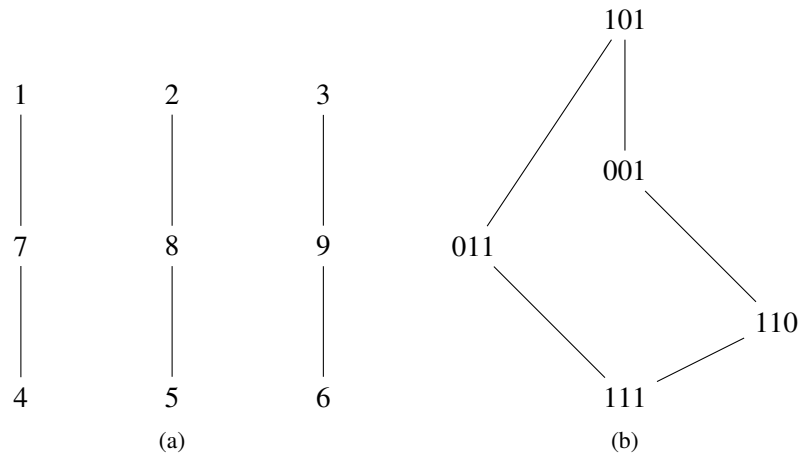


Figura 4.12: Diagramas de Hasse para a questão 4.16.

■ **Exercício 4.18** Considere o conjunto $X = \{a, b, c, d, e, f, g, h, i\}$ e R a relação de ordem parcial representada pela Figura 4.13, calcule o máximo, mínimo, elementos maximais, elementos minimais, conjuntos dos majorantes, conjuntos dos minorantes, supremo e ínfimo dos seguintes subconjuntos de X :

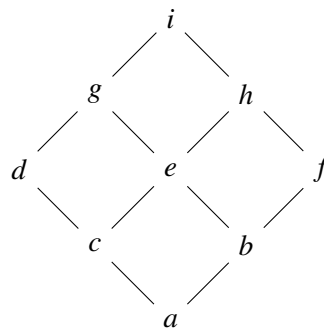


Figura 4.13: Legenda

- (a). $A_1 = \{d, e, b\}$.
- (b). $A_2 = \{a, c, e, b\}$.
- (c). $A_3 = \{g, e, h\}$.
- (d). $A_4 = \{d, e, f\}$.

(e). $A_5 = \{a, c, f\}$.

(f). $A_6 = \{a, e\}$.

(g). $A_7 = \{i\}$.

(h). $A_8 = \{d, f, a, i\}$.

(i). $A_9 = \{g, c, b, h\}$.

(j). $A_{10} = \{a, c, f, e\}$.

■ **Exercício 4.19** Considere o conjunto $X = \{0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10\}$ e R a relação de ordem parcial representada pela Figura 4.14, calcule o máximo, mínimo, elementos maximais, elementos minimais, conjuntos dos majorantes, conjuntos dos minorantes, supremo e ínfimo dos seguintes subconjuntos de X :

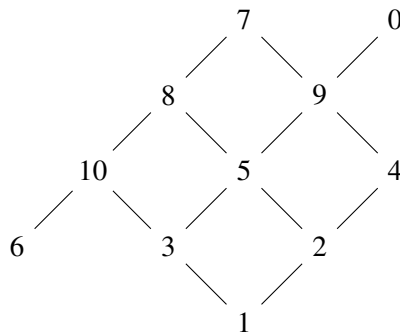


Figura 4.14: Legenda

(a). $A_1 = \{10, 2, 9\}$.

(b). $A_2 = \{0, 6, 4\}$.

(c). $A_3 = \{10, 3, 6, 7\}$.

(d). $A_4 = \{6, 7, 0, 1\}$.

(e). $A_5 = \{9, 8, 3, 2\}$.

(f). $A_6 = \{5, 8, 2\}$.

(g). $A_7 = \{3, 9, 5, 2\}$.

(h). $A_8 = \{10, 1, 4, 7\}$.

(i). $A_9 = \{0, 9, 4, 8\}$.

(j). $A_{10} = \{6, 10, 9, 3, 0, 7\}$.

■ **Exercício 4.20** Considere um conjunto A qualquer e uma relação binária R sobre A demonstre que R será uma relação de ordem parcial e de equivalência simultaneamente se, e somente se, R for a relação de identidade em A .

■ **Exercício 4.21** Dê exemplo de:

- (a). Duas relação de ordem parcial sobre um conjunto A , tal que a união das duas não seja uma ordem parcial sobre A .
- (b). Duas relação de ordem parcial sobre um conjunto A , tal que a composição das duas não seja uma ordem parcial sobre A .

■ **Exercício 4.22** Demostre que se o *poset* $\langle A, \sqsubseteq \rangle$ não tem menor elemento, ou seja, não existe $\inf(A)$, então existem no mínimo $x, y \in A$ que são minimais de A .

■ **Exercício 4.23** Demostre que se o *poset* $\langle A, \sqsubseteq \rangle$ não tem maior elemento, ou seja, não existe $\sup(A)$, então existem no mínimo $x, y \in A$ que são maximais de A .

■ **Exercício 4.24** Demostre que se o *poset* $\langle A, \sqsubseteq \rangle$ é tal que \sqsubseteq é uma ordem total, então para todo $x \in A$ existe um único par de elementos $y_1, y_2 \in A$ tal que $y_1 \sqsubseteq x$ e $x \sqsubseteq y_2$.

■ **Exercício 4.25** Considerando o conjunto $P = \{t, u, v, w, x, y, z\}$ responda as questões a seguir.

- (a). Quantas relações de ordem parcial podem ser definida sobre P ?
- (b). Quantas relações de ordem parcial podem ser definida sobre P de forma que v seja “menor” que w ?
- (c). Quantas relações de ordem parcial podem ser definida sobre P de forma que z seja “menor” que y ?

Capítulo 5

Funções

“Nada que vale a pena é fácil”.

Eric Cartman, South Park

“O ego é o pior inimigo do Eu, mas o Eu é o melhor amigo do ego... O ego é um péssimo senhor, mas é um ótimo servidor”.

Bhagavad Gita

5.1 Conceitos, Definições e Nomenclaturas

Após o conceito importantíssimo de conjunto o componente mais importante na matemática é provavelmente a noção de função, o autor deste manuscrito não hesita em afirmar que você leitor com certeza já teve contato com a ideia de função, seja em seus cursos do primário, secundário ou mesmo mais recentemente em suas disciplinas de nível superior tais como cálculo diferencial e integral ou alguma disciplina de física.

Dado este encontra anterior do leitor sobre o assunto o autor fica confortável a pedir que leitor faça uma pequena pausa e tente lembrar de seus cursos anteriores e respondo para si mesmo ao questionamento: **o que é uma função?**

Bem, para muitos físicos, estatísticos e alguns matemáticos (não todos¹), uma função é vista meramente como sendo um mapeamento (ou transformação) entre os elementos de dois conjuntos [2]. Por outro lado, em obras tais como [35, 62, 63, 64, 86] uma função é vista como uma caso particular de relação entre dois conjuntos, ou seja, em última análise para esse grupo de pessoas uma função é exatamente

¹Um visão interessante é aquela apontada em [57], que descreve uma função como sendo um objeto com quatro descrições simultâneas: uma algébrica, uma numérica, uma gráfica e uma descritiva (ou em palavras).

um conjunto². Já em [96, 106] é apresentada uma visão mais mecanicista da ideia de função, essa visão captura a ideia de função enquanto uma máquina³ (ou caixa preta) que transforma as entradas (*inputs*) em saídas (*outputs*), essa visão é ilustrada pela Figura 5.1.

Há também a ideia de função como sendo uma estrutura [40], com componentes bem estabelecidos. Essa visão é capaz (como será mostrado a seguir) de capturar todas as outras ideias de função. Neste manuscrito a formalização da ideia de função como sendo uma estrutura será apresentada de forma gradual traçando paralelos com as linguagens de programação que possuam um sistema de tipos, isso será adotado para tornar o texto mais didático e interessante ao leitor de computação, além disso, irá aproximar os tópicos teóricos (as funções) dos tópicos práticos (programação) cujo leitor desse manuscrito naturalmente tem contato e provável interesse. Entretanto, essa forma de apresentação não será menos rigorosa que outras fontes bibliográficas, na verdade será o oposto, o texto aqui apresentado tende a ser mais preciso e detalhado que a apresentação rasa feita em [40], por exemplo.

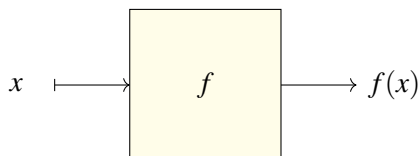


Figura 5.1: Visão de uma função de uma variável enquanto uma máquina (ou caixa preta).

Este manuscrito irá iniciar o estudo sobre funções apresentado ao leitor a ideia básica de assinatura de função, isto é, a seguir será apresentado a estrutura sintática que descreve as funções, ou seja, o componente descritivo mencionado em [57].

Definição 5.1 — Assinatura de Função. Sejam A e B conjuntos a assinatura da função de A em B nomeada como f corresponde a uma palavra da forma $f : A \rightarrow B$.

Note que a Definição 5.1 permite facilmente deduzir que em qualquer função existem três componentes básicos, sendo eles: um nome (rótulo ou símbolo funcional), um conjunto de partida e um conjunto de chegada. Por convenção, o nome de uma função deve ser sempre iniciado por caracteres latinos, no caso de usar índices apenas o último caractere do nome deve ser indexado.

■ **Exemplo 5.1** São exemplos de assinaturas de funções:

(a) $g : \mathbb{N} \rightarrow \mathbb{Z}$.

(b) $\text{sqrt} : \mathbb{R} \rightarrow \mathbb{R}$.

(c) $k_1 : A \times B \rightarrow \mathbb{C}$.

²Para melhor entender essa visão talvez o leitor deva revistar os Capítulos 1 e 3 e revisar as definições apresentadas nos mesmos.

³Para cientistas da computação os termos máquina e programa são sinônimos.

(d) $loc_2 : D \rightarrow \mathbb{R} \times \{0, 1\}$.

(e) $min : \mathbb{R}^n \rightarrow \mathbb{R}$.

(f) $BUSCA : \mathbb{Z}^n \times \mathbb{Z} \rightarrow \{0, 1\}$.

(g) $BUSCA : \mathbb{R}^n \times \mathbb{R} \rightarrow \{0, 1\}$.


Observação 5.1 O leitor deve observar que apesar das assinaturas nos itens (f) e (g) do Exemplo 5.1 terem o mesmo nome, elas não são iguais, pois diferem no conjunto de partida, e assim não são a mesma assinatura. Esse tipo de cenário na computação recebe o nome de sobrecarga, isto é, o símbolo funcional *BUSCA* está sobrecarregado para identificar duas funções distintas.

Diversas linguagens de programação tais como C, C++ e Java apresentam a possibilidade de definir assinaturas de funções, na linguagem C por exemplo as assinaturas de funções que compõem uma biblioteca são reunidas em um arquivo de *header*, isto é, um arquivo com a extensão .h, para mais detalhes consulte [38], a seguir é exemplificado um arquivo de header.

```
1 /* Assinaturas */
2 int reverse(int x);
3 unsigned int strlen(char *x);
4 int strcmp(char *x, char *y);
5 char *strcpy(char *y, char *x);
```

Figura 5.2: Exemplo de um arquivo .h contendo assinaturas na linguagem C.

Um conceito indiretamente esboçado pela ideia de assinatura de função, é o de tipagem da função, sempre que a assinatura é da forma $f : A \rightarrow B$ pode-se dizer que f é uma função do tipo “A em B”, ou mesmo que “ f é um tipo seta de A para B”, a noção de “tipo seta” é uma nomenclatura diretamente ligado a ideia de Teoria das categorias.

 **Nota 5.1** Em geral programadores focados em linguagens imperativas tais como C (ou C++, Java e etc.) tende a errar a seguinte questão: “Qual é o tipo da função reverse (definida na linha 2 da Figura 5.2)?” A critério de esclarecimento tal função é do tipo seta de int em int, ou seja, matematicamente sua assinatura seria da forma, reverse: $\text{int} \rightarrow \text{int}$.

Como dito anteriormente uma função pode ser vista como uma máquina que transforma entradas em saídas, mas note que para que isso aconteça a máquina deve de alguma forma realizar ações sobre a entrada, ou seja, a máquina deve “operar” sobre a entrada. Esse conceito de como a máquina deve operar sobre as entradas é descrito por uma propriedade P que define uma relação de mapeamento⁴

⁴Alguns textos usam a nomenclatura lei de formação, ver por exemplo [21].

Definição 5.2 — Relação de mapeamento. Dado dois conjuntos A e B e seja $x \in A$ e $y \in B$ a relação de mapeamento definida por uma propriedade P corresponde ao seguinte conjunto

$$\varepsilon = \{(x, y) \mid P\}$$

tal que ε satisfaz a seguinte condição: se $(x, y_1), (x, y_2) \in \varepsilon$, então $y_1 = y_2$.

Note que a Definição 5.2 apenas descreve que as entradas (as variáveis) x_1, \dots, x_n se relacionam como uma única saída y por uma certa propriedade P .

■ **Exemplo 5.2** São exemplos de relações de construção:

- (a) $\{(x, y) \in \mathbb{R}^2 \mid y = \log_2(x + 1)\}$
- (b) $\{(w_1 w_2 w_3 \cdots w_m, y) \in E^2 \mid y = w_3 \cdots w_m w_1 w_2\}$ onde E é o conjunto de todas as palavras do português com três letras ou mais e w_i representa o i -ésimo símbolo de uma palavra w .
- (c) $\{(x, y) \in \mathbb{N}^2 \mid y = 14\}$.
- (d) $\{(x_1, x_2, x_3, y) \in \mathbb{R} \times \mathbb{R} \times \mathbb{R}_+^* \times \mathbb{R} \mid y = \sqrt[3]{\frac{1}{2}x_1 + x_2}\}$.

Não são exemplos de relações de construção:

- (e) $\{(x, y) \in N_P \times I_P\}$ onde N_P é o conjunto de todos os nomes de pessoas e I_P é o conjunto de naturais que representam idades, note que em tal relação é permitido que $(\text{Fátima}, 10), (\text{Fátima}, 55)$ estejam nesse conjunto, portanto, esse conjunto não satisfaz a Definição 5.2.
- (f) $\{(x, y) \in \mathbb{R} \mid \sqrt{y} = x\}$, note que $(5, 25)$ e $(-5, 25)$ pertence a tal conjunto e, portanto, esse conjunto não satisfaz a Definição 5.2.

Observação 5.2 O item (c) do Exemplo 5.2 é um caso interessante, a propriedade da relação descreve que para qualquer entrada x a mesma sempre irá devolver como resposta um $y = 14$, ou seja, para todo x tem-se que o par $(x, 14)$ está na relação. Esse tipo de relação de mapeamento será aqui chamada de **relação de mapeamento constante**.

Agora que foram apresentados estes conceitos fundamentais pode-se continuar o desenvolvimento deste texto com a formalização da ideia de função.

Definição 5.3 — Função. Dado uma assinatura $f : A \rightarrow B$ e seja $\varepsilon \subseteq A \times B$ uma relação de mapeamento, a estrutura $\langle f : A \rightarrow B, \varepsilon \rangle$ é uma função.

■ **Exemplo 5.3** São exemplos de funções:

- (a) $\langle \text{dob} : \mathbb{N} \rightarrow \mathbb{N}, \{(x, y) \in \mathbb{N}^2 \mid y = 2x\} \rangle$.

- (b) $\langle \text{mul} : \mathbb{R} \times \mathbb{R} \rightarrow \mathbb{N}, \{(x, y), z\} \in \mathbb{R}^2 \times \mathbb{R} \mid z = xy\} \rangle$.
- (c) $\langle \text{sqrt} : \mathbb{N} \rightarrow \mathbb{N}, \{(x, y) \in \mathbb{N}^2 \mid y^2 = x\} \rangle$.
- (d) $\langle \text{one} : \mathbb{Z}^2 \rightarrow \{1\}, \{(x, y), 1\} \in \mathbb{Z}^2 \times \{1\} \mid 1 = x + y\} \rangle$.
- (e) $\langle \text{sign} : \mathbb{R} \rightarrow \{0, 1\}, \{(x, y) \in \mathbb{R} \times \{0, 1\} \mid y = 1 \text{ sempre que } x > 0.5 \text{ e } y = 0 \text{ se } x \leq 0.5\} \rangle$.


De forma similar a apresentação feita em [21] aqui será usado a própria relação de mapeamento para definir as noções de domínio e imagem de uma função.

Definição 5.4 — Domínio e Imagem de função. Seja $\langle f : A \rightarrow B, \varepsilon \rangle$ uma função o domínio e a imagem de f , denotados respectivamente por $\text{dom}(f)$ e $\text{ima}(f)$, corresponde exatamente ao domínio e a imagem de ε , ou seja, $\text{dom}(f) = \text{Dom}(\varepsilon)$ e $\text{ima}(f) = \text{Ima}(\varepsilon)$.

■ **Exemplo 5.4** Considere a função $\langle k : \mathbb{R} \rightarrow \mathbb{R}, \{(x, y) \in \mathbb{R}^2 \mid y = \sqrt{x}\} \rangle$ tem-se que $\text{dom}(k) = \text{ima}(k) = \mathbb{R}^+$, pois claramente $(x, y) \in \varepsilon$ se, e somente se, $(x, y) = (x, \sqrt{x})$, agora obviamente \sqrt{x} só existe para $x \geq 0$, logo $\text{Dom}(\varepsilon) = \mathbb{R}^+$, em contra partida para qualquer seja $\sqrt{x} \in \mathbb{R}$ tem-se que $\sqrt{x} \geq 0$, consequentemente, $\text{Ima}(\varepsilon) = \mathbb{R}^+$.

■ **Exemplo 5.5** Considere a função $\langle \text{prev} : \mathbb{N} \rightarrow \mathbb{N}, \{(x, y) \in \mathbb{N}^2 \mid y = x - 1\} \rangle$, note que $(x, y) \in \varepsilon$ se, e somente se, $(x, y) = (x, x - 1)$, mas $x - 1 \in \mathbb{N}$ apenas se $x > 0$, portanto, tem-se que $\text{dom}(\text{prev}) = \mathbb{N}_*$. Por outro lado, para qualquer $x > 0$ tem-se que $x - 1 \in \mathbb{N}$ e, portanto, não é difícil verificar que $\text{Ima}(\varepsilon) = \mathbb{N}$, consequentemente, $\text{Ima}(\text{prev}) = \mathbb{N}$.

Agora é comum ao estudar funções como destacado em [21], não ficar declarando a estrutura da função o tempo inteiro, em vez disso, em geral é descrita a assinatura da função junto com o açúcar sintático detalhado a seguir.

 **Nota 5.2 — Açúcar sintático.** Seja $\langle f : A \rightarrow B, \varepsilon \rangle$ uma função, $f(x)$ é um açúcar sintático para dizer que f está recebendo como entrada x , se escreve $f(x) = y$ como açúcar sintático da frase: “ao calcular f com entrada x é gerado y como saída”, mas para isso necessário que $(x, y) \in \varepsilon$. Agora observe que quando a relação ε de uma função descreve y a partir de uma igualdade da forma $y = E$, em que E é uma expressão válida, pode-se em vez de, fazer $f(x) = y$ escreve diretamente $f(x) = E$, e desde que $y = E$ a expressão $f(x) = E$ é um refinamento do açúcar sintático.

■ **Exemplo 5.6** Usando as ideias do açúcar sintático descrito na Nota 5.2 tem-se que:

- (a) A função $\langle \text{dob} : \mathbb{N} \rightarrow \mathbb{N}, \{(x, y) \in \mathbb{N}^2 \mid y = 2x\} \rangle$ pode simplesmente ser escrita usando sua assinatura $\text{dob} : \mathbb{N} \rightarrow \mathbb{N}$ e dizendo que $\text{dob}(x) = 2x$.
- (b) A função $\langle \text{pot} : \mathbb{R} \times \mathbb{N} \rightarrow \mathbb{R}, \{(x, y), z\} \in (\mathbb{R} \times \mathbb{N}) \times \mathbb{R} \mid z = x^y\} \rangle$ pode simplesmente ser escrita usando sua assinatura $\text{pot} : \mathbb{R} \times \mathbb{N} \rightarrow \mathbb{R}$ e dizendo que $\text{pot}(x, y) = x^y$.

Observação 5.3 Deste ponto em diante sempre que for possível será usado apenas a assinatura para se referir a uma função.

Note que se f é uma função com assinatura $f : A \rightarrow B$, isto é, f é uma função do tipo seta de A em B , e x um objeto do tipo A ⁵ pode-se pensar em uma regra capaz de deduzir o tipo da saída de $f(x)$, tal regra poderia ser nomeada como $dType$ e poderia ser escrita como:

$$\frac{x : A \quad f : A \rightarrow B}{f(x) : B} dType$$

essa regra não é algo novo criado nesse manuscrito, na verdade a mesma em um certo ponto de vista é uma versão da famosa regra de *modus ponens* (ver Capítulo 7), a existência de tal regra é uma manifestação da profunda conexão que existe entre lógica e computação. Tal conexão é conhecida como isomorfismo de Curry–Howard, e é um aspecto fundamental em áreas como teoria dos tipos [83, 105], teoria da prova [83, 98], λ -cálculo [9, 10, 17] e programação funcional [105, 106].

Agora note que a Definição 5.4 não impõe de forma alguma que todos os elementos do conjunto de partida de uma função estejam no seu domínio, tendo isso em mente pode-se classificar funções em duas categorias definidas a seguir.

Definição 5.5 — Funções totais e parciais. Uma função $f : A \rightarrow B$ é dita ser total sempre que $dom(f) = A$ e será dita parcial sempre que $dom(f) \subseteq A$.

```

1 int sqroot(int x){
2   int y = 0;
3   while(y*y != x){
4     y = y + 1;
5   }
6   return y;
7 }
```

Figura 5.3: Código da implementação da função no item (c) do Exemplo 5.3 escrito na linguagem C.

Observação 5.4 Obviamente a linha 1 no código esboçado na Figura 5.3 claramente fez respeito a assinatura da função, já o conteúdo das linhas 2 até 6 são responsáveis por fazer uma “busca” de um y tal que dado um x de entrada tem-se que $f(x) = y$.

Agora note um fato sobre a função *sqroot* e sua implementação no código esboçado na Figura 5.3, quando tal função recebe

⁵Em teoria dos tipos [83, 105] se A é um tipo e x é um objeto do tipo A pode-se escrever que $x : A$, isto é algo semelhante a teoria dos conjuntos ao dizer que $x \in A$.

Parte II

Álgebra Universal

Parte III

Lógica

Capítulo 6

Introdução à Lógica

6.1 O que é Lógica?

Antes de apresentar uma descrição histórica da lógica, este texto começa pela árdua tarefa de apresentar de forma sucinta uma resposta para a pergunta, “**o que é a lógica?**”. Como dito em [14, 26], a palavra lógica e suas derivações são familiares a quase todas (se não todas) as pessoas, de fato, é comum durante o cotidiano do dia a dia as pessoas recorrerem ao uso do termo lógica ou de seus derivados, sendo que na maioria das vezes seu uso está ligada à ideia de obviedade (ou certeza), por exemplo nas frases:

- (a) É lógico que vou na festa.
- (b) É lógico que ciência da computação é um curso difícil.
- (c) Logicamente o Vasco não pode ganhar o título da primeira divisão nacional em 2021.
- (d) Logicamente se eu tomar banho, vou ter que me molhar.

Essa forma de usar os derivados da palavra lógica enquanto entidades para transmissão de certeza pode ser usada como gatilho “fácil e preguiçoso” para enunciar que a lógica se trata de uma ciência (ou disciplina) acerca das certezas sobre os fatos do mundo material.

Existe outra resposta comumente encontrada na literatura acadêmica (ver [1, 14, 70]) para o que seria a lógica, esta segunda alternativa de resposta descreve a lógica como sendo um mecanismo utilizado durante o raciocínio estruturado e correto¹, isto é, uma ferramenta do raciocínio que possibilita a inferência de conclusões a partir de premissas [1, 26, 47, 57], por exemplo, dado as premissas:

- (a) Toda quinta-feira é servido peixe no almoço.
- (b) Hoje é quarta-feira.

¹Uma visão semelhante a esta é descrita em [67], que diz que a lógica está preocupada com a avaliação de argumentos e a separação dos argumentos bons dos ruins.

O raciocínio munido da “ferramenta de inferência” contida na lógica permite deduzir a afirmação: **Amanhã será servido peixe no almoço**, como conclusão. Note que esta segunda resposta estabelece que a lógica é um tipo de procedimento mental capaz de transformar informações de entrada (as premissas) em informações de saída (a conclusão).

Essas duas formas de encarar a lógica não estão totalmente erradas, entretanto, também não exibem de forma completa o real significado do que seria a lógica em si. Uma terceira resposta para a pergunta “O que é a lógica?” aparece na edição de 1953 da Encyclopædia Britannica na seguinte forma: “*Logic is the systematic study of the structure of propositions and of the general conditions of valid inference by a method which abstracts from the content or matter of the propositions and deals only with their logical form*”. Note que essa resposta utiliza-se de autorreferência², pois a mesma tenta definir o que é a lógica em função do termo “forma lógica”.

Apesar dessa definição recursiva³, a resposta da Encyclopædia Britannica apresenta duas características muito marcantes para a apresentação da lógica enquanto ciência (ou disciplina) em nossos dias atuais. A primeira característica é a validade das afirmações derivadas (ou concluídas) pelo mecanismos de inferência. A segunda característica é a importância da forma de representação (a escrita) dos termos lógicos.

A validade remonta a ideia de um significado dual (verdadeiro e falso) para as afirmações, ou seja, fornece indícios da existência de interpretações das afirmações, e isto significa que existem diferentes significados para as afirmações a depender de um fator que pode ser chamado de contexto, por exemplo considere a seguinte afirmação:

“O atual presidente americano é um democrata”.

Note que o contexto temporal muda drasticamente o valor lógico interpretativo (semântico) dessa afirmação pois no ano de 2021 essa afirmação era interpretada como verdadeira, porém no ano de 2019 a mesma era falsa. Assim os valores interpretativos (semânticos) dentro do universo da lógica não são imutáveis, isto é, os valores das interpretações da lógica são passíveis de mudança a depender do contexto.

Dado então estes componentes sintáticos e semânticos pode-se concluir a partir das definições linguísticas que a **lógica é uma linguagem**, entretanto, vale salientar que não é uma linguagem natural como o português, como será visto nos próximos capítulos a lógica é uma linguagem formal [15], no sentido de que todas as construções linguísticas possuem uma forma precisa e sem ambiguidade determinada por uma gramática geradora [49, 61], pode-se inclusive estabelecer que a lógica é a linguagem

²Autorreferência é um fenômeno que ocorre nas linguagens naturais e formais, tal fenômeno consiste de uma oração ou fórmula que refere-se a si mesma.

³Em matemática a ideia de definição recursiva está ligada a ideia de uma estrutura que apresenta autorreferência.

da ciência da inferência racional, ou seja, a linguagem usada para representar argumentos, inferência e conclusões sobre um certo universo do discurso.

6.2 Um Pouco de História

A história do desenvolvimento da lógica remonta até a Grécia antiga e a nomes como: Aristóteles (384-322 a.C.), Sócrates (469-399 a.C.), Zenão de Eléia (490-420 a.C.), Parmenides (515-445 a.C.), Platão (428-347 a.C.), Eudemos de Rodas (350-290 a.C.), Teofrastus de Lesbos (378-287 a.C.), Euclides de Megara (435-365 a.C.) e Eubulides de Mileto⁴ (384-322 a.C.). De fato o nome lógica vem do termo grego *logike*, cunhado por Alexandre de Afrodísias no fim do século II depois de Cristo. Como explicado em [1], os mais antigos registros sobre o estudo da lógica como uma disciplina (ciência) são encontrados exatamente na obra de Aristóteles intitulada como “T da metafísica”. Todavia, após seu desenvolvimento inicial dado pelos gregos antigos, a lógica permaneceu quase que intocada⁵ por mais de 1800 anos.

Os primeiros a profanar a santidade da lógica de forma contundente, abalando as estruturas da ideia que a lógica era uma ciência completa⁶ foram os matemáticos George Boole (1815-1864) e Augustus De Morgan (1806-1871), que introduziram a moderna ideia da lógica como uma ciência simbólica, isto é, eles semearam os conceitos iniciais que depois iriam convergir para as ideias da lógica enquanto linguagem formal apresentadas pelo matemático e filósofo alemão Gottlob Frege (1848-1925), que via a lógica como uma linguagem, que continha em seu interior todo o rigor da matemática.

Ainda no século XIX os maiores defensores das ideias de Frege, os britânicos Alfred Whitehead (1861-1947) e Bertrand Russell (1872-1970), usaram muitas de suas ideias e sua linguagem na publicação monumental em três volumes intitulada “*Principia Mathematica*” [5], que é ainda hoje considerada por muitos o maior tratado matemático do século XIX. Como dito em [14], outro influenciado por Frege que apresentou importantes contribuições foi filósofo austríaco Ludwig Wittgenstein (1889-1951), que em seu “*Tractatus Logico-Philosophicus*” apresentou pela primeira vez a lógica proposicional através das tabelas verdade. Muitos autores, como é o caso de [1], consideram que a lógica moderna se iniciou verdadeiramente com a publicação do *Principia*, de fato, alguns usam exatamente a visão de Whitehead que diz: “A lógica atual está para a lógica aristotélica como a matemática moderna está para a aritmética das tribos primitivas”.

Outra vertente emergente na lógica do século XIX era aquela apoiada puramente por interesses matemáticos, isto é, a visão da lógica não apenas como linguagem, mas também como um objeto algebrizável (um cálculo). Tal escola de lógica encontra alguns de seus expoentes nos nomes de: Erns Zermelo⁷

⁴Creditado como criador do paradoxo do mentiroso.

⁵Aqui não está sendo levando em conta as tentativas de Gottfried Wilhelm Leibniz (1646-1716) de desenvolver uma linguagem universal através da precisão matemática.

⁶No sentido de que não havia nada novo a se fazer, estudar ou provar.

⁷Zermelo junto com Fraenkel desenvolveu o sistema formal hoje conhecido como teoria axiomática dos conjuntos.

(1871-1953), Thoralf Skolem (1887-1963), Ludwig Fraenkel-Conrad (1910-1999), John von Neumann (1903-1957), Arend Heyting (1898-1980) entre outros. Uma das grandes contribuições feitas por essa escola foi incluir uma formulação explícita e precisa das regras de inferência no desenvolvimento de sistemas axiomáticos.

Uma ramificação desta escola “matemática” ganhou força na Polônia sobre a tutela e liderança do lógico e filósofo Jan Łukasiewicz (1878-1956), o foco da escola polonesa era como dito em [14], analisar os sistemas axiomáticos da lógica proposicional, lógica modal e das álgebras booleanas. Foi esta escola que primeiro considerou interpretações alternativas da linguagem (da lógica) e questões da meta-lógica tais como: consistência, corretude e completude. Por fim, foi na escola polonesa que houve pela primeira vez duas visões separadas sobre a lógica, uma em que a lógica era vista puramente como uma linguagem, e a segunda visão que via a lógica puramente como um cálculo [14].

Instigado pelo problema número dois da lista de Hilbert o jovem matemático e lógico austríaco Kurt Gödel (1906-1978) fez grandes contribuições para a lógica, inicialmente ele provou o teorema da completude para a lógica de primeira ordem em sua tese de doutorado em 1929, tal resultado estabelece que uma fórmula de primeira ordem é dedutível se e somente se ela é universalmente válida [14]. Outra contribuição monumental de Gödel são seus teoremas da incompletude [42], em especial o primeiro que deu uma resposta negativa ao problema número dois da lista Hilbert, de forma sucinta o resultado de Gödel estabelece que não pode haver uma sistematização completa da Aritmética [1].

Outros contemporâneos de Gödel também contribuíram fortemente para a lógica, Alfred Tarski (1901-1983) foi o responsável pela matematização do conceito de verdade como correspondência [1, 104], já o francês Jacques Herbrand (1908-1931) introduziu as funções recursivas e apresentou os resultados hoje chamados de teoria de Herbrand. Entre os resultados de Herbrand se encontra o teorema que relaciona um conjunto insatisfatível de fórmulas da lógica de primeira ordem com um conjunto insatisfatível de fórmulas proposicionais.

Outra enorme revolução matemática do século XX que foi escrita na linguagem da lógica foi a prova da independência entre a hipótese do *continuum*⁸ e o axioma da escolha da teoria de conjuntos de Zermelo–Fraenkel ou teoria dos conjuntos axiomática como também é chamada.

De forma sucinta pode-se então concluir que a lógica uma ciência nascida na Grécia antiga se desenvolveu de forma exponencial após o século XIX, e que seu desenvolvimento foi em boa parte guiado por matemáticos, de fato, pode-se dizer que a lógica contemporânea se caracteriza pela tendência da matematização da lógica [11]. Muitos outros estudiosos, além dos que foram aqui mencionados, também apresentaram resultados diretos em lógica ou em área correlatas como a teoria da prova e a teoria da

⁸A hipótese do *continuum* é uma conjectura proposta por Georg Cantor e que fazia parte da lista inicial de 10 problemas estabelecida por David Hilbert. Esta conjectura consiste no seguinte enunciado: **Não existe nenhum conjunto com cardinalidade maior que a do conjunto dos números inteiros e menor que a do conjunto dos números reais.**

recursão, tornando a lógica e suas ramificações e aplicações um dos assuntos dominantes no séculos XX e XXI.

6.3 Argumentos, Proposições e Predicados

Como qualquer outra disciplina para entender de fato o que é a lógica deve-se estudar a mesma [26], antes de qualquer coisa é bom saber que diferente de outras ciências a lógica não apresentar fronteiras bem definidas, na verdade como dito em [70], a lógica pode ser compreendida como a tênue linha que separa as ciências da filosofia e da matemática, no que diz respeito a isto, este manuscrito irá se debruçar primariamente sobre os aspectos matemáticos da lógica.

É sabido que para se estudar uma ciência deve-se saber quais são as entidades fundamentais de interesse dessa ciência, no caso da lógica, estas entidades fundamentais são os argumentos dentro de um discurso. Antes de apresentar a noção formal de argumento é conveniente apresentar a ideia de frase declarativa (ou asserção). As frases declarativas usadas para construção de argumentos são aquelas que como dito em [70], enunciam como as entidades em um certo discurso são ou poderia ter sido, em outras palavras, as frase declarativas falam sobre as propriedades das entidades.

■ **Exemplo 6.1** As frase:

- A lua é feita de queijo.
- O Flamengo é um time carioca.
- O uniforme principal da seleção brasileira é azul.

São ambas frases declarativas. Por outro lado, as frase:

- Que horas são?
- Forneça uma resposta para o exercício.
- Faça exatamente o que eu mandei.
- Cuidado!

Não são frases declarativas.

Uma forma de identificar se uma frase é declarativa é verificado se a mesma admite ser classificada como verdadeira ou falso. Na lógica as frase declarativas podem ser “tipadas” com dois rótulos: **proposições** e **predicados**, formalizados em momentos futuros deste manuscrito.

Definição 6.1 — Argumento. Um argumento é um par formado por dois componentes básicos, a saber:

- (1) Um conjunto de frases declarativas, em que cada frase é chamada de premissa.
- (2) Uma frase declarativa, chamada de conclusão.

Para representar um argumento pode-se como visto em [26, 70] usar uma organização de linhas, por exemplo, para representar um argumento que possua n premissas primeiro serão distribuídas nas n primeiras linhas as tais premissas do argumento depois na linha $n + 1$ é usado o símbolo \therefore para separar as premissas da conclusão, sendo esta última colocada na linha $n + 2$.

■ **Exemplo 6.2** São exemplos de argumentos:

A sopa foi preparada sem cebola
Toda quarta-feira é servida sopa para as crianças.
Hoje é quinta-feira.
 \therefore
Ontem as crianças tomaram sopa.

e

A lua é feita de queijo
Os ratos comem queijo
 \therefore
O imperador da lua é um rato.

A validade de um argumento pode ser analisada em dois aspectos, o semântico e o sintático, este tipo de análise será estudada em capítulos futuros deste manuscrito.

Definição 6.2 — Proposição. Uma proposição é uma frase declarativa sobre as propriedades de indivíduos específicos em um discurso.

■ **Exemplo 6.3** São exemplos de proposições:

- (a) $3 < 5$.
- (b) A lua é feita de queijo.
- (c) Albert Einstein era francês.
- (d) O Brasil é penta campeão de futebol masculino.

Definição 6.3 — Predicados. Predicados são frase declarativas sobre as propriedades de indivíduos não específicos em um discurso.

Pela Definição 6.3 pode-se entender que um predicado fala das propriedades de indivíduos sem explicitamente dar nomes a tais indivíduos.

■ **Exemplo 6.4** São exemplos de predicados:

- (a) Para qualquer $x \in \mathbb{N}$ tem-se que $x < x + 1$.
- (b) Para todo $x \in \mathbb{R}$ sempre existem dois números $y_1, y_2 \in \mathbb{R}$ tal que $y_1 < x < y_2$.
- (c) Existe algum professor cujo nome da mãe é Maria de Fátima.
- (d) Há um estado brasileiro que não tem litoral.
- (e) Todo os moradores de Salgueiro são pernambucanos.

Agora note que nas frase (a) e (b) do Exemplo 6.4 o símbolo x se torna um mecanismo que faz o papel dos números naturais e reais respectivamente, mas sem ser os próprios números em si, o mesmo vale para y_1 e y_2 . Similarmente na frase (c) o termo **professor** representa todo um conjunto de pessoas, mas nunca sendo uma pessoa em particular, já na frase (d) o termo **estado brasileiro** representa novamente todos os indivíduos de um conjunto, mas ele nunca é um indivíduo particular. Os termos em um predicado que tem essa capacidade de representação são chamados de **variáveis do predicado**.

Observação 6.1 Um predicado que tem suas variáveis substituídas por valores específicos se torna uma proposição.

■ **Exemplo 6.5** Considere o predicado: **Existe algum professor cujo nome da mãe é Maria de Fátima**, se for atribuído o valor **Valdigleis** no lugar da variável **professor** será gerado a proposição: **O nome da mãe de Valdigleis é Maria de Fátima**.

6.4 Conectivos, Quantificadores e Negação

As proposições e os predicados podem ser classificados em duas categorias: simples ou composto. Uma proposição (ou predicado) é dita(o) composta(o) sempre que for possível dividi a (o) proposição (predicado) em proposições (predicados) menores. E no caso contrário é dito que a proposição (ou predicado) é simples (ou atômicas).

Definição 6.4 — Conectivos. Conectivos são termos linguísticos que fazem a ligação entre as proposições ou (e) predicados.

Os principais conectivos são: a conjunção, a disjunção, a implicação e a bi-implicação.

Observação 6.2 Dependendo do idioma mais de um termo da linguagem pode representar um determinado conectivo.

A seguir são listados os termos na língua portuguesa que são conectivos, ressaltamos que o símbolo _____ será usado como uma variável para representar a posição de proposições (ou predicados).

Conectivo	Termo em Português
Conjunção	_____ e _____
	_____ mas _____
	_____ também _____
	_____ além disso _____
Disjunção	_____ ou _____
Implicação	Se _____, então _____
	_____ implica _____
	_____ logo, _____
	_____ só se _____
	_____ somente se _____
	_____ segue de _____
	_____ é uma condição suficiente para _____
Bi-implicação	Basta _____ para _____
	_____ é uma condição necessária para _____
	_____ se, e somente se _____
	_____ é condição suficiente e necessária para _____

Tabela 6.1: Termos em português que representamos conectivos.

■ **Exemplo 6.6** Usando as proposições do Exemplo 6.3 e os predicados do Exemplo 6.4 pode-se criar:

- (a) $3 < 5$ e para qualquer $x \in \mathbb{N}$ tem-se que $x < x + 1$.
- (b) Há um estado brasileiro que não tem litoral ou O Brasil é penta campeão de futebol masculino.
- (c) Se para todo $x \in \mathbb{R}$ sempre existem dois números $y_1, y_2 \in \mathbb{R}$ tal que $y_1 < x < y_2$, então Albert Einstein era francês.
- (d) Para qualquer $x \in \mathbb{N}$ tem-se que $x < x + 1$ se, e somente se, para todo $x \in \mathbb{R}$ sempre existem dois números $y_1, y_2 \in \mathbb{R}$ tal que $y_1 < x < y_2$.
- (e) A lua é feita de queijo ou $3 < 5$.

Observação 6.3 O Exemplo 6.6 mostra que os conectivos podem ser usado para combinar proposições com proposições, predicados com predicados, predicados com proposições e vice-versa.

Como já mencionado antes um predicado não especifica diretamente os indivíduos, em vez disso, usa variáveis para não mencionar os indivíduos especificamente. Essas variáveis por sua vez, estão conectadas a termos da linguagem que determinam a quantidade de elementos que podem vim a ser

atribuído a tais variáveis, tais termos são chamados de quantificadores. Os quantificadores por sua vez, podem ser tipados em duas categoria: universais e existenciais.

Quando uma variável é ligada a um quantificador universal significa que o predicado será verdadeiro se para a atribuição de cada um dos elementos do universo discurso a proposição gerada com a atribuição é também verdadeira, no caso contrário o predicado é falso. Por outro lado, quando uma variável é ligada a um quantificador existencial significa que tal predicado será verdadeiro se para pelo menos um dos elementos do discurso ao ser atribuído a variável gera uma proposição verdadeira, e no caso contrário o predicado será falso.

De forma similar aos conectivos os quantificadores também são “representados” por termos da língua portuguesa como mostrado na tabela a seguir.

Quantificador	Termo em Português
Universal	Todo(a)s _____
	Para todo(a) _____
	Para qualquer _____
	Para cada _____
Existencial	Existe _____
	Algum(a) _____
	Para algum _____
	Para um _____

Tabela 6.2: Termos em português que representamos quantificadores.

Anteriormente já foi dito que na lógica as proposições e predicados podem ser interpretados como sendo verdadeiros ou falsos, dito isto, para qualquer proposição ou predicado sempre é possível obter uma proposição ou predicado com um valor de interpretação oposta, isto é, se a proposição (ou predicado) original for verdadeira a proposição (ou predicado) oposta será falsa, ou vice-versa. Esse “construtor” que gerar as proposições (ou predicados) opostas(os) é chamado de negação e a tabela a seguir exhibe como os termos na língua portuguesa podem ser usados para representar a negação.

Termos em português
Não _____
É falso que _____
Não é verdade que _____

Tabela 6.3: Termos em português para designar a negação de uma proposição ou predicado.

■ **Exemplo 6.7** São exemplo de negação:

- (a) Não é verdade que a França faz fronteira com o Brasil.
- (b) Não existe um natural maior que 0.
- (c) Não é verdade que todos os números pares são múltiplos de 6.

6.5 Representação simbólica

Ao estudar lógica é comum adotar o uso de símbolos para representar as proposições, os predicados e os conectivos. Nesse sentido o estudo da lógica está interessada na estrutura dos objetos (proposições e predicados) e não nas frases em si [70]. Nesta seção será apresentado a simbologia básica sem entrar propriamente nos aspectos sintáticos da lógica.

A tabela a seguir apresenta a simbologia dos conectivos, dos quantificadores e também da negação que serão adotados neste manuscrito.

Objeto	Símbolo
Conjunção	\wedge
Disjunção	\vee
Implicação	\Rightarrow
Bi-implicação	\Leftrightarrow
Negação	\neg
Quantificador universal	\forall
Quantificador existencial	\exists

Tabela 6.4: Símbolos usados na Lógica simbólica.

Observação 6.4 No caso do quantificador de existência e unicidade, isto é, aquele que descreve a sentença **existe um único**, o mesmo é representado simbolicamente por $\exists!$, isto é, é usado o símbolo do quantificador existencial seguido de uma exclamação.

Definição 6.5 — Representação das Proposições. As proposições deve ser representadas usando letras maiúsculas do alfabeto latino.

■ **Exemplo 6.8** Representando as proposições “ $2 > 5$ ”, “hoje é quarta feira” e “Alice é a professor de Introdução à Ciência da Computação” respectivamente pela letras P , Q e R tem-se que:

- (a) $P \wedge Q$ representa a proposição: “ $2 > 5$ e hoje é quarta feira”.
- (b) $P \vee P$ representa a proposição: “ $2 > 5$ ou $2 > 5$ ”.
- (c) $R \Rightarrow Q$ representa a proposição: “Se Alice é a professor de Introdução à Ciência da Computação, então hoje é quarta feira”.
- (d) $\neg R \Rightarrow P \vee R$ representa a proposição: “Se não é verdade que Alice é a professor de Introdução à Ciência da Computação, então $2 > 5$ ou Alice é a professor de introdução à Ciência da Computação”.
- (e) $P \Leftrightarrow Q$ representa a proposição: “ $2 > 5$ se, e somente se hoje é quarta feira”.

Agora a representação de um predicado é um pouco mais complexa, primeiro entre parênteses deve-se inserir o símbolo do quantificador e as variáveis ligadas a esse quantificador, se necessário pode-se incluir também o universo a qual essas variáveis pertencem. Em seguida, entre colchetes é inserido a representação de sentença que pode ou não conter as variáveis ligadas ao quantificador, ressaltando que as asserções internas aos colchetes devem ser letras maiúscula do alfabeto latino seguidas imediatamente das variáveis do predicado entre parênteses caso necessário, os exemplos a seguir ilustram estas ideias.

Observação 6.5 Vale ressaltar que os colchetes são símbolos usados para determinar o alcance do quantificador e de suas variáveis.

■ **Exemplo 6.9** O predicado: “Existe um *professor* que a mãe se chama Fátima”, usando p para representar a variável *professor* e MF para representar a asserção da mãe do professor se chamar Fátima, pode-se representar tal predicado como: $(\exists p)[MF(p)]$.

■ **Exemplo 6.10** O predicado: “Existe uma *pessoa* tal que a terra é quadrada”, usando p para representar a variável *pessoa* e T_p para representar a asserção da terra ser plana, pode-se representar tal predicado como: $(\exists p)[T_p]$.

■ **Exemplo 6.11** O predicado: “Existe uma tampa, para fechar toda panela”, pode ser representada da seguinte forma, $(\exists t)[(\forall p)[F(t, p)]]$, aqui t representa a variável tampa e p representa a variável panela, por fim, $F(t, p)$ pode-ser interpretado como a asserção de t fechar p .

■ **Exemplo 6.12** O predicado “Para todo número real, a terra é um planeta”. Pode ser representado por $(\forall n \in \mathbb{R})[P]$ aqui P representa a proposição “a terra é um planeta”.

■ **Exemplo 6.13** O predicado “Todos os homens são mortais”. Pode ser representado por $(\forall h)[M(h)]$ aqui h representa a variável homem e a asserção do homem ser mortal é representado por $M(h)$.

■ **Exemplo 6.14** O predicado: “Para todo x inteiro e todo y inteiro, existe um número inteiro z tal que $x + y = z$ ”. Pode ser representado simbolicamente como, $(\forall x, y \in \mathbb{Z})[(\exists z \in \mathbb{Z})[x + y = z]]$.

Observação 6.6 Aqui vale ressaltar que não existe nada que proíba representar $x + y = z$ por uma palavra da forma $S(x, y, z)$, e assim a representação exibida no exemplo 6.14 poderia ser escrita forma $(\forall x, y \in \mathbb{Z})[(\exists z \in \mathbb{Z})[S(x, y, z)]]$.

O leitor deve ter notado que x, y no Exemplo 6.14 estão juntos no mesmo quantificador, isso ocorre pelo fato de ambos pertencerem ao mesmo universo do discurso (conjunto), estarem ligados a quantificadores do mesmo tipo e seus quantificadores são enunciado de forma aninhada direta⁹, outra forma para apresentar tal predicado seria $(\forall x \in \mathbb{Z})[(\forall y \in \mathbb{Z})[(\exists z \in \mathbb{Z})[x + y = z]]]$.

⁹Dois quantificadores estão aninhados diretamente quando um está logo após o outro, ou seja, o escopo de um é seguido do escopo do outro.

6.6 Lógica e Ciência da Computação

Para finalizar este capítulo introdutório é conveniente falar mesmo que de forma superficial sobre os tipos de lógica e suas relações com a Ciência da Computação. A lógica assim como a física pode ser dividida em duas categorias ou tipos bem definidos, a saber, clássicas e as não clássicas. Como mencionado em [14, 31], as lógicas clássicas são aquelas que apresentam a característica de obedecer os seguintes princípios:

- **Princípio da não contradição:** Qualquer proposição (predicado) não pode ser verdadeira(o) e falsa(o) ao mesmo tempo;
- **Princípio do terceiro excluído:** Toda(o) proposição (predicado) só pode ser falsa(o) ou verdadeira(o), não existe uma terceira possibilidade.

Logo a lógica clássica é bi-valorada [31], ou seja, as interpretações sobre as proposições e predicados só podem ser valoradas por dois valores, a saber: verdadeiro ou falso. E por sua vez, a própria lógica clássica é sub-dividida em duas partes, sendo estas: a lógica proposicional e a lógica de primeira ordem (ou lógica dos predicados).

Com respeito a aplicações a lógica clássica tem um papel fundamental e central para a Ciência da Computação, uma vez que, todos computadores são construídos pela combinação de circuitos digitais e estes por sua vez implementam operações da lógica proposicional [1, 34]. Outra área de destaque da aplicação da lógica dentro da Ciência da Computação é no campo de Inteligência Artificial, onde a mesma é o principal formalismo de representação do conhecimento e portanto é muito útil no desenvolvimento de sistemas especialistas e sistemas multi-agentes [14], algumas outras áreas de aplicação da lógica clássica são:

- Banco de dados: através descrição de consultas e no relacionamento das tabelas em bancos de dados dedutivos.
- Ontologias web: como uma linguagem para descrever ontologias e representar o conhecimento.
- Engenharia de software: usada como formalismo para especificação e verificação formal das propriedades dos sistemas¹⁰.

As lógica não clássicas por sua vez, podem se apresentar de duas forma. (1) não obedecem algum dos princípios apresentados acima ou (2) estendem a lógica clássica através de teoremas e meta-teoremas (discutidos nos próximos capítulos) não válidos para as lógicas clássicas. Como exemplos de lógica não

¹⁰Em especial sistemas críticos como software para controle aéreo são exemplo de sistemas cujas propriedades deve ser especificadas e verificadas com alta precisão matemática dado a importância do mesmo para a manutenção da vida humanas que dependem dele.

clássicas estão: lógica intuicionista [66], lógica paraconsistente [30], lógicas multivaloradas [14, 67], lógicas modais [67] e lógicas temporais [44, 45, 68]. Com respeito a aplicações relacionadas Ciência da Computação tem-se por exemplo:

- A utilização da lógica modal para a verificação da propriedades de sistemas e software [45].
- A lógica temporal usada para especificação e verificação de programas concorrentes [68] e também para especificar circuitos síncronos [44].
- As lógicas multi-valoradas usadas para lidar com a simulação e representação de incertezas presente no raciocínio aproximado [14], principalmente na área de reconhecimento de padrões.

Obviamente com dito em [14], existem muitas outras lógicas não clássicas que têm aplicações ou ainda servem de fundamentação para diversas áreas ou disciplinas da computação. Porém, como esta parte do texto é apenas uma introdução não cabe neste escopo se debruçar tão profundamente assim neste assunto, em capítulos futuros as lógicas não clássicas (em especial a modal) serão estudadas mais a fundo.

6.7 Questionário

■ **Exercício 6.1** Examine cada uma das frases declarativas abaixo e diga se as mesmas são proposições ou predicados e também diga se são simples ou compostas, justifique suas respostas no caso de proposição (ou predicado) composta(o).

- (a). Existe um gato amarelo.
- (b). Alguns patos são marrons.
- (c). Não é verdade que o gato de Júlio é amarelo.
- (d). O Flamengo joga hoje.
- (e). Todos os ratos tem olhos azuis.
- (f). 4 é o menor número composto pelo produto de primos.
- (g). Meu cachorro é branco, alguns outros são vermelhos.
- (h). Alguns gatos são cinza, mas meu gato não é cinza, além disso, Tadeu tem um gato preto ou Sormany tem um rato amarelo.
- (i). Os carros são amarelos se, e somente, se eles não são italianos.

- (j). Se todos os jogadores da seleção jogam na Europa ou Neymar está machucado, então o Brasil não vence a Argentina.
- (k). Basta mais um ponto na carteira de motorista para Sormany perde a aposta ou Juca terá que pagar o almoço.
- (l). Se Lucas é irmão de Pedro, então Natalia vai casar com Gabriel se, e somente se, Francisco voltar da Espanha.
- (m). Se $\frac{10^2}{50} = 2!$, então o $\pi - 2x = 0$ para todo $x \in \mathbb{C}$.
- (n). Se a terra é plana e existe chip nas vacinas, então o Brasil vai conquistar o país de Juvenal.
- (o). A bola é preta.
- (p). Eu tirei foto com meu avô hoje.
- (q). $3 < 5$ segue do fato que o voto no papel é mais rápido que voto eletrônico.
- (r). O sorvete ser de uva segue do fato de Juliane está grávida.
- (s). Bill escreveu o DOS em 1978 é condição necessária de Steven ter lançado o *Apple 2*.
- (t). Se o Cruzeiro é um time da primeira divisão, então todo macaco come macarrão ou não é o caso de Patricia ser professora de matemática.

■ **Exercício 6.2** Determine as frases simples (ou atômicas) que compõem as proposições e predicados que se seguem.

- (a). Juca não irá a festa, mas Pedro irá ou Flaviana irá.
- (b). Fui com a minha família ontem ao parque, e tomei sorvete de chocolate.
- (c). Juca vai com família ou irá sozinho, mas se Anabel aparecer no parque, então Juca e Paula não vão ao cinema.
- (d). Se Paulo chegou, então ele está na sala. Mas não é verdade que Paulo chegou.
- (e). Valdileis toca clarinete somente se, Katia tocar flauta ou Raíssa sabe desenhar com carvão.
- (f). Eu sou aluno da computação somente se, eu passei em Matemática discreta. Mas eu não passei em Introdução à programação ou reprovei todas as disciplinas do primeiro período.
- (g). Para qualquer navio no porto, existe um marinheiro bêbado no bar ou todos os soldados estão dormindo na praia.

- (h). Se Romero tivesse vindo vê o filme, então Katia teria ido para a sorveteria com ele. Mas Romero foi para a praia com Julinha.
- (i). Todos os números primos são números ímpares ou o número π pode ser escrito como produto de dois números.
- (j). Vou a padaria se, e somente se, estiver fazendo frio ou se Juliane quiser tomar sopa.
- (k). Não é verdade que a terra é esférica se, e somente se, não existe pessoas morando em Recife.
- (l). Raíssa e Altair foram para o Chile, mas Luiza também foi.
- (m). Basta que eu tire 8 em Matemática discreta para eu ter um noite de festa.
- (n). A gripe é uma condição suficiente para ser declarado morto.
- (o). Se para todo flor existe um vaso, então os jardins de Jaçanã são cheio de cerejeiras.
- (p). Se ontem choveu a noite toda e hoje é meu aniversário de 14 anos, então Pedro vai a Califórnia se, e somente se, Tom e Frajola forem amigos do Manda-chuva.
- (q). Não é verdade que Raíssa sabe desenhar com carvão, mas sabe desenhar com lápis e pincel.
- (r). O Catatau comeu o mel somente se, o Puffy saiu para passear com o Jack ou Jerry é vizinho do Mickey.
- (s). Para todo homem existe uma mulher que é sua mãe ou Saturno tem várias luas.
- (t). Existe um carro amarelo se, e somente se, todas bicicletas são roxas, mas as motos são azuis.

■ **Exercício 6.3** Converta para notação simbólica todas as questões as sentenças nos Exercícios 6.1 e 6.2.

■ **Exercício 6.4** Converta para notação simbólica as frase declarativas a seguir.

- (a). Todo número natural é maior ou igual que 0.
- (b). Rosas são vermelhas e violetas são azuis.
- (c). Rosas são vermelhas e, violetas são azuis ou açúcar é doce.
- (d). Julia gosta de doces mas Marcos prefere salgados.
- (e). Existe um número real que divide 0.
- (f). Todos os mamíferos sabem nadar.

- (g). Se a galinha nasceu de um ovo, então todas as aves são dinossauros.
- (h). Todos os cisnes são brancos.
- (i). Todos os ratos gostam de leite.
- (j). Se Abel torce para o Botafogo ou o para o Palmeiras, então ele não sabe o que é ganhar um mundial.
- (k). Zico não ganhou uma copa, mas Vampeta ganhou logo, o Vasco vai voltar a primeira divisão.
- (l). Todos os alunos passaram em Matemática Discreta, mas só alguns alunos conseguem tirar 10 em Cálculo logo, não é verdade que todos os alunos reprovaram Administração.
- (m). Não é verdade que, Carla come peixe se, e somente se, Thiago lavar a louça.
- (n). Se Alfredo é paraibano, então Paulo passou em Física ou não é verdade que Manoel é canhoto.

Capítulo 7

Lógica Proposicional

“Ou a matemática é muito grande para a mente humana, ou a mente humana é mais do que uma máquina.”

Kurt Gödel

7.1 Introdução

Como todo bom material de lógica, este capítulo irá tratar inicialmente de forma separada dos aspectos semânticos e sintáticos da lógica proposicional. Inicialmente será apresentado a estrutura da sintaxe e diversos meta-teoremas da linguagem proposicional, após isso, será apresentada a estrutura semântica e vários resultados sobre a mesma, por fim, será desenvolvido um estudo relacional sobre as duas facetas (sintaxe e semântica) da lógica proposicional, ou seja, serão estudadas as noções de corretude e completude.

7.2 A linguagem Proposicional

Este capítulo tem como objetivo apresentar ao leitor o cálculo proposicional, ou seja, o estudo da lógica proposicional, em seus dois aspectos já bem estabelecido por matemáticos e filósofos, isto é, sua sintaxe e sua semântica¹. Assim este capítulo começa com a formalização da linguagem da lógica proposicional, isto é, a linguagem proposicional. A seguir é apresentado formalmente a noção de alfabeto proposicional.

¹O aspecto pragmático da lógica, por ainda se encontrar em um estágio primitivo de seu desenvolvimento, do ponto de vista matemático, não será abordado neste texto, para este assunto ver [94, 99].

Definição 7.1 — Alfabeto Proposicional. O alfabeto proposicional corresponde ao conjunto enumerável $\Sigma = \Sigma_s \cup \Sigma_o \cup \Sigma_p \cup \{\perp\}$ onde:

- $\Sigma_s = \{A, \dots, P, Q, R, P_1, Q_{12}, \dots\}$ é um conjunto enumerável, chamado conjunto de símbolos proposicionais;
- $\Sigma_o = \{\wedge, \vee, \neg, \Rightarrow\}$ é o conjunto dos símbolos operacionais^a;
- $\Sigma_p = \{(\,, \,)\}$ é o conjunto dos símbolos de pontuação e
- \perp é o símbolo do absurdo.

^aTambém é comum encontrar na literatura (ver [70]) a nomenclatura conjunto de conectivos.

Qualquer sequência de símbolos do alfabeto proposicional é chamada de palavra, entretanto, nem toda palavra será considerada como sendo parte da linguagem proposicional. A Definição 7.2 a seguir formaliza o conjunto que corresponde a linguagem proposicional.

Definição 7.2 — Linguagem Proposicional. Dado o alfabeto proposicional Σ a linguagem proposicional, denotada por L_{Prop} , é o menor conjunto indutivamente gerado pelas seguintes regras:

1. Para todo $\alpha \in \Sigma_s \cup \{\perp\}$, tem-se que $\alpha \in L_{Prop}$.
2. Se $\alpha \in L_{Prop}$, então $(\neg\alpha) \in L_{Prop}$.
3. Se $\alpha, \beta \in L_{Prop}$, então $(\alpha \wedge \beta), (\alpha \vee \beta), (\alpha \Rightarrow \beta) \in L_{Prop}$.

■ **Exemplo 7.1** Dado $P, Q, R, S, T \in \Sigma_s \cup \{\perp\}$ tem-se que:

- (a) P
- (b) $(P \wedge Q)$
- (c) $(R \Rightarrow S)$
- (d) $((Q \vee S) \Rightarrow T)$

são todas palavras da linguagem L_{Prop} . Por outro lado, as palavras:

- (e) $P \wedge$
- (f) $\Rightarrow Q$
- (g) $P \vee \wedge Q$

não são palavras da linguagem L_{Prop} .

Observação 7.1 Para simplificar a escrita das palavras de L_{Prop} é comum omitirem-se os parênteses mais exteriores das palavras, por exemplo, é escrito apenas $(Q \vee S) \Rightarrow T$ em vez de $((Q \vee S) \Rightarrow T)$.

É possível enriquecer² a linguagem proposicional adicionando mais símbolos operacionais no alfabeto da mesma, essa introdução é feita utilizando o conceito de abreviação. Uma abreviação na lógica formal consiste na ação de usar um novo símbolo para criar uma nova palavra não presente originalmente na linguagem proposicional, mas que representa uma palavra da linguagem.

Observação 7.2 A palavra $\alpha \Leftrightarrow \beta$ para todo $\alpha, \beta \in L_{Prop}$ poderia ser considerado como uma abreviação para a palavra de L_{Prop} na forma $(\alpha \Rightarrow \beta) \wedge (\beta \Rightarrow \alpha)$. Entretanto, isso não será considerado neste manuscrito, em vez disso, o símbolo \Leftrightarrow será usado para denotar uma relação de equivalência semântica entre palavras de L_{Prop} .

De fato, muitos dos símbolos operacionais que foram tomados como símbolos básicos do alfabeto proposicional (Definição 7.1) poderiam ser removidos, pois como muito bem explicado em [14, 70] a lógica proposicional pode ser definida sobre a linguagem que contém apenas os símbolos operacionais de \Rightarrow e \neg , os demais símbolos podem ser obtidos via abreviação sem qualquer perda no estudo da lógica proposicional, para mais detalhes ver [14].

7.3 Sobre o Sistema de Dedução Natural

A ideia de sistemas dedutivos para a lógica formal remonta aos trabalhos publicados³ no ano de 1934 pelo matemático e filósofo alemão Gerhard Gentzen (1909-1945) e pelo lógico polonês Stanisław Jaśkowski (1906-1965). Existem diversos sistemas dedutivos para a lógica proposicional, cada um possuindo suas próprias características, vantagens e desvantagens, no entanto, todos os sistemas dedutivos compartilham a característica em comum de possuírem um conjunto finito de regras de inferência, esse conjunto de regras de inferência é também chamado de sistema regras ou sistema de dedução [31].

O sistema dedutivo introduzido por Gentzen e Jaśkowski é conhecido por dedução natural, aqui ele será apresentado de forma similar a exposição feita em [70]. O conjunto de regras de inferência da dedução natural é composto pelas regras: de introdução e eliminação de conectivos, regra de reiteração, introdução de hipóteses e a regra do absurdo. Entretanto, antes de apresentar as regras do sistema de dedução natural é conveniente apresentar o conceito de demonstração, para isso deve-se escolher uma notação para as provas da dedução natural.

Existem diversas formas de se escrever (ou representar) uma demonstração no sistema de dedução natural, entre elas destacam-se as árvores de prova de Gentzen [14], o estilo linear [26, 79] e o estilo de

²No sentido de adicionar mais símbolos operacionais.

³Esses trabalhos podem ser encontrados re-editados respectivamente em [102] e [51].

Fitch [70, 39]. Este último é exatamente o sistema usado para os diagramas de bloco do Capítulo 2.

Neste texto será adotado o estilo de Fitch como modelo padrão para a escrita das demonstrações do sistema de dedução natural para a lógica proposicional e posteriormente para a lógica de primeira ordem, assim é conveniente apresentar de forma sucinta o estilo de Fitch [18].

O estilo de Fitch foi introduzido pelo lógico americano Frederic Brenton Fitch (1908 - 1987) e corresponde a diagramas hierárquicos formados por linhas e barras (verticais e horizontais) que representam o raciocínio para a partir de um conjunto de premissas se obter uma determinada conclusão ou objetivo (em inglês *goal*).

O diagrama de Fitch é organizado por linhas numeradas, onde cada linha i pode conter uma única palavra de L_{Prop} , sendo essa palavra uma premissa ou sendo ela obtida pela aplicação de alguma regra de inferência sobre uma ou mais linhas anteriores a linha i . As barras verticais nos diagramas de Fitch são usadas de duas formas:

- (1) Para separar a demonstração em escopos, sendo que um escopo consiste de uma sequencia de várias linhas (ou passos) para demonstrar uma conclusão.
- (2) Como um mecanismo para saber quais palavras de L_{Prop} estão ativas⁴ na prova, como explicado em [70].

As barras horizontais no diagrama de Fitch indicam a divisão entre as afirmações que estão sendo assumidas (podendo ser premissas e (ou) hipóteses) e as palavras que se seguem delas, sejam conclusões intermediárias ou o objetivo final. No caso das hipóteses a barra horizontal também cria um novo “escopo”, isto é, adiciona uma indentação em relação ao escopo anterior, vale salientar que cada escopo é na verdade uma prova para um (sub-)objetivo.

Por fim, é comum na notação dos diagramas de Fitch escrever mais à direita de cada linha a regra de inferência que gerou a palavra na linha, ou o fato da palavra ser uma premissa ou hipótese. Agora pode-se apresentar formalmente o conceito de prova que será adotado neste capítulo.


Definição 7.3 — Prova. Uma prova para $\alpha \in L_{Prop}$ consiste de um diagrama de Fitch como uma quantidade finita de linhas, de forma que a última linha contém a palavra α e cada linha i anterior contém uma palavra $\beta_i \in L_{Prop}$ tal que β_i ou é uma premissa ou é obtida via aplicação de alguma regra de inferência.

Agora pode-se definir precisamente o conceito que relaciona um conjunto de premissas com uma palavra de L_{Prop} , este conceito é conhecido como relação de consequência sintática sobre a linguagem L_{Prop} , o mesmo descreve a noção de derivabilidade.

⁴Uma palavra de L_{Prop} está ativa em uma demonstração, enquanto o escopo da mesma está aberto na demonstração.

Definição 7.4 — Consequência Sintática. Seja L_{Prop} a linguagem proposicional, dado $\alpha \in L_{Prop}$ e $\Gamma \subseteq L_{Prop}$, diz-se que α é consequência sintática de Γ , denotado por $\Gamma \vdash \alpha$, sempre que existir uma prova de α a partir do conjunto de premissas Γ .

Observação 7.3 Note que uma instância de consequência sintática pode ser vista como um elemento de $\mathcal{P}(L_{Prop}) \times L_{Prop}$, isto é, a consequência sintática (\vdash) pode ser vista como uma relação no sentido usual da teoria ingênua dos conjuntos.

 **Nota 7.1** Uma interpretação que se pode ter sobre a relação de consequência sintática $\Gamma \vdash \alpha$ é que a mesma pode ser vista como: a partir das palavras em Γ é sintaticamente possível escrever a palavra α .

A seguir são apresentadas as regras de inferência do sistema de dedução natural.

7.4 Regras de Dedução Natural

Aqui será feito a apresentação das regras de dedução natural iniciando pelas regras que não envolvem diretamente os símbolos operacionais, isto é, as regras que não agem diretamente para eliminar ou introduzir os elementos de Σ_o na demonstração.

Definição 7.5 — Regra das premissas. Seja $\Gamma = \{\alpha_1, \dots, \alpha_n\}$ um conjunto finitos (possivelmente vazio) de premissas, então a regra das premissas fixa que a construção do diagrama de Fitch para uma prova de $\Gamma \vdash \alpha$ dispões nas n primeiras linhas do diagrama as n premissas contidas Γ , onde na linha i se encontra a premissa α_i , além disso, existe uma barra vertical contínua^a a esquerda das premissas e após a linha n há uma barra horizontal separando as premissas do resto da prova, ou seja:

1	α_1	Premissa
\vdots	\vdots	
n	α_n	Premissa
\vdots	\vdots	

^aCada linha vertical contínua é um escopo dentro da demonstração.

Seguindo com as regras mais básicas do sistema de dedução natural pode-se agora apresentar a regra chamada de regra da reiteração, repetição, copia ou clonagem. Tal regra estabelece que se já existe uma dedução de α em um diagrama, então pode-se duplicar a fórmula α no diagrama. Neste manuscrito tal regra será denotada apenas por REI.

Definição 7.6 — Regra da reiteração. Dado uma palavra $\beta \in L_{Prop}$ que já foi deduzida em uma linha i durante a prova, pode-se escrever novamente β em uma linha j com $j > i$, desde que o escopo que contém β ainda esteja ativo^a. Na notação de Fitch tem-se o seguinte diagrama:

$$\begin{array}{c|c}
 \vdots & \vdots \\
 i & \beta \\
 \vdots & \vdots \\
 n & \beta \quad \text{REI, } i \\
 \vdots & \vdots
 \end{array}$$

^aA noção de escopo ativo diz respeito se uma (sub-)prova foi concluída ou ainda está em desenvolvimento, este conceito será melhor trabalhado mais adiante neste manuscrito.

Agora que já foram apresentadas as regras que não agem diretamente sobre os símbolos operacionais pode-se dar sequência no texto apresentando as regras de inferência do sistema de dedução natural que atuam diretamente sobre os símbolos. Como em muitos textos (ver [67]) será inicialmente apresentado a regra de introdução da conjunção.

Definição 7.7 — Regra de introdução da conjunção ($\wedge I$). Se em uma prova já foram deduzidas as palavras $\alpha, \beta \in L_{Prop}$ nas linhas i e j respectivamente, então pode-se deduzir a palavra $\alpha \wedge \beta$ em uma linha k com $i < j < k$, na notação do diagrama de Fitch tem-se:

$$\begin{array}{c|c}
 \vdots & \vdots \\
 i & \alpha \\
 \vdots & \vdots \\
 j & \beta \\
 \vdots & \vdots \\
 k & \alpha \wedge \beta \quad \wedge I, i, j \\
 \vdots & \vdots
 \end{array}$$

Observação 7.4 Para critérios de rigorosidade, a regra de introdução da conjunção impõe que a palavra que está na linha i seja fixada à esquerda do símbolo \wedge e a palavra na linha j seja fixada à direita do símbolo \wedge .

A próxima regra de dedução natural a ser apresentada é a eliminação da conjunção, tal regra possui duas formas de uso, o que contrasta com a regra da introdução da conjunção que possui apenas uma única forma. Primeiramente note que o operador \wedge combina duas palavras $\alpha, \beta \in L_{Prop}$, assim quando tal operador for removido deve-se optar por qual das duas palavras será mantida como uma conclusão

(intermediária ou final) da prova. A seguir é definida formalmente a regra de eliminação de conjunção.

Definição 7.8 — Regra de eliminação da conjunção ($\wedge E$). Se em uma prova for deduzida a palavra $\alpha \wedge \beta$ na linha i , então pode-se deduzir a palavra α ou então a palavra β em uma linha j com $i < j$, na notação do diagrama de Fitch tem-se:

\vdots	\vdots		\vdots	\vdots
i	$\alpha \wedge \beta$		i	$\alpha \wedge \beta$
\vdots	\vdots	ou	\vdots	\vdots
j	α	$\wedge E, i$	j	β
\vdots	\vdots		\vdots	$\wedge E, i$
\vdots	\vdots		\vdots	\vdots

Agora será aberto um parêntese na apresentação das regras de inferência dos símbolos operacionais para que possa ser discutido neste texto a noção de prova hipotética. As provas hipotéticas são muito importantes dentro do sistema de dedução natural, tais provas com dito em [70], podem ser pensadas como sendo um ambiente (ou escopo) de sub-prova em que além das premissas que iniciaram a prova são assumidas outras informações adicionais na forma de hipóteses.

Como argumentado em [26, 70], uma prova hipotética surge quando a regra de introdução hipótese é aplicada, e ao se introduzir essa nova hipótese na prova é gerado um novo escopo dentro da prova que se estava demonstrando, isto é, é criada uma sub-prova que terá seu próprio objetivo.

Definição 7.9 — Regra de introdução de hipótese. Dado uma demonstração com n passos, se for necessário assumir uma hipótese $\beta \in L_{Prop}$ no passo $n + 1$, então é inserida a hipótese β junto com uma barra vertical de escopo, e abaixo de β é inserida a barra horizontal de separação para destacar a hipótese, aqui será usado a palavra **Assuma** para referenciar a regra de introdução de hipótese^a.

\vdots	\vdots		
n	\vdots		
$n + 1$	β	Assuma	
\vdots	\vdots		

^aNa literatura em língua inglesa é comum o uso do termo *Assumption*.

Como dito em [31], uso da regra de inferência de introdução de hipótese está intimamente ligada ao uso da regra de introdução da implicação definida a seguir, por isso a necessidade de apresentá-la antes da regra de introdução da implicação.

Definição 7.10 — Introdução da implicação ($\Rightarrow I$). Se partindo de uma suposição hipotética α na linha m for possível deduzir um certo β na linha n com $m < n$, então no escopo externo da prova hipotética é concluído na linha $n + 1$ que $\alpha \Rightarrow \beta$, na notação dos diagrama de Fitch tem-se:

\vdots		\vdots	
m		α	Assuma
\vdots		\vdots	
n		β	
$n + 1$		$\alpha \Rightarrow \beta$	$\Rightarrow I, m-n$
\vdots		\vdots	

Observação 7.5 Note que a regra de introdução da implicação pode ser vista como um mecanismo que desativa um escopo de prova, isto é, quando a mesma é aplicada um escopo de prova terá sido completado e assim estará desativado (ou fechado).

Vale destacar que ao desativar um escopo de prova todas as palavras contidas unicamente entre as linhas i e j que foram o escopo, não podem mais ser utilizadas na sequência da demonstração, isso ocorre pela razão de tais palavras só existirem no escopo “local” da sub-prova que foi concluída.

Prosseguindo com a apresentação das regras de inferência do sistema de dedução natural agora será definida formalmente a regra de eliminação da implicação, também conhecida como *modus ponens*, que surge da expressão em latin, *modus ponendo ponens*⁵.

Definição 7.11 — Regra da eliminação da implicação ($\Rightarrow E$). Se em uma prova na linha i existe uma palavra α e em uma linha j existe uma palavra $\alpha \Rightarrow \beta$ com $i < j$, então na linha k tal que $j < k$ é possível deduzir a palavra β , em diagrama tem-se:

i		α	
\vdots		\vdots	
j		$\alpha \Rightarrow \beta$	
\vdots		\vdots	
k		β	$\Rightarrow E, i, j$
\vdots		\vdots	

⁵Que em português pode ser traduzido como: o modo de afirmar, afirmando

Observação 7.6 Novamente para manter a rigorosidade sintática a Definição 7.11 especifica que o termo hipotético α deve aparecer na prova antes do termo condicional $\alpha \Rightarrow \beta$, para que se possa aplicar a regra $\Rightarrow E$.

A próxima regra de inferência do sistema de dedução natural que será apresentada neste texto é chamada de regra de introdução do absurdo, a mesma é utilizada para introduzir na demonstração o símbolo do absurdo⁶ (\perp).

Definição 7.12 — Regra de introdução do absurdo ($\perp I$). Se na linha i de uma prova existe uma palavra β e no mesmo escopo de prova na linha j existe uma palavra $\neg\beta$ com $i < j$, então na linha k desta prova é deduzido \perp com $j < k$, em diagrama tem-se:

$$\begin{array}{c|c}
 \vdots & \vdots \\
 i & \beta \\
 \vdots & \vdots \\
 j & \neg\beta \\
 \vdots & \vdots \\
 k & \perp \quad \perp I, i, j \\
 \vdots & \vdots
 \end{array}$$

Observação 7.7 Em alguns texto também é mencionado que o absurdo é na verdade uma abreviação para a palavra $\alpha \wedge \neg\alpha$.

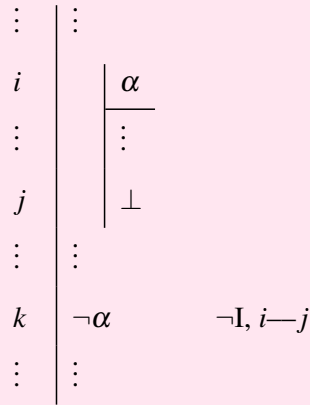
Observação 7.8 O leitor deve ficar atendo ao fato da Definição 7.12 estabelecer que a palavra β deve vim antes da palavra $\neg\beta$ no escopo da prova.

Agora utilizando a regra de introdução do absurdo é gerada uma nova regra do sistema de dedução natural, tal regra é responsável por introduzir a negação. Assim como a regra de introdução da implicação ela também é realizada em uma estrutura de sub-prova.

Observação 7.9 O leitor atento notará a seguir que a introdução da negação sobre uma palavra α não é nada além de uma abreviação para a palavra $\alpha \Rightarrow \perp$.

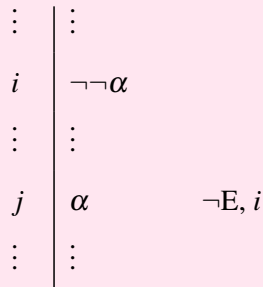
Definição 7.13 — Regra de introdução da negação ($\neg I$). Se existe uma sub-prova iniciada com a hipótese α na linha i que deduz \perp em uma linha j tal que $i < j$, então na linha k com $j < k$ o escopo da sub-prova é fechado e é escrito a palavra $\neg\alpha$. No que diz respeito ao diagrama de Fitch tem-se:

⁶É comum na literatura em língua inglesa principalmente na área de lógica algébrica achar o símbolo do absurdo sendo chamado *bottom*.



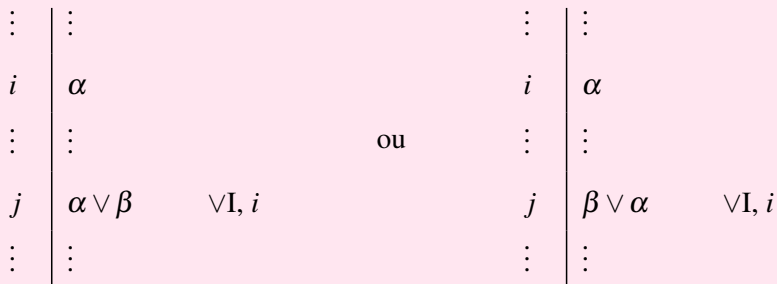
De forma dual a regra a seguir descreve um mecanismo para a eliminação da negação, tal regra pode ser interpretada como a ideia de que negar uma palavra (argumento) duas vezes é o mesmo que afirmar tal palavra (argumento).

Definição 7.14 — Regra de eliminação da negação ($\neg E$). Sempre que existir uma palavra $\neg\neg\alpha$ em uma linha i , então em uma linha j pode-se deduzir α com $i < j$. Em notação de diagrama tem-se:



Por fim, serão agora apresentadas as regras de introdução e eliminação para a disjunção para o sistema de dedução natural.

Definição 7.15 — Regra de introdução da disjunção ($\vee I$). Se em uma prova aparece na linha i uma palavra α , então em uma linha j tal que $i < j$ pode-se deduzir para algum $\beta \in L_{Prop}$ uma das seguintes palavras: $\alpha \vee \beta$ ou $\beta \vee \alpha$. Na notação do diagrama de Fitch tem-se:



A regra de eliminação da disjunção é um pouco mais complicada, pois para ser realizada a mesma invoca duas sub-provas hipotéticas, formalmente tal regra é definida como se segue.

Definição 7.16 — Regra de eliminação da disjunção ($\vee E$). Sempre que existe uma palavra $\alpha \vee \beta$ na i -ésima linha da prova e for possível deduzir γ a partir de sub-provas iniciadas com α e β como hipótese, então na linha n tal que $i < n$ é possível deduzir a palavra γ . Na notação de diagramas tem-se que:

i	$\alpha \vee \beta$	
\vdots	\vdots	
j	α	
\vdots	\vdots	
$j + l_1$	γ	
k	β	
\vdots	\vdots	
$k + l_2$	γ	
\vdots	\vdots	
n	γ	$\vee E, i, (j \text{---} j + l_1, k \text{---} k + l_2)$

Neste ponto do texto tem-se que foram apresentadas todas as regras básicas de inferência para o sistema de dedução natural da linguagem proposicional que foi apresentada na Definição 7.2. Entretanto, pode-se pensar em extensões da linguagem proposicional baseado na ideia de abreviações, assim é natural que os novos símbolos criados também possuam suas regras de introdução e eliminação [21].

7.5 Construção de Demonstrações em Dedução Natural

7.6 Propriedades do Sistema de Dedução Natural

7.7 Sistemas Axiomáticos ao Estilo de Hilbert

7.8 A Semântica da Linguagem L_{Prop}

7.9 Propriedades do Sistema Semântico

7.10 Corretude, Consistência e Completude

Capítulo 8

Lógica de Primeira Ordem

Parte IV

Linguagens Formais e Autômatos

Capítulo 9

Introdução

“Nós só podemos ver um pouco do futuro, mas o suficiente para perceber que há muito a fazer.”

Alan M. Turing

9.1 Sobre as Linguagens Formais

9.2 Noções Fundamentais

Neste primeiro momento para o estudo da teoria das linguagens formais e dos autômatos finitos serão apresentados alguns conceitos fundamentais de extrema importância para o desenvolvimento das próximas seções e capítulos.

Definição 9.1 — Alfabetos e Palavras. Qualquer conjunto finito e não vazio Σ será chamado de alfabeto. Qualquer sequência finita de símbolos na forma $a_1 \cdots a_n$ com $a_i \in \Sigma$ para todo $1 \leq i \leq n$ será chamada de palavra sobre o alfabeto Σ .

■ **Exemplo 9.1** Os conjuntos $\{0, 1, 2, 3\}$, $\{a, b, c\}$, $\{\heartsuit, \spadesuit, \diamondsuit, \clubsuit\}$ e $\{n \in \mathbb{N} \mid n \leq 25\}$ são todos alfabetos, já os conjuntos \mathbb{N} e \mathbb{R} não são alfabetos.

■ **Exemplo 9.2** Dado o alfabeto $\Sigma = \{0, 1, 2, 3\}$ tem-se que as sequências 0123, 102345, 1 e 0000 são todas palavras sobre Σ .

Definição 9.2 — Comprimento das palavras. Seja w uma palavra qualquer sobre um certo alfabeto Σ , o comprimento^a de w , denotado por $|w|$, corresponde ao número de símbolos existentes em w .

^aPor conta desta notação em alguns texto é usado o termo módulo em vez de comprimento.

■ **Exemplo 9.3** Dado o alfabeto $\Sigma = \{a, b, c, d\}$ e as palavras $abcd, aacbd, c$ e $ddaacc$ tem-se que: $|abcd| = 4, |aa| = 2, |c| = 1$ e $|ddaacc| = 6$.

Observação 9.1 Em especial quando $|w| = 1$, é dito que w é uma palavra unitária, isto é, a mesma contém apenas um único símbolo do alfabeto.

Como muito bem explicado em [15, 49, 61], pode-se definir uma série de operações sobre palavras, sendo a primeira delas a noção de concatenação.

Definição 9.3 — Concatenação de palavras. Sejam $w_1 = a_1 \cdots a_m$ e $w_2 = b_1 \cdots b_n$ duas palavras quaisquer, tem-se que a concatenação de w_1 e w_2 , denotado por $w_1 w_2$, corresponde a uma sequência iniciada com os símbolos que forma w_1 imediatamente seguido dos símbolos que forma w_2 , ou seja, $w_1 w_2 = a_1 \cdots a_m b_1 \cdots b_n$.

Observação 9.2 O leitor deve ficar atento ao fato de que a concatenação apenas combina duas palavras em uma nova palavra, sendo que, não a qualquer tipo de exigência sobre os alfabeto sobre os quais as palavras usadas na concatenação estão definidas.

■ **Exemplo 9.4** Dado duas palavras $w_1 = abra$ e $w_2 = cadabra$ tem-se que $w_1 w_2 = abracadabra$ e $w_2 w_1 = cadabraabra$.

Note que o Exemplo 9.4 estabelece que a operação de concatenação entre duas palavras não é comutativa, isto é, a ordem com que as palavras aparecem é responsável pela forma da palavra resultante.

Teorema 9.1 — Associatividade da Concatenação. Para quaisquer w_1, w_2 e w_3 tem-se que $(w_1 w_2) w_3 = w_1 (w_2 w_3)$.

Demonstração. Dado três palavras quaisquer $w_1 = a_1 \cdots a_i, w_2 = b_1 \cdots b_j$ e $w_3 = c_1 \cdots c_k$ tem-se que,

$$\begin{aligned} (w_1 w_2) w_3 &= (a_1 \cdots a_i b_1 \cdots b_j) c_1 \cdots c_k \\ &= a_1 \cdots a_i b_1 \cdots b_j c_1 \cdots c_k \\ &= a_1 \cdots a_i (b_1 \cdots b_j c_1 \cdots c_k) \\ &= w_1 (w_2 w_3) \end{aligned}$$

o que conclui a prova. □

Sobre qualquer alfabeto Σ sempre é definida uma palavra especial chamada **palavra vazia** [49, 61], esta palavra especial não possui nenhum símbolo, e em geral é usado o símbolo λ para denotar a palavra vazia [15, 27]. Como mencionado em [15, 28] sobre a palavra vazia é importante destacar que:

$$w\lambda = \lambda w = w \quad (9.1)$$

$$|\lambda| = 0 \quad (9.2)$$

Isto é, a palavra vazia é neutra para a operação de concatenação, além disso, a mesma apresenta comprimento nulo.

Definição 9.4 — Potência das palavras. Seja w uma palavra sobre um alfabeto Σ a potência de w é definida recursivamente para todo $n \in \mathbb{N}$ como sendo:

$$w^0 = \lambda \quad (9.3)$$

$$w^{n+1} = ww^n \quad (9.4)$$

■ **Exemplo 9.5** Sejam $w_1 = ab, w_2 = bac$ e $w_3 = cbb$ palavras sobre $\Sigma = \{a, b, c\}$ tem-se que:

$$(a) \ w_1^3 = w_1 w_1^2 = w_1 w_1 w_1^1 = w_1 w_1 w_1 w_1^0 = w_1 w_1 w_1 \lambda = ababab.$$

$$(b) \ w_2^2 = w_2 w_2^1 = w_2 w_2 w_2^0 = w_2 w_2 \lambda = w_2 w_2 = bacbac.$$

■ **Exemplo 9.6** Seja $u = 01$ e $v = 231$ tem-se que:

$$uv^3 = uvv^2 = uvvv^1 = uvvv\lambda = uvvv = 01231231231$$

e também

$$u^2v = uu^1v = uu\lambda v = uuv = 0101231$$

Proposição 9.1 Para toda palavra w e todo $m, n \in \mathbb{N}$ tem-se que:

$$(i) \ (w^m)^n = w^{mn}.$$

$$(ii) \ w^m w^n = w^{m+n}.$$

Demonstração. Direto das Definições 9.3 e 9.4, e portanto, ficará como exercício ao leitor. \square

Outro importante conceito existente sobre a ideia de palavra é a noção de palavra reversa (ou inversa) formalmente definida como se segue.

Definição 9.5 — Palavra Reversa. [27] Seja $w = a_1 \cdots a_n$ uma palavra qualquer, a palavra reversa de w denotada por w^r , é tal que $w^r = a_n \cdots a_1$.

■ **Exemplo 9.7** Dado as palavras $u = aba, v = 011101$ e $w = 3021$ tem-se que $u^r = aba, v^r = 101110$ e $w^r = 1203$.

Observação 9.3 Com respeito a noção de palavra reversa tem-se em particular que vale a seguinte igualdade $\lambda^r = \lambda$.

Além das palavras, pode-se também formalizar uma série de operações sobre a própria noção de alfabeto. Em primeiro lugar, uma vez que, alfabetos são conjuntos, obviamente todas operações usuais de união, interseção, complemento, diferença e diferença simétrica (ver Capítulo 1) também são válidas sobre alfabetos. Além dessas operações, também esta definida a operação de potência e os fechos positivo e de Kleene sobre alfabetos.

Definição 9.6 — Potência de um alfabeto. [15] Seja Σ um alfabeto a potência de Σ é definida recursivamente para todo $n \in \mathbb{N}$ como:

$$\Sigma^0 = \{\lambda\} \quad (9.5)$$

$$\Sigma^{n+1} = \{aw \mid a \in \Sigma, w \in \Sigma^n\} \quad (9.6)$$

■ **Exemplo 9.8** Dado $\Sigma = \{a, b\}$ tem-se que $\Sigma^3 = \{aaa, aab, aba, baa, abb, bab, bba, bbb\}$ e $\Sigma^1 = \{a, b\}$

■ **Exemplo 9.9** Seja $\Sigma = \{0, 1, 2\}$ tem-se que $\Sigma^2 = \{00, 01, 02, 10, 11, 12, 20, 21, 22\}$ e $\Sigma^0 = \{\lambda\}$.

O leitor mais atencioso e maduro matematicamente pode notar que para qualquer que seja $n \in \mathbb{N}$ o conjunto potência tem a propriedade de que todo $w \in \Sigma^n$ é tal que $|w| = n$, além disso, é claro que todo Σ^n é sempre finito¹.

Definição 9.7 — Fecho Positivo e de Kleene. Seja Σ um alfabeto o fecho positivo e o fecho de Kleene de Σ , denotados respectivamente por Σ^+ e Σ^* , correspondem aos conjuntos:

$$\Sigma^+ = \bigcup_{i=1}^{\infty} \Sigma^i \quad (9.7)$$

e

$$\Sigma^* = \bigcup_{i=0}^{\infty} \Sigma^i \quad (9.8)$$

Obviamente como dito em [15], o fecho de positivo pode ser reescrito em função do fecho de Kleene usando a operação de diferença de conjunto, isto é, o fecho positivo corresponde a seguinte identidade, $\Sigma^+ = \Sigma^* - \{\lambda\}$. Sobre o fecho de Kleene com destacado em [28] o mesmo corresponde ao monoide livremente² gerado pelo conjunto Σ munida da operação de concatenação.

¹Essa afirmação é facilmente verificável, uma vez que, a mesma nada mais é do que um exemplo de arranjo com repetição.

²Relembre que uma álgebra é livremente gerada quando todo elemento possui fatoração única (a menos de isomorfismo).

Definição 9.8 — Prefixos e Sufixos. Uma palavra $u \in \Sigma^*$ é um prefixo de outra palavra $w \in \Sigma^*$, denotado por $u \preceq_p w$, sempre que $w = uv$, com $v \in \Sigma^*$. Por outro lado, uma palavra u é um sufixo de outra palavra w , denotado por $u \preceq_s w$, sempre que $w = vu$.

■ **Exemplo 9.10** Seja $w = abracadabra$ tem-se que as palavras ab e $abrac$ são prefixos de w , por outro lado $cadabra$ e bra são sufixos de w , e a palavra $abra$ é prefixo e também sufixo. Já a palavra $cada$ não é prefixo e nem sufixo de w .

Definição 9.9 — Conjunto dos Prefixos e Sufixos. Seja $w \in \Sigma^*$ o conjunto de todos os prefixos de w corresponde ao conjunto:

$$PRE(w) = \{w' \in \Sigma^* \mid w' \preceq_p w\} \quad (9.9)$$

e o conjunto de todos os sufixos de w corresponde ao conjunto:

$$SUF(w) = \{w' \in \Sigma^* \mid w' \preceq_s w\} \quad (9.10)$$

■ **Exemplo 9.11** Seja $w = univasf$ tem-se que:

$$PRE(w) = \{\lambda, u, un, uni, univ, univa, univas, univasf\}$$

e

$$SUF(w) = \{\lambda, f, sf, asf, vasf, ivasf, nivasf, univasf\}$$

■ **Exemplo 9.12** A seguir é apresentado alguns exemplos de palavras e seus conjuntos de prefixos e sufixos.

(a) Se $w = ab$, então $PRE(w) = \{\lambda, a, ab\}$ e $SUF(w) = \{\lambda, b, ab\}$.

(b) Se $w = 001$, então $PRE(w) = \{\lambda, 0, 00, 001\}$ e $SUF(w) = \{\lambda, 1, 01, 001\}$.

(c) Se $w = \lambda$, então $PRE(w) = \{\lambda\}$ e $SUF(w) = \{\lambda\}$

(d) Se $w = a$, então $PRE(w) = \{\lambda, a\}$ e $SUF(w) = \{\lambda, a\}$.

Com respeito a cardinalidade dos conjuntos de prefixos e sufixos, os mesmo apresentam as propriedades descritas pelo teorema a seguir.

Teorema 9.2 Para qualquer que seja $w \in \Sigma^*$ as seguintes asserções são verdadeiras.

- (i) $\#PRE(w) = |w| + 1$.
- (ii) $\#PRE(w) = \#SUF(w)$.
- (iii) $\#(PRE(w) \cap SUF(w)) \geq 1$.

Demonstração. Dado uma palavra w tem-se que:

- (i) Sem perda de generalidade assumindo que $w = a_1 \cdots a_n$ logo $w \in \Sigma^n$ (o caso quando $w = \lambda$ é trivial e não será demonstrado aqui) logo $|w| = n$ para algum $n \in \mathbb{N}$, assim existem exatamente n palavras da forma $a_1 \cdots a_i$ com $1 \leq i \leq n$ tal que $a_1 \cdots a_i \preceq_p w$, portanto, para todo $1 \leq i \leq n$ tem-se que $a_1 \cdots a_i \in PRE(w)$, além disso, é claro que $w = \lambda w$, e portanto, $\lambda \in PRE(w)$, consequentemente, $\#PRE(w) = n + 1 = |w| + 1$.
- (ii) É suficiente mostrar que $\#SUF(w) = |w| + 1$, para isso como antes sem perda de generalidade assumo que $w = a_1 \cdots a_n$ e assim tem-se que $w \in \Sigma^n$ logo $|w| = n$ com $n \in \mathbb{N}$, dessa forma existem exatamente n palavras da forma $a_i \cdots a_n$ com $1 \leq i \leq n$ tal que $a_i \cdots a_n \preceq_s w$, portanto, para todo $1 \leq i \leq n$ tem-se que $a_i \cdots a_n \in SUF(w)$, além disso, é claro que $w = w\lambda$, logo $\lambda \in SUF(w)$, consequentemente, $\#SUF(w) = n + 1 = |w| + 1$, e portanto, $\#PRE(w) = \#SUF(w)$. O caso $w = \lambda$ é trivial e não será demonstrado aqui.
- (iii) Trivial, pois $\lambda \in (PRE(w) \cap SUF(w))$, e portanto, tem-se que $\#(PRE(w) \cap SUF(w)) \geq 1$.

□

Corolário 9.1 Toda palavra tem pelo menos um prefixo e um sufixo.

Demonstração. Direto do item (iii) do Teorema 9.2.

□

Seguindo com o texto deste manuscrito pode-se finalmente formalizar o pilar fundamental (a ideia de linguagem) necessário para desenvolver o estudo da computabilidade neste e nos próximos capítulos.

Definição 9.10 — Linguagem. Dado um alfabeto Σ , qualquer subconjunto $L \subseteq \Sigma^*$ será chamado de linguagem.

■ **Exemplo 9.13** Seja $\Sigma = \{0, 1\}$ tem-se que os conjuntos a seguir são todos linguagens sobre Σ .

- (a) Σ^* .
- (b) $\{0^n b^n \mid n \in \mathbb{N}\}$.
- (c) $\{\lambda, 0, 1\}$.

(d) Σ^{22} .(e) \emptyset .

Similarmente ao que ocorre com os alfabetos, as linguagens por serem conjuntos “herdam” as operações básicas da teoria dos conjuntos [63, 64, 2], isto é, estão definidas sobre as linguagens as operações de união, interseção, complemento, diferença e diferença simétrica. E como par aos alfabetos novas operações são definidas.

Definição 9.11 — Concatenação de Linguagens. Sejam L_1 e L_2 duas linguagens, a concatenação de L_1 com L_2 , denotado por L_1L_2 , corresponde a seguinte linguagem:

$$L_1L_2 = \{xy \in (\Sigma_1 \cup \Sigma_2)^* \mid x \in L_1, y \in L_2\} \quad (9.11)$$

■ **Exemplo 9.14** Dado as três linguagens $L_1 = \{\lambda, ab, bba\}$, $L_2 = \{0^{2n}1 \mid n \in \mathbb{N}\}$ e $L_3 = \{a^p \mid p \text{ é primo}\}$ tem-se que:

(a) $L_1L_2 = \{w \mid w = 0^{2n}1 \text{ ou } w = ab0^{2n}1 \text{ ou } w = bba0^{2n}1 \text{ com } n \in \mathbb{N}\}$.

(b) $L_3L_1 = \{w \mid w = a^p \text{ ou } w = a^{p+1}b \text{ ou } a^pbba \text{ onde } p \text{ é um número primo}\}$.

(c) $L_2L_3 = \{0^{2n}1a^p \mid n \in \mathbb{N}, p \text{ é um número primo}\}$.

Definição 9.12 — Linguagem Reversa. Seja L uma linguagem, a linguagem reversa de L , denotada por L^r , corresponde ao conjunto $\{w^r \mid w \in L\}$.

■ **Exemplo 9.15** Considerando as linguagens L_1, L_2 e L_3 do Exemplo 9.14 e a linguagem $\{01, 011\}$ tem-se que:

(a) $L_1^r = \{\lambda, ba, abb\}$.

(b) $L_2^r = \{10^{2n} \mid n \in \mathbb{N}\}$.

(c) $L_3^r = \{a^p \mid n \in \mathbb{N}, p \text{ é um número primo}\}$.

(d) $\{01, 011\}^r = \{10, 110\}$.

Observação 9.4 O leitor mais atento pode perceber que a propriedade involutiva da operação reversa sobre palavras é “herdada” para a reversão sobre linguagens, isto é, para qualquer linguagem L tem-se que $(L^r)^r = L$.

Definição 9.13 — Linguagem Potência. Seja L uma linguagem, a linguagem potência de L , denotada por L^n , é definida recursivamente para todo $n \in \mathbb{N}$ como:

$$L^0 = \{\lambda\} \quad (9.12)$$

$$L^{n+1} = LL^n \quad (9.13)$$

Utilizando o conceito de linguagem potência a seguir é apresentado a formalização para os fechos positivo e de Kleene sobre linguagens.

Definição 9.14 — Fecho positivo e Fecho de Kleene de Linguagens. Seja L uma linguagem, o fecho positivo (L^+) e o fecho de Kleene (L^*) de L são dados pelas equações a seguir.

$$L^+ = \bigcup_{i=1}^{\infty} L^i \quad (9.14)$$

$$L^* = \bigcup_{i=0}^{\infty} L^i \quad (9.15)$$

Por fim, esta seção irá apresentar a noção de linguagem dos prefixos e sufixos.

Definição 9.15 — Linguagem de Prefixos e Sufixos. Seja L uma linguagem, a linguagem dos prefixos e dos sufixos de L , respectivamente $PRE(L)$ e $SUF(L)$, são exatamente os seguintes conjuntos:

$$PRE(L) = \{w' \in \Sigma^* \mid w' \preceq_p w, w \in L\}$$

$$SUF(L) = \{w' \in \Sigma^* \mid w' \preceq_s w, w \in L\}$$

■ **Exemplo 9.16** Considere a linguagem $\{0, 10, 11010\}$ tem-se que:

$$PRE(\{0, 10, 11010\}) = \{\lambda, 0, 1, 10, 11, 110, 1101, 11010\}$$

$$SUF(\{0, 10, 11010\}) = \{\lambda, 0, 10, 010, 1010, 11010\}$$

Nos próximos capítulos deste manuscrito irão ser apresentadas as formalizações da ideia de linguagens formais na visão “mecânica” de Turing [107], entretanto, em vez de apresentar de forma direta os conceitos ligados as máquinas de Turing e as computações por elas realizadas, este manuscrito opta por fazer um estudo seguindo a ideia dos livros texto de linguagens formais [15, 61, 74], assim sendo, esta parte do manuscrito irá apresentar as linguagens formais da mais simples para a mais complexas seguindo a hierarquia de Chomsky [24], ou seja, serão aqui estudadas as linguagens formais na seguintes ordem: regulares, livres do contexto, recursivas e recursivamente enumeráveis. Nas próximas seções será apresentado de forma superficial a ideia de autômatos e gramáticas.

9.3 Sobre Autômatos Finitos

Como dito em [27, 28] uma definição informal do conceito de autômato finito (ou máquina de estado finita) e que tais dispositivos podem ser vistos como sendo máquinas (ou computadores) com dois componentes fundamentais: (1) Um conjunto finito de memórias³, estas sendo subdivididas em células, cada uma das quais capaz de comportar um único símbolo por vez e (2) Uma unidade de controle⁴ que administra o estado atual do autômato e é responsável por executar as instruções (programa) da máquina.

Com respeito as memórias é comum assumir a existência de um **dispositivo de leitura e (ou) escrita**⁵ que é capaz de acessar uma única célula por vez, e assim pode lê e (ou) escrever na célula. A depender do tipo de autômato podem existir vários dispositivos de leitura/escrita ou apenas um [15].

A(s) memória(s) de um autômato finito serve(m) para guarda dados (os símbolos) usados durante o funcionamento do autômato. O funcionamento de um autômato por sua vez, pode ser descrito em tempo discreto [27, 28], assim sendo, em qualquer momento no tempo t , a **unidade de controle** do autômato estará sempre em algum **estado** interno possível e a(s) **unidade(s) de leitura/escrita** tem acesso a alguma(s) **célula(s)** da(s) memória(s).

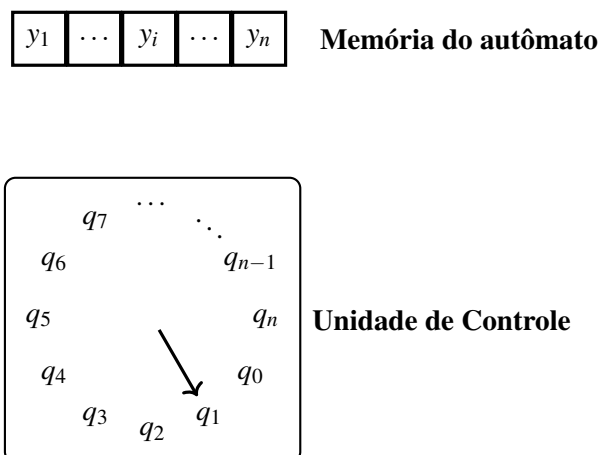


Figura 9.1: Conceito informal de autômato finito com uma única memória retirado de [28].

Formalmente pode-se dizer como apontado em [28], que a teoria dos autômatos finitos, ou simplesmente teoria dos autômatos, teve seu desenvolvimento inicial entre os anos de 1940 e 1960 sendo este início os trabalhos de McCulloch e Pitts [72], Kleene [54], Mealy [73], Moore [77], Rabin e Scott [91, 92]. De forma geral os autômatos finitos são os mais simples modelos abstratos de máquinas de computação [37], sendo eles máquinas de Turing limitadas.

³Na literatura também é usado o termo fita em vez de memória [74].

⁴Também chamada de unidade central de processamento (UCP).

⁵Também é usado a nomenclatura cabeçote [28, 27].

9.4 Sobre Gramática Formais

Agora que foram introduzidos os conceitos fundamentais para a teoria dos autômatos, este prossegue apresentado formalmente o conceito de estrutura geradora ou gramática formal, o leitor mais atento e com maior conhecimento sobre lógica de primeira ordem e teoria da prova [6, 19] pode notar que gramáticas formais são na verdade outro nome para sistemas de reescrita [7].

Definição 9.16 — Gramática formal. Uma gramática formal é uma estrutura com a seguinte forma $G = \langle V, \Sigma, S, P \rangle$ onde V é um conjunto não vazio de símbolos chamados variáveis tal que $V \cap \Sigma = \emptyset$, Σ é um alfabeto, $S \in V$ é uma variável destacada chamada de **variável inicial** e P é um conjunto de regras de reescrita^{ab} da forma $w \triangleright w'$ onde $w \in (V \cup \Sigma)^+$ e $w' \in (V \cup \Sigma)^*$.

^aTamém é comum encontrar na literatura a nomenclatura regras de produção [15, 61].

^bUm leitor com o mínimo de maturidade matemática pode observar que o símbolo P é na verdade o rótulo de uma relação binária, sendo claramente $P \subseteq (V \cup \Sigma)^+ \times (V \cup \Sigma)^*$.

■ **Exemplo 9.17** A estrutura $G = \langle \{A, B\}, \{a\}, A, P \rangle$ em que P é formado pelas regras $A \triangleright aABa, A \triangleright B$ e $B \triangleright \lambda$ é uma gramática formal.

Qualquer gramática então pode ser visto com um sistema para a geração de palavras através de um mecanismo chamado derivação descrito a seguir.

Definição 9.17 — Derivação de palavras. Dado uma gramática $G = \langle V, \Sigma, S, P \rangle$, a palavra XwY deriva a palavra $Xw'Y$ na gramática G , denotado por $XwY \vdash_G Xw'Y$, sempre que existe uma regra forma $w \triangleright w' \in P$.

■ **Exemplo 9.18** Dado a gramática do Exemplo 9.17 tem-se que $aABa \vdash_G aaABaBa$, pois existe em P a regra $A \triangleright aABa$.

Rigorosamente \vdash_G na verdade é uma relação entre $(V \cup \Sigma)^+$ e $(V \cup \Sigma)^*$, e assim \vdash_G^* denota o fecho transitivo e reflexivo de \vdash_G , além disso, sempre que não causar confusão é comum eliminar a escrita do rótulo da gramática, ou seja, são escritos respectivamente \vdash^* e \vdash em vez de \vdash_G^* e \vdash_G .

■ **Exemplo 9.19** Considerando ainda a gramática exibida no Exemplo 9.17 tem-se que $aABa \vdash^* aaaABaBaBa$, uma vez que, $aABa \vdash aaABaBa \vdash aaaABaBaBa$.

■ **Exemplo 9.20** A estrutura $G = \langle \{A, B, S\}, \{0, 1\}, S, P \rangle$ em que P é formado pelas regras $S \triangleright 11A, A \triangleright B0$ e $B \triangleright 000$ é uma gramática formal e assim $11A \vdash^* 110000$, pois tem-se que, $11A \vdash^* 11B0 \vdash^* 110000$.

Como dito em [15], dado uma gramática formal G tem-se que sempre houver uma sequência de derivações $w_1 \vdash w_2 \vdash \dots \vdash w_n$ acontecer, as palavras w_1, w_2, \dots, w_n são chamadas de formas sentenciais, ou simplesmente, sentenças. Assim uma derivação nada mais é do que uma sequência finita de formas sentenciais.

Definição 9.18 — Igualdade de Derivações. Dado uma gramática $G = \langle V, \Sigma, S, P \rangle$ e duas derivações $S \vdash w_1 \vdash^* w_n$ e $S \vdash w'_1 \vdash^* w'_n$ sobre G , será dito que estas derivações são iguais sempre que $w_i = w'_i$ para todo $1 \leq i \leq n$.

Desde que \vdash^* é de fato uma relação tem-se que a igualdade entre derivações nada mais é do que a igualdade entre tuplas ordenadas.

Definição 9.19 — Linguagem de uma gramática. Dado uma gramática $G = \langle V, \Sigma, S, P \rangle$ a linguagem gerada por G , denotada por $\mathcal{L}(G)$, corresponde ao conjunto formado por todas as palavras sobre Σ que são deriváveis a partir do variável inicial da gramática, ou seja, $\mathcal{L}(G) = \{w \in \Sigma^* \mid S \vdash^* w\}$.

■ **Exemplo 9.21** Não é difícil verificar que a gramática do Exemplo 9.17 gera a linguagem $\{w \in \{a\}^* \mid |w| = 2k, k \in \mathbb{N}\}$.

O leitor pode ter notado que como gramáticas formais possuem um conjunto finito de regras, as linguagens por elas geradas nada mais são do que conjuntos indutivamente gerados.

9.5 Questionário

■ **Exercício 9.1** Dado o alfabeto $\Sigma = \{a, b, c\}$ e as palavras $u = aabcbab$, $v = bbccabac$ e $w = ccbabbaaca$ determine o que é solicitado.

- (a). A palavra uv^r .
- (b). A palavra $(w^r)^2u$.
- (c). A palavra $((u^r)^2v^0)^rv$.
- (d). A palavra uu^2v^rw .
- (e). A palavra $((wuv)^r)^2u$.
- (f). O valor da expressão $|w^3| + 2|v^2u| - |u|$.
- (g). O valor da expressão $2|w^r| - |uv|$.
- (h). O valor da expressão $|w^raaw| - |w|$.
- (i). O valor da expressão $|uv^r| - 4$.
- (j). O valor da expressão $\frac{|(w^r)^2u|}{2} - \frac{|u|}{6}$.

■ **Exercício 9.2** Demonstre para quaisquer palavras u e v e para todo $n \in \mathbb{N}$ as asserções a seguir.

- (a). Se u é um prefixo de v , então $|u| \leq |v|$.
- (b). $|u^n| = n|u|$.
- (c). $|(uv)^r| = |vu|$.
- (d). Se $|u| = n$, então $n \leq |uv|$.

■ **Exercício 9.3** Considere a linguagem $L = \{\lambda, abb, a, abba\}$ e determine o que é solicitado a seguir.

- (a). $L^r - \{\lambda, a\}$.
- (b). L^3 .
- (c). $PRE(L)$.
- (d). $SUF(L^2)$.
- (e). w tal que $|w| = \sqcup \{|w'| \mid w' \in L^3\}$.

■ **Exercício 9.4** Prove que para qualquer linguagem L e quaisquer $m, n \in \mathbb{N}$ as seguintes asserções.

- (a). $(L^m)^n = L^{mn}$.
- (b). $L^m L^n = L^{m+n}$.
- (c). $(L^r)^n = (L^n)^r$.
- (d). $\overline{L^r} = \overline{L^r}$.
- (e). $PRE(L) = (SUF(L^r))^r$.

■ **Exercício 9.5** Dado duas linguagens quaisquer L_1 e L_2 demonstre que:

- (a). Se $L_1 \cap L_2 \neq \emptyset$, então $PRE(L_1) \cap PRE(L_2) = \emptyset$.
- (b). Se $L_1 \subseteq L_2$, então $SUF(L_1) \cap SUF(L_2) = \emptyset$.
- (c). Se $L_1 \subseteq L_2$, então $L_1' \subseteq L_2'$.
- (d). Se $L_1 \subseteq L_2$, então para todo L tem-se que $LL_1 \subseteq LL_2$.

■ **Exercício 9.6** Demonstre ou refute o predicado $(\forall L \subseteq \Sigma^*)[(\forall n \in \mathbb{N})[\overline{L^n} = \overline{L^n}]]$.

■ **Exercício 9.7** Dado uma linguagem finita e não vazia L , demonstre por indução L^n é sempre finita para qualquer $n \in \mathbb{N}$.

■ **Exercício 9.8** Esboce formalmente em que condições a igualdade $PRE(L) = (SUF(L))^r$ é verdadeira.

Capítulo 10

Autômatos Finitos e Linguagens Regulares

“Se as pessoas não acreditam que a matemática é simples, é só porque eles não percebem o quão complicada a vida é”

John Von Neumann

10.1 Autômato Finito Determinístico

Como explicado em diversas obras tais como [15, 49, 61, 74], os autômatos finitos podem ser separados em dois tipos bem definidos, a saber Autômato Finito Determinístico (AFD) e Autômato Finito Não-determinístico (AFN). Agora este manuscrito inicia o estudo dos AFD apresentando sua forma algébrica equacional.

Definição 10.1 — Autômato Finito Determinístico. Um AFD é uma estrutura $A = \langle Q, \Sigma, \delta, q_0, F \rangle$ onde: Q é um conjunto finito de estados, Σ é um alfabeto, $\delta : Q \times \Sigma \rightarrow Q$ é uma função total (chamada função de transição), $q_0 \in Q$ é um estado destacado (chamado estado inicial) e $F \subseteq Q$ é o conjunto de estados finais^a.

^aEm algumas referências também é usado o termo conjunto de estados de aceitação [32].

■ **Exemplo 10.1** A estrutura $A = \langle \{q_0, q_1\}, \{a\}, \delta, q_0, \{q_1\} \rangle$ onde a função de transição é definida por: $\delta(q_0, a) = q_1$ e $\delta(q_1, a) = q_0$, é um AFD.

■ **Exemplo 10.2** A estrutura $B = \langle \{q_0, q_1, q_2\}, \{a, b\}, \delta, q_0, \{q_0\} \rangle$ onde a função de transição é definida por:

$$\begin{aligned} \delta(q_0, a) &= q_1 & \delta(q_1, a) &= q_2 & \delta(q_2, a) &= q_1 \\ \delta(q_0, b) &= q_1 & \delta(q_1, b) &= q_2 \end{aligned}$$

não é um AFD, pois $\delta(q_2, b)$ não está definido, e portanto, δ não é uma função total.



Nota 10.1 Para simplificar a escrita neste manuscrito as siglas AFD e AFN serão usado tanto para designar o singular quanto o plural, ficando a distinção a critério dos conectivos antes de cada sigla.

A função de transição (δ) pode ser interpretada semanticamente como sendo o programa que o autômato executa, assim uma aplicação qualquer de δ é uma instrução do programa do autômato, por exemplo, a aplicação $\delta(q, x) = p$ significa que, se o AFD está no estado atual q e o mecanismo de leitura está lendo um x , então mude para o estado p .

Uma representação comum para os AFD é baseada no uso de grafos de transição [28]. Em um grafo de transição os vértices irão ser representados por círculos, que neste caso são usados para representar os estados do autômato, isto é, os círculos representam os elementos de Q . Cada aresta (q_i, q_j) são rotuladas por x representando assim a transição da forma $\delta(q_i, x) = q_j$. Por fim, os estados finais, isto é, cada $q \in F$ será representado por vértices desenhados como um círculo duplo em vez de um círculo simples e o estado inicial é marcado com uma seta.

■ **Exemplo 10.3** A representação por grafo de transição do AFD descrito no Exemplo 10.1 corresponde a figura a seguir.

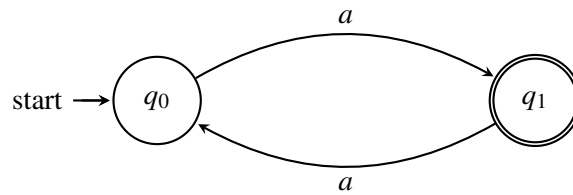


Figura 10.1: Representação visual do AFD no Exemplo 10.1.

■ **Exemplo 10.4** O AFD $S = \langle \{s_0, s_1, s_2\}, \{0, 1\}, \delta, s_0, \emptyset \rangle$ onde a função de transição é definida como sendo: $\delta(s_0, 0) = s_1$, $\delta(s_1, 0) = s_2$, $\delta(s_2, 0) = s_1$, $\delta(s_0, 1) = s_2$, $\delta(s_1, 1) = s_1$ e $\delta(s_2, 1) = s_1$, é um AFD e pode ser representado pela Figura 10.2 a seguir.

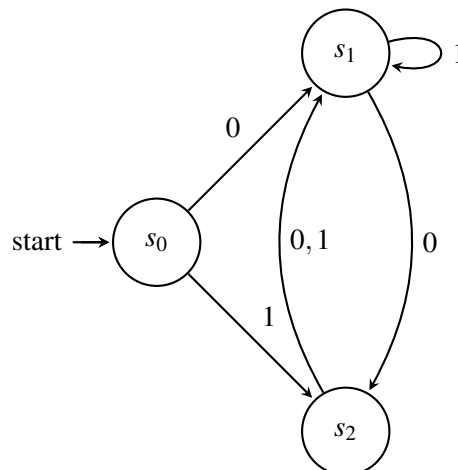


Figura 10.2: Representação visual do AFD S do Exemplo 10.4.

Pode-se agora então estender a função de transição, para que o autômato possa vir a processar palavras, em vez de apenas símbolos individuais.

Definição 10.2 — Função de Transição Estendida. Seja $A = \langle Q, \Sigma, \delta, q_0, F \rangle$ um AFD a função δ é estendida para uma função $\hat{\delta} : Q \times \Sigma^* \rightarrow Q$ usando recursividade como se segue.

$$\hat{\delta}(q, \lambda) = q \quad (10.1)$$

$$\hat{\delta}(q, wa) = \delta(\hat{\delta}(q, w), a) \quad (10.2)$$

onde $q \in Q, a \in \Sigma$ e $w \in \Sigma^*$.

A partir da definição de função de transição estendida é definida a noção de computação para os AFD, tal conceito é formalizado a seguir.

Definição 10.3 — Computação em AFD. Seja $A = \langle Q, \Sigma, \delta, q_0, F \rangle$ um AFD e seja $w \in \Sigma^*$ uma computação de w em A corresponde a aplicação $\hat{\delta}(q_0, w)$.

Note que a definição de computação em AFD pode ser interpretada como sendo a resposta ao seguinte questionamento: “Em que estado o autômato (ou a máquina) estará após iniciar o processamento no estado inicial e ter lido todos os símbolos da palavra de entrada w ?”

■ **Exemplo 10.5** Considere o AFD do Exemplo 10.1 e a palavra de entrada $aaaa$ tem-se que a computação desta palavra corresponde a:

$$\begin{aligned} \hat{\delta}(q_0, aaaa) &= \delta(\hat{\delta}(q_0, aaa), a) \\ &= \delta(\delta(\hat{\delta}(q_0, aa), a), a) \\ &= \delta(\delta(\delta(\hat{\delta}(q_0, a), a), a), a) \\ &= \delta(\delta(\delta(\delta(\hat{\delta}(q_0, \lambda), a), a), a), a), a) \\ &= \delta(\delta(\delta(\delta(q_0, a), a), a), a), a) \\ &= \delta(\delta(\delta(q_1, a), a), a) \\ &= \delta(\delta(q_0, a), a) \\ &= \delta(q_1, a) \\ &= q_0 \end{aligned}$$

■ **Exemplo 10.6** Considere o AFD do Exemplo 10.4 e a palavra de entrada 0101 tem-se que a computação desta palavra corresponde a:

ção desta palavra corresponde a:

$$\begin{aligned}
 \widehat{\delta}(s_0, 0101) &= \delta(\widehat{\delta}(s_0, 010), 1) \\
 &= \delta(\delta(\widehat{\delta}(s_0, 01), 0), 1) \\
 &= \delta(\delta(\delta(\widehat{\delta}(s_0, 0), 1), 0), 1) \\
 &= \delta(\delta(\delta(\delta(\widehat{\delta}(s_0, \lambda), 0), 1), 0), 1) \\
 &= \delta(\delta(\delta(\delta(s_0, 0), 1), 0), 1) \\
 &= \delta(\delta(\delta(s_1, 1), 0), 1) \\
 &= \delta(\delta(s_1, 0), 1) \\
 &= \delta(s_2, 1) \\
 &= s_1
 \end{aligned}$$

De pose da definição de computação pode-se formalizar o conceito de reconhecimento (ou aceitação) de palavras nos AFD.

Definição 10.4 — Reconhecimento de palavras em AFD. [15] Sejam $A = \langle Q, \Sigma, \delta, q_0, F \rangle$ um AFD e seja $w \in \Sigma^*$. A palavra w é dita aceita (reconhece ou computada) por A sempre que $\widehat{\delta}(q_0, w) \in F$ e é rejeitada por A em qualquer outro caso.

É fácil perceber que $\widehat{\delta}(q_0, w) \in F$ com $w = a_1 a_2 \cdots a_n$ se, e somente se, existir uma sequência finita de estados $(q_i)_{i \in I}$ tal que $\delta(q_0, a_1) = q_{i_1}, \delta(q_{i_1}, a_2) = q_{i_2}, \dots, \delta(q_{i_{n-1}}, a_n) = q_{i_n}$ com $q_n \in F, I$ sendo uma sequência de números naturais e $i_1, i_2, i_{n-1}, i_n \in I$. O leitor pode notar que em particular tem-se que $\widehat{\delta}(q_0, \lambda) \in F$ se, e somente se, $q_0 \in F$.

■ **Exemplo 10.7** Considerando os Exemplos 10.5 e 10.6 tem-se que a palavra $aaaa$ não é aceita pelo AFD do Exemplo 10.5, uma vez que, $q_0 \notin F$. Já a palavra 0101 também não é aceita pelo AFD do Exemplo 10.6, uma vez que, $s_1 \notin F$, de fato o leitor atento pode notar que o AFD do Exemplo 10.6 não aceita qualquer palavra de entrada pois $F = \emptyset$.

Observação 10.1 Note que se um AFD não aceitar qualquer palavra, é diferente de dizer que ele não realiza computação, pois como já mencionado anteriormente uma computação é apenas a aplicação da função $\widehat{\delta}$, já não aceitar palavras diz respeito ao fato que para toda palavra w tem-se que $\widehat{\delta}(q_0, w) \notin F$, isto é, a computação das palavras não terminam em um estado final do AFD.

■ **Exemplo 10.8** Considere o AFD representado pelo grafo de transições da Figura 10.3 a seguir. Por indução sobre o tamanho das palavras é fácil mostrar que este AFD reconhece palavras as palavras $1(10)^n 1$ e $0(01)^n 0$ com $n \in \mathbb{N}$.

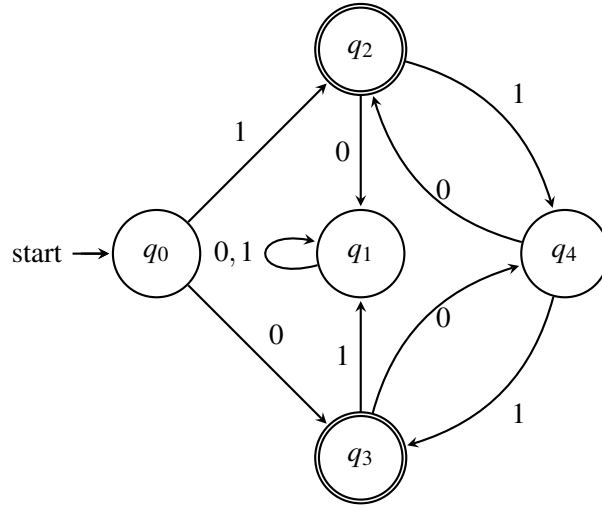


Figura 10.3: Um AFD com dois estados finais.

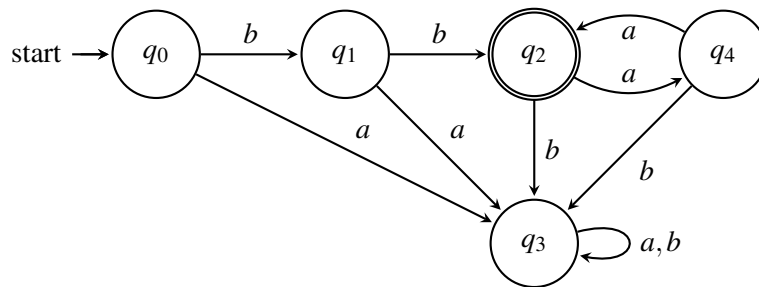
Tendo definido precisamente as noções de AFD e de computação em AFD, agora é possível definir formalmente a ideia de linguagem reconhecida (ou computada) por um AFD.

Definição 10.5 — Linguagem de um AFD. Seja $A = \langle Q, \Sigma, \delta, q_0, F \rangle$ um AFD a linguagem reconhecida (ou computada) por A , denotada por $\mathcal{L}(A)$, corresponde ao conjunto de todas as palavras aceitas por A , formalmente tem-se que:

$$\mathcal{L}(A) = \{w \in \Sigma^* \mid \widehat{\delta}(q_0, w) \in F\} \quad (10.3)$$

Utilizando a definição acima o leitor deve ser capaz de perceber que se um AFD reconhece uma linguagem $L \subseteq \Sigma^*$, então ele para em estados finais apenas para as palavras $w \in L$. Em outra palavra para mostrar que uma linguagem L é a linguagem de um AFD A , deve-se provar que $L = \mathcal{L}(A)$, ou seja, deve-se provar que $w \in L \iff w \in \mathcal{L}(A)$, em geral quando L é infinito tal prova é por indução.

■ **Exemplo 10.9** A seguir você encontrará a prova de que a linguagem $L = \{bba^{2n} \mid n \in \mathbb{N}\}$ é reconhecida pelo AFD A_1 na Figura 10.4 a seguir.

Figura 10.4: AFD A_1 que reconhece a linguagem $\{bba^{2n} \mid n \in \mathbb{N}\}$.

Demonstração. (\Rightarrow) Suponha que $w \in L$ assim $w = bba^{2n}$ e por indução sobre o tamanho das palavras tem-se que,

- **Base da indução:**

Quando $n = 0$ vale que $w = bba^{2 \cdot 0}$ e usando a definição do AFD tem-se que,

$$\widehat{\delta}(q_0, bba^{2 \cdot 0}) = \widehat{\delta}(q_0, bb) = \delta(\widehat{\delta}(q_0, b), b) = \delta(\delta(\widehat{\delta}(q_0, \lambda), b), b) = q_2$$

como $q_2 \in F$ tem-se que $bb \in \mathcal{L}(A_1)$.

- **Hipótese indutiva (HI):**

Suponha que para todo $n \in \mathbb{N}$ tem-se que $\widehat{\delta}(q_0, bba^{2n}) \in F$, ou seja, $\widehat{\delta}(q_0, bba^{2n}) = q_2$.

- **Passo indutivo:**

Dado $w = bba^{2(n+1)}$ tem-se que

$$\begin{aligned} \widehat{\delta}(q_0, bba^{2(n+1)}) &= \widehat{\delta}(q_0, bba^{2n+2}) \\ &= \widehat{\delta}(q_0, bba^{2n}aa) \\ &= \delta(\delta(\widehat{\delta}(q_0, bba^{2n}), a), a) \\ &\stackrel{\text{(HI)}}{=} \delta(\delta(q_2, a), a) \\ &= \delta(q_3, a) \\ &= q_2 \end{aligned}$$

Logo $\widehat{\delta}(q_0, bba^{2n}) \in \mathcal{L}(A_1)$.

(\Leftarrow) Suponha que $w \in \mathcal{L}(A_1)$, assim pela definição do AFD A_1 tem-se que $\widehat{\delta}(q_0, w) = q_2$, entretanto, pela definição de δ (ver Figura 10.4) tem-se que q_2 só é acessado pelas transições $\delta(q_1, b)$ e $\delta(q_4, a)$, ou seja, $w = w_1a$ ou $w = w_2b$ com $w_1, w_2 \in \Sigma^*$. Agora analisando cada possibilidade em separado tem-se que:

- Para realizar o acesso via q_1 é necessário obviamente chegar em q_1 e isso só é possível a partir da transição $\delta(q_0, b)$, logo o acesso a q_2 via q_1 só é permitido para palavras com o prefixo bb , agora como toda palavra é prefixo de si mesmo isso já garante que $bb \in \mathcal{L}(A_1)$.
- Já o acesso via q_4 só é permitido pela transição $\delta(q_2, a)$ e como visto no caso anterior tem-se que o estado q_2 só pode ser acessado por palavras com prefixo bb , note porém, que as transições $\delta(q_2, a) = q_4$ e $\delta(q_4, a) = q_2$ formam um *loop* e assim pode-se concluir que o acesso a q_2 via q_4 obrigatoriamente é realizado por palavras da forma bba^{2n} com $n \geq 1$.

Note que a palavra bb pode ser escrita como sendo bba^0 , portanto, pelas duas análises anteriores pode-se concluir que se $\widehat{\delta}(q_0, w) = q_2$, então $w = bba^{2n}$ com $n \in \mathbb{N}$, logo $w \in L$, completando assim a prova. \square

■ **Exemplo 10.10** O AFD A do Exemplo 10.1 reconhece a linguagem $L = \{a^{2n+1} \mid n \in \mathbb{N}\}$.

Demonstração. (\Rightarrow) Suponha que $w \in L$ assim $w = a^{2n+1}$, agora por indução sobre o tamanho das palavras tem-se que,

- **Base da indução:**

Quando $n = 0$ vale a igualdade $w = a^{2 \cdot 0 + 1}$, agora usando a definição do AFD A tem-se que,

$$\widehat{\delta}(q_0, a^{2 \cdot 0 + 1}) = \widehat{\delta}(q_0, a^1) = \delta(\widehat{\delta}(q_0, \lambda), a) = \delta(q_0, a) = q_1$$

e como $q_1 \in F$ tem-se que $a^{2 \cdot 0 + 1} \in \mathcal{L}(A)$, ou seja, $w \in \mathcal{L}(A)$.

- **Hipótese indutiva (HI):**

Suponha que para todo $n \in \mathbb{N}$ tem-se que $\widehat{\delta}(q_0, a^{2n+1}) \in F$, ou seja, $\widehat{\delta}(q_0, a^{2n+1}) = q_1$.

- **Passo indutivo:**

Dado $w = a^{2(n+1)+1}$ tem-se que,

$$\begin{aligned} \widehat{\delta}(q_0, a^{2(n+1)+1}) &= \widehat{\delta}(q_0, a^{2n+1+2}) \\ &= \widehat{\delta}(q_0, a^{2n+1}aa) \\ &= \delta(\delta(\widehat{\delta}(q_0, a^{2n+1}), a), a) \\ &\stackrel{(HI)}{=} \delta(\delta(q_1, a), a) \\ &= \delta(q_0, a) \\ &= q_1 \end{aligned}$$

(\Leftarrow) A volta fica como exercício argumentativo ao leitor. □

Pode-se agora formalizar a primeira das classes de linguagens sendo esta a classe das linguagens regulares, tal classe foi primeiramente definida por Kleene em seu trabalho [54], entretanto, em tal ocasião tais linguagens foram chamadas de eventos regulares, como será visto é momentos futuros nesse manuscrito a classe das linguagens regulares é aquela que possui o menor nível complexidade computacional.

Definição 10.6 — Linguagens Regulares. Uma linguagem L qualquer é dita ser regular se, e somente se, existe um AFD A tal que $L = \mathcal{L}(A)$. A classe de todas as linguagens regulares é denotada por \mathcal{L}_{Reg} .

10.2 Autômato Finito Não-determinístico

Como explicado por Peter Linz em [61], um autômato finito não-determinístico, ou simplesmente AFN, é um autômato que se diferencia dos AFD apenas no quesito da função de transição. A diferença consiste no fato de que, enquanto a imagem da função de transição em um AFD é sempre um estado, nos AFN a imagem da função de transição é um subconjunto de estados, em um sentido moderno da teoria dos autômatos, um AFN seria uma máquina que algumas transições geraria uma superposição de estados [28]. Formalmente um AFN é como se segue.

Definição 10.7 — Autômato Finito Não-determinístico. Um AFN é uma estrutura $A = \langle Q, \Sigma, \delta_N, q_0, F \rangle$ onde: Q, Σ, q_0 e F são da mesma forma que na Definição 10.1, já $\delta_N : Q \times \Sigma \rightarrow \wp(Q)$ é uma função total (chamada função de transição não determinística).

■ **Exemplo 10.11** A estrutura $A = \langle \{q_0, q_1, q_2\}, \{a, b\}, \delta_N, q_0, \{q_0, q_1\} \rangle$ onde a função δ é descrita pela Tabela 10.1 a seguir é um AFN.

$Q \backslash \Sigma$	a	b
q_0	$\{q_1\}$	$\{q_0\}$
q_1	$\{q_2\}$	$\{q_0, q_2\}$
q_2	$\{q_2\}$	$\{q_1\}$

Tabela 10.1: Tabela de transição para a função δ_N do AFN no Exemplo 10.11.

A representação visual de um AFN usando grafos de transição é construída exatamente da mesma forma que a representação de um AFD, a diferença é o fato de poder existir múltiplas arestas rotuladas por um símbolo $a \in \Sigma$ saindo de um vértice q_i e chegando nos vértices $q_j \in X$ onde $X \subseteq Q$ e existe a transição $\delta_N(q_i, a) = X$.

■ **Exemplo 10.12** O grafo de transição representado na Figura 10.5 a seguir é uma representação para o AFN do Exemplo 10.11.

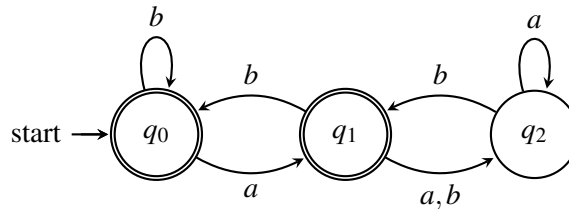


Figura 10.5: Grafo de transição do AFN do Exemplo 10.11.

Observação 10.2 Para as transições da forma $\delta_N(q_0, a) = \emptyset$ tem-se que as mesmas não são representadas no grafo de transição de um AFN.



Nota 10.2 Em muitos texto como por exemplo [15, 61, 100], as funções de transição não determinísticas são representadas simplesmente por δ , neste texto está sendo usada a notação δ_N simplesmente para enfatizar a natureza não determinística do autômato na esperança de torna o entendimento mais simples para o leitor.

Como para o caso determinístico a função de transição δ_N também pode ser estendida para uma função $\widehat{\delta}_N$ usando recursividade como se segue.

Definição 10.8 — Transição não-determinística estendida. Seja $A = \langle Q, \Sigma, \delta_N, q_0, F \rangle$ um AFN, a função de transição estendida é uma função $\delta_N : Q \times \Sigma^* \rightarrow \wp(Q)$ definida pela seguinte recursão.

$$\widehat{\delta}_N(q, \lambda) = \{q\} \quad (10.4)$$

$$\widehat{\delta}_N(q, wa) = \bigcup_{q' \in \widehat{\delta}_N(q, w)} \delta_N(q', a) \quad (10.5)$$

Como para os AFD a noção de computação em qualquer AFN consiste simplesmente da aplicação da função $\widehat{\delta}_N$ sobre alguma palavra $w \in \Sigma^*$ e um estado q .

■ **Exemplo 10.13** Considerando o AFN ilustrado na Figura 10.5 e a palavra “abb” tem-se que,

$$\widehat{\delta}_N(q_0, abb) = \bigcup_{q' \in \widehat{\delta}_N(q_0, ab)} \delta_N(q', b) \quad (10.6)$$

e também tem-se que,

$$\widehat{\delta}_N(q_0, ab) = \bigcup_{q'' \in \widehat{\delta}_N(q_0, a)} \delta_N(q'', b) \quad (10.7)$$

e

$$\begin{aligned} \widehat{\delta}_N(q_0, a) &= \bigcup_{q''' \in \widehat{\delta}_N(q_0, \lambda)} \delta_N(q''', a) \\ &= \bigcup_{q''' \in \{q_0\}} \delta_N(q''', a) \\ &= \delta_N(q_0, a) \\ &= \{q_1\} \end{aligned} \quad (10.8)$$

substituindo a Equação (10.8) na Equação (10.7) tem-se que,

$$\widehat{\delta}_N(q_0, ab) = \{q_0, q_2\} \quad (10.9)$$

e finalmente substituindo a Equação (10.9) na Equação (10.6) tem-se que,

$$\widehat{\delta}_N(q_0, aba) = \{q_0, q_1\} \quad (10.10)$$

ou seja, a computação da palavra “abb” pelo AFN da Figura 10.5 termina no conjunto de estados $\{q_0, q_1\}$.

Pelo exemplo anterior o leitor mais atento pode ter notado que diferente do caso determinístico, a computação em um AFN não é linear, no sentido de que não existe um único caminho de computação¹, em vez disso, a computação em um AFN pode ser vista como uma árvore em que a união dos estados em cada nível da árvore representa a superposição de estados assumida pela unidade de controle do autômato a cada símbolo consumido (computado ou lido) da palavra w , o exemplo a seguir ilustra bem essa ideia de árvore de computação.

■ **Exemplo 10.14** Considerando o AFN ilustrado na Figura 10.5 e a palavra “abab” tem-se que o processo de computação para tal palavra poder ser representado pela árvore da Figura 10.6 a seguir.

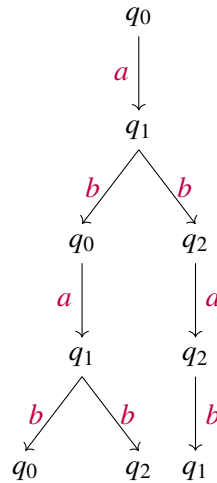


Figura 10.6: Árvore de computação da palavra “abab” no AFN da Figura 10.5.

Pode-se agora apresentar a noção de aceitação (reconhecimento ou computação) de palavras nos AFD.

Definição 10.9 — Reconhecimento de palavras em AFN. Sejam $A = \langle Q, \Sigma, \delta_N, q_0, F \rangle$ um AFN e seja $w \in \Sigma^*$. A palavra w é dita aceita (reconhece ou computada) por A sempre que $\widehat{\delta}(q_0, w) \cap F \neq \emptyset$ e é rejeitada por A em qualquer outro caso.

Note que a Definição 10.9 pode ser informalmente interpretada da seguinte forma, uma palavra é aceita por um AFN A se existe pelo menos um caminho de computação para w que termine em um estado final, isto é, pelo menos uma das folhas na árvore de computação deve ser um estado $q \in F$, neste

¹Como explicado em [28] um caminho de computação é uma sequência de estados assumidos pela unidade central do autômato durante o processamento de uma palavra de entrada.

caso w é aceita por A .

■ **Exemplo 10.15** Considerando o AFN representado pela Figura 10.7 a seguir,

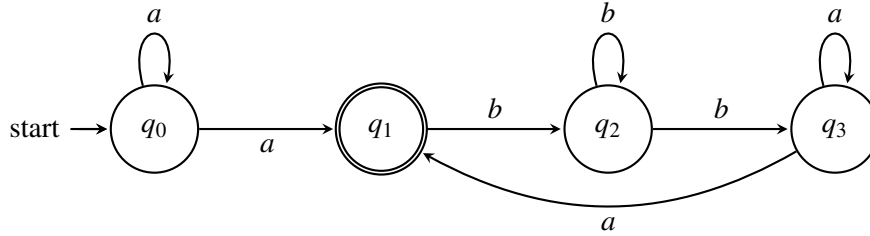


Figura 10.7: Grafo de transição de um AFN do Exemplo 10.15.

para as palavras “aabbba” e “aabb” tem-se que:

$$\widehat{\delta}_N(q_0, aabbba) = \{q_1, q_3\}$$

e

$$\widehat{\delta}_N(q_0, aabb) = \{q_2, q_3\}$$

logo a palavra “aabbba” é aceita por tal AFN. Por outro lado, a palavra “aabb” não é aceita pelo AFN.

Usando a definição apresentada anteriormente de palavra aceita pode-se finalmente introduzir formalmente a noção de linguagem aceita (computada ou reconhecida) pelos AFN.

Definição 10.10 — Linguagem de um AFN. Seja $A = \langle Q, \Sigma, \delta_N, q_0, F \rangle$ um AFN a linguagem reconhecida (ou computada) por A , denotada por $\mathcal{L}(A)$, corresponde ao conjunto de todas as palavras aceitas por A , formalmente tem-se que:

$$\mathcal{L}(A) = \{w \in \Sigma^* \mid \widehat{\delta}_N(q_0, w) \cap F \neq \emptyset\} \quad (10.11)$$

De forma similar ao que ocorre com os AFD, para mostrar que uma linguagem L é aceita por algum AFN A deve-se provar a igualdade $L = \mathcal{L}(A)$, ou seja, deve-se provar que $w \in L \iff w \in \mathcal{L}(A)$.

■ **Exemplo 10.16** A linguagem $L = \{a^i(ba)^j \mid i \geq 1, j \geq 0\}$ é aceita pelo AFN A representado pelo grafo de transição da Figura 10.8 a seguir.

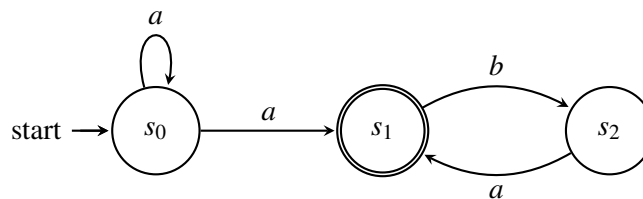


Figura 10.8: Grafo de transição de um AFN.

Demonstração. (\Rightarrow) Suponha que $w \in L$, portanto, $w = a^m(ba)^n$, e agora por indução dupla sobre o par (m, n) tem-se que:

- **Base da indução:**

Quando com $m = 1$ e $n = 0$ vale a igualdade $w = a^1(ba)^0 = a$, agora usando a definição de δ_N do AFN A como representado na Figura 10.8 tem-se que,

$$\widehat{\delta}_N(s_0, a) = \bigcup_{s' \in \widehat{\delta}(s_0, \lambda)} \delta_N(s', a) = \delta_N(s_0, a) = \{s_0, s_1\}$$

uma vez que, $s_1 \in F$ tem-se que $\widehat{\delta}_N(s_0, a) \cap F \neq \emptyset$, e portanto, $w \in \mathcal{L}(A)$. Agora suponha que para $w = a^1(ba)^n$ com $n \geq 0$ tem-se que $\widehat{\delta}_N(s_0, a^1(ba)^n) \cap F \neq \emptyset$. Assim dado $a^1(ba)^{n+1}$ por definição tem-se que:

$$\begin{aligned} \widehat{\delta}_N(s_0, a^1(ba)^{n+1}) &= \widehat{\delta}_N(s_0, a^1(ba)^n ba) \\ &= \bigcup_{s' \in \widehat{\delta}_N(s_0, a^1(ba)^n b)} \delta_N(s', a) \end{aligned} \quad (10.12)$$

agora fazendo,

$$K = \bigcup_{s'' \in \widehat{\delta}_N(s_0, a^1(ba)^n)} \delta_N(s'', b) \quad (10.13)$$

e reescrevendo a Equação (10.12) usando a Equação (10.13) tem-se que,

$$\widehat{\delta}_N(s_0, a^1(ba)^{n+1}) = \bigcup_{s' \in K} \delta_N(s', a) \quad (10.14)$$

entretanto, por hipótese tem-se que $\widehat{\delta}_N(s_0, a^1(ba)^n) \cap F \neq \emptyset$, consequentemente, tem-se que $s_1 \in \widehat{\delta}_N(s_0, a^1(ba)^n)$ dessa forma pela Equação (10.13) é claro que $\delta_N(s_1, b) \subseteq K$. Mas $\delta_N(s_1, b) = \{s_2\}$ logo pela Equação (10.14) tem-se que $\delta_N(s_2, a) \subseteq \widehat{\delta}_N(s_0, a^1(ba)^{n+1})$, desde que $\delta_N(s_2, a) = \{s_1\}$, tem-se $s_1 \in \widehat{\delta}_N(s_0, a^1(ba)^{n+1})$, portanto, $\widehat{\delta}_N(s_0, a^1(ba)^{n+1}) \cap F \neq \emptyset$, consequentemente $a^1(ba)^{n+1} \in \mathcal{L}(A)$.

- **Hipótese indutiva (HI):**

Assuma que para todo $n \geq 0$ tem-se que $\widehat{\delta}_N(s_0, a^n(ba)^n) \cap F \neq \emptyset$.

- **Passo indutivo:**

Primeiro seja $w \in L$ de forma que $w = a^{m+1}(ba)^0$ logo pela hipótese indutiva segue que,

$$\widehat{\delta}_N(s_0, a^{m+1}(ba)^0) \cap F \neq \emptyset$$

consequentemente, $a^{m+1}(ba)^0 \in \mathcal{L}(A)$. Por outro lado, sendo $w \in L$ tal que $w = a^{m+1}(ba)^n$, usando a definição de $\widehat{\delta}_N$ tem-se para $a^{m+1}(ba)^{n+1}$ que,

$$\begin{aligned} \widehat{\delta}_N(s_0, a^{m+1}(ba)^{n+1}) &= \widehat{\delta}_N(s_0, a^{m+1}(ba)^n ba) \\ &= \bigcup_{s' \in \widehat{\delta}_N(s_0, a^{m+1}(ba)^n b)} \delta_N(s', a) \end{aligned} \quad (10.15)$$

agora desenvolvendo o termo $\widehat{\delta}_N(s_0, a^{m+1}(ba)^n b)$ tem-se

$$\widehat{\delta}_N(s_0, a^{m+1}(ba)^n b) = \bigcup_{s'' \in \widehat{\delta}_N(s_0, a^{m+1}(ba)^n)} \delta_N(s'', b)$$

pela hipótese indutiva tem-se que $\widehat{\delta}_N(s_0, a^{m+1}(ba)^n) \cap F = \emptyset$, consequentemente, $s_1 \in \widehat{\delta}_N(s_0, a^{m+1}(ba)^n)$, logo $\delta_N(s_1, b) \subseteq \widehat{\delta}_N(s_0, a^{m+1}(ba)^n)$, uma vez que, $\delta_N(s_1, b) = \{s_2\}$, tem-se que $\{s_2\} \subseteq \widehat{\delta}_N(s_0, a^{m+1}(ba)^n)$, assim pela Equação (10.15) segue que $\delta_N(s_2, a) \subseteq \widehat{\delta}_N(s_0, a^{m+1}(ba)^{n+1})$, mas por definição $\delta_N(s_2, a) = \{s_1\}$, portanto, tem-se que $\{s_1\} \subseteq \widehat{\delta}_N(s_0, a^{m+1}(ba)^{n+1})$, logo $\widehat{\delta}_N(s_0, a^{m+1}(ba)^{n+1}) \cap F \neq \emptyset$ e assim $a^{m+1}(ba)^{n+1} \in \mathcal{L}(A)$.

(\Leftarrow) Suponha que $w \in \mathcal{L}(A)$ assim $s_1 \in \widehat{\delta}_N(s_0, w)$, note porém que s_1 só é acessível a partir de duas transições:

- (1) $\delta_N(s_0, a)$ e
- (2) $\delta_N(s_2, a)$.

Note que devido ao *loop* fornecido pelo fato de que $s_0 \in \delta_N(s_0, a)$ a transição (1) pode ser executada m vezes com $m \geq 1$, em que para cada execução um novo ramo com o estado s_1 é gerado na árvore de computação de A , entretanto, executar m vezes a transição $\delta_N(s_0, a)$ implica em executar a computação $\widehat{\delta}_N(s_0, a^m)$, pelo fato² de que $s_1 \in \widehat{\delta}_N(s_0, a^m)$ tem-se que $a^m \in \mathcal{L}(A)$, e uma vez que $a^m = a^m(ba)^0$ tem-se que a primeira forma de $\widehat{\delta}_N(s_0, w) \cap F \neq \emptyset$ é que $w = a^m(ba)^0$ e assim $w \in L$. Por outro lado, para acessar s_1 via a transição (2) é necessário antes chegar a um ramo de computação em que o estado s_2 seja uma folha, mas pela definição de A isso só é possível se a transição $\delta_N(s_1, b)$ for usada, note entretanto, que as transições $\delta_N(s_1, b) = \{s_2\}$ e $\widehat{\delta}_N(s_2, a) = \{s_1\}$ também geram um *loop* que pode ser executado n vezes com $n \geq 0$, mas executar esse *loop* n vezes corresponde a executar $\widehat{\delta}_N(s_1, (ba)^n)$, e como dito

²Fica para o leitor a tarefa de provar que para todo $m \geq 1$ tem-se que $s_1 \in \widehat{\delta}_N(s_0, a^m)$.

anteriormente, s_1 só é acessível pela definição de A usando a computação $\widehat{\delta}_N(s_0, a^m)$, portanto, para que $s_1 \in \widehat{\delta}_N(s_0, w) \cap F$, obrigatoriamente, $w = a^m(ba)^n$ com $m \geq 1, n \geq 0$, e portanto, $w \in L$. \square

■ **Exemplo 10.17** O AFN S representado no grafo de transição exposto na Figura 10.9 a seguir reconhece a linguagem $L = \{uv \mid u \in \{0, 1\}^*, v \in \{0, 1\}\}$.

Demonstração. (\Rightarrow) A ida fica a cargo do leitor. (\Leftarrow) Suponha que $w \in \mathcal{L}(A)$ assim por definição $\widehat{\delta}_N(s_0, w) \cap \{s_1, s_2\} \neq \emptyset$, agora pela definição de δ_N é claro que toda árvore de computação de A apresenta a propriedade de sempre conter um dos estados s_1 ou s_2 , mas nunca os dois simultaneamente³, além disso, o fato de $s_0 \in \widehat{\delta}_N(s_0, a)$ para todo $a \in \{0, 1\}$, garante que qualquer palavra não a vazia u sobre o alfabeto $\{0, 1\}$ pode ser gerada, por fim, no último passo de computação é claro que s_1 ou s_2 será uma folha da árvore, entretanto, s_1 só será tal folha no caso da palavra terminar em 0 caso contrário a folha será s_2 , e portanto, todo $w \in \mathcal{L}(A)$ tem a forma uv com $u \in \{0, 1\}^*$ e $v \in \{0, 1\}$, consequentemente $w \in L$. \square

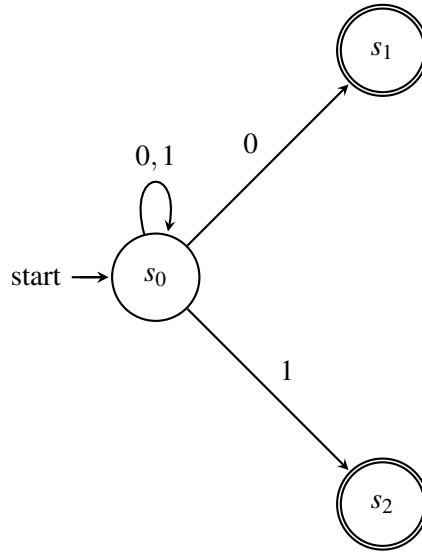


Figura 10.9: Grafo de transição de um AFN S do Exemplo 10.17.

De forma ingênua o leitor pode vir a imaginar que a possibilidade da unidade de controle de um AFN poder assumir mais de um estado interno simultaneamente, faz com que os AFN sejam mais poderosos que os AFD, entretanto, como será exibido pelos resultados a seguir, isso não ocorre, de fato, como dito [15, 61] apesar de tornar mais fácil a tarefa de construir um autômato quem reconheça uma linguagem L , o não-determinismo não aumenta em nada o poder de computação dos autômatos finitos.

Teorema 10.1 — Transformação AFN - AFD. Se $L = \mathcal{L}(A)$ para algum AFN A , então existe um AFD A' tal que $L = \mathcal{L}(A')$.

³A prova desta propriedade fica como exercício ao leitor.

Demonstração. Suponha que $L = \mathcal{L}(A)$ para algum AFN $A = \langle Q, \Sigma, \delta_N, q_0, F \rangle$, agora é construído um autômato $A' = \langle \wp(Q), \Sigma, \delta, \{q_0\}, F' \rangle$ onde para todo $X \in \wp(Q)$ e $a \in \Sigma$ tem-se

$$\delta(X, a) = \bigcup_{q \in X} \delta_N(q, a) \quad (10.16)$$

claramente este autômato é realmente determinístico, e para todo $X \in \wp(Q)$ tem-se que $X \in F'$ se, e somente se, $X \cap F \neq \emptyset$. Agora será mostrado por indução sobre o tamanho de $w \in \Sigma^*$ que:

$$\widehat{\delta}(\{q_0\}, w) = \widehat{\delta}_N(q_0, w)$$

- **Base da indução:**

Quando $|w| = 0$ isto é $w = \lambda$ tem-se trivialmente pela definição das funções de transição estendidas que $\widehat{\delta}(\{q_0\}, \lambda) = \widehat{\delta}_N(q_0, \lambda)$.

- **Hipótese indutiva (HI):**

Suponha que para todo $w \in \Sigma^*$ com $|w| \geq 0$ tem-se que $\widehat{\delta}(\{q_0\}, w) = \widehat{\delta}_N(q_0, w)$.

- **Passo indutivo:**

Dado $w = ua$ com $u \in \Sigma^*$, $|u| \geq 0$ e $a \in \Sigma$ tem-se que,

$$\begin{aligned} \widehat{\delta}(\{q_0\}, w) &= \widehat{\delta}(\{q_0\}, ua) \\ &= \delta(\widehat{\delta}(\{q_0\}, u), a) \\ &\stackrel{\text{(HI)}}{=} \delta(\widehat{\delta}_N(q_0, u), a) \\ &\stackrel{\text{Eq. (10.16)}}{=} \bigcup_{q \in \widehat{\delta}_N(q_0, u)} \delta_N(q, a) \\ &= \widehat{\delta}_N(q_0, ua) \\ &= \widehat{\delta}_N(q_0, w) \end{aligned}$$

Portanto, pode-se concluir que $w \in \mathcal{L}(A)$ se, e somente se, $w \in \mathcal{L}(A')$, ou seja, $L = \mathcal{L}(A')$ o que completa a prova. \square

O Teorema 10.1 mostra que toda linguagem reconhecida por um AFN também pode ser reconhecida por um AFD, assim as linguagens do AFN são realmente linguagem regulares.

Teorema 10.2 — Transformação AFD - AFN. Se $L = \mathcal{L}(A)$ para algum AFD A , então existe um AFN A' tal que $L = \mathcal{L}(A')$.

Demonstração. A prova é trivial e fica como exercício ao leitor. \square

Como consequência destes resultados segue o seguinte corolário que caracteriza as linguagens regulares em termos dos AFN.

Corolário 10.1 Uma linguagem L é regular se, e somente se, existe um AFN A tal que $L = \mathcal{L}(A)$.

Demonstração. Direto dos Teoremas 10.1 e 10.2. □

É importante destacar que o método de construção do AFD usado na prova do Teorema 10.1, conhecido como método de construção das partes introduzido por Rabin e Scott em [92], tem a característica de poder vir a produzir durante sua execução alguns estados inacessíveis⁴ no AFD resultante. Outro ponto é que em alguns cenários pode ser tornar impraticável, pois se o AFN de entrada possuir n estados, o AFD resultante do método terá 2^n estados, ou seja, o crescimento no número de estados do AFD resultante do método cresce proposicional a uma potência de 2, o que rapidamente gera um número exponencialmente grande de estados.

Algoritmo 1: Algoritmo para converter AFN em AFD sem estados inacessíveis.

Entrada: Um AFN $A = \langle Q, \Sigma, \delta_N, q_0, F \rangle$

Saída: Um AFD $A' = \langle Q', \Sigma, \delta, \{q_0\}, F' \rangle$

```

1  início
2  | Inicialize os conjuntos  $Q_u$  e  $Q'$  com um estado rotulado por  $\{q_0\}$ 
3  | Inicialize o conjunto  $F'$  como sendo vazio
4  | repita
5  |   | Selecione um estado  $X \in Q_u$ 
6  |   | para cada  $a \in \Sigma$  faça
7  |   |   | Determine o conjunto  $Y = \bigcup_{q \in X} \delta_N(q, a)$ 
8  |   |   | se  $Y \notin Q'$  então
9  |   |   |   | Adicione um estado rotulado por  $Y$  em  $Q'$  e em  $Q_u$ 
10 |   |   |   | Defina a transição  $\delta(X, a) = Y$ 
11 |   |   | senão
12 |   |   |   | Defina a transição  $\delta(X, a) = Y$ 
13 |   |   | fim
14 |   |   | se  $Y \cap F \neq \emptyset$  então
15 |   |   |   | Adicione  $Y$  ao conjunto  $F'$ 
16 |   |   | fim
17 |   | fim
18 |   | Remova  $X$  de  $Q_u$ 
19 | até  $Q_u = \emptyset$ ;
20 | retorna  $A' = \langle Q', \Sigma, \delta, \{q_0\}, F' \rangle$ 
21 fim

```

O Algoritmo 1 é uma melhoria do método proposto por Rabin e Scott (ver [92]), a melhoria no algoritmo se dá pelo fato dele não construir simplesmente o conjunto $\mathcal{P}(Q)$ para o AFD de saída, em vez

⁴Um estado q em um AFD é dito inacessível se não existe um $w \in \Sigma^*$ tal que $\widehat{\delta}(q_0, w) = q$. Vale também ressaltar como destaque em [15, 49] que estados inacessíveis não aumentam o poder de computação nos AFD.

disso, ele constrói iterativamente um conjunto de estados $Q' \subseteq \wp(Q)$, que no pior caso⁵ $Q' = \wp(Q)$, os exemplos a seguir mostra como é significativa a diferença entre os AFD produzidos pelo método original de Rabin e Scott e pelo Algoritmo 1.

■ **Exemplo 10.18** Usando o método original de Rabin e Scott sobre o AFN representado pelo grafo de transição da Figura 10.7 gera o AFD $M = \langle \wp(Q), \{a, b\}, \delta, \{q_0\}, F' \rangle$ onde:

$$\wp(Q) = \left\{ \{q_0\}, \{q_1\}, \{q_2\}, \{q_3\}, \{q_0, q_1\}, \{q_0, q_2\}, \{q_0, q_3\}, \{q_1, q_2\}, \{q_1, q_3\}, \{q_2, q_3\}, \right. \\ \left. \{q_0, q_1, q_2\}, \{q_0, q_1, q_3\}, \{q_0, q_2, q_3\}, \{q_1, q_2, q_3\}, \{q_0, q_1, q_2, q_3\}, \emptyset \right\}$$

A função de transição δ é definida como:

δ	a	b
$\{q_0\}$	$\{q_0, q_1\}$	\emptyset
$\{q_1\}$	\emptyset	$\{q_2\}$
$\{q_2\}$	\emptyset	$\{q_2, q_3\}$
$\{q_3\}$	$\{q_1, q_3\}$	\emptyset
$\{q_0, q_1\}$	$\{q_0, q_1\}$	$\{q_2\}$
$\{q_0, q_2\}$	$\{q_0, q_1\}$	$\{q_2, q_3\}$
$\{q_0, q_3\}$	$\{q_0, q_1, q_3\}$	\emptyset
$\{q_1, q_2\}$	\emptyset	$\{q_2, q_3\}$
$\{q_1, q_3\}$	$\{q_1, q_3\}$	$\{q_2\}$
$\{q_2, q_3\}$	$\{q_1, q_3\}$	$\{q_2, q_3\}$
$\{q_0, q_1, q_2\}$	$\{q_0, q_1\}$	$\{q_2, q_3\}$
$\{q_0, q_1, q_3\}$	$\{q_0, q_1, q_3\}$	$\{q_2\}$
$\{q_0, q_2, q_3\}$	$\{q_0, q_1, q_3\}$	$\{q_2, q_3\}$
$\{q_1, q_2, q_3\}$	$\{q_1, q_3\}$	$\{q_2, q_3\}$
$\{q_0, q_1, q_2, q_3\}$	$\{q_0, q_1, q_3\}$	$\{q_2, q_3\}$
\emptyset	\emptyset	\emptyset

e o conjunto $F' = \{\{q_1\}, \{q_1, q_2\}, \{q_1, q_3\}, \{q_0, q_1, q_2\}, \{q_0, q_1, q_3\}, \{q_1, q_2, q_3\}, \{q_0, q_1, q_2, q_3\}\}$.

■ **Exemplo 10.19** Usando o Algoritmo 1 tendo o AFN representado pelo grafo de transição da Figura 10.7 como entrada será obtido o AFD $M = \langle \{\{\{q_0\}, \{q_0, q_1\}, \{q_2\}, \{q_2, q_3\}, \{q_1, q_3\}\}, \emptyset\}, \{a, b\}, \delta, s_0, F' \rangle$ onde $F' = \{\{q_0, q_1\}, \{q_1, q_3\}\}$ e a função de transição δ é definida pela tabela a seguir.

δ	a	b
$\{q_0\}$	$\{q_0, q_1\}$	\emptyset
$\{q_0, q_1\}$	$\{q_0, q_1\}$	$\{q_2\}$
$\{q_2\}$	\emptyset	$\{q_2, q_3\}$
$\{q_2, q_3\}$	$\{q_1, q_3\}$	$\{q_2, q_3\}$
$\{q_1, q_3\}$	$\{q_1, q_3\}$	$\{q_2\}$
\emptyset	\emptyset	\emptyset

⁵A expressão “no pior caso” é típica da análise de algoritmos, em momentos futuros essa ideia de pior caso será melhor desenvolvida neste manuscrito.

10.3 Autômatos Finitos Não-determinísticos com Movimentos Vazios

Os λ -Autômatos Finitos Não-determinísticos, ou simplesmente, λ -AFN são como dito em [74], uma generalização do modelo de AFN em que são permitidas transições entre estados diferentes usando (ou consumindo) a palavra vazia, tais transições recebem o nome de λ -transições, a seguir tais autômatos serão apresentados formalmente.

Definição 10.11 — λ -Autômatos Finitos Não-determinísticos. Um λ -AFN é uma estrutura $A = \langle Q, \Sigma, \delta_N, q_0, F \rangle$ onde: Q, Σ, q_0 e F são da mesma forma que na Definição 10.1, já $\delta_N : Q \times (\Sigma \cup \{\lambda\}) \rightarrow \wp(Q)$ é uma função total (chamada λ -função de transição não determinística).

A representação usando grafos de transição dos λ -AFN é similar a representação dos AFN da seção anterior, a única diferença é que podem haver transições rotuladas pelo símbolo λ , isto é, podem existir no grafo arestas entre vértices que são rotuladas por λ , e o mesmo vale para a representação das árvores de computação.

■ **Exemplo 10.20** A estrutura $A = \langle \{q_0, q_1, q_2\}, \{0, 1\}, \delta_N, q_0, \{q_0\} \rangle$ com δ_N sendo especificada pela Tabela 10.2 a seguir é um λ -AFN.

$Q \backslash \Sigma \cup \{\lambda\}$	0	1	λ
q_0	\emptyset	\emptyset	$\{q_1\}$
q_1	$\{q_1\}$	$\{q_2\}$	$\{q_2\}$
q_2	$\{q_2\}$	$\{q_2\}$	$\{q_0, q_2\}$

Tabela 10.2: Tabela de transição para a função δ_N do AFN no Exemplo 10.20.

Observação 10.3 Uma interpretação para as transições da forma $\delta_N(q, \lambda) = X$ é que a unidade de controle do autômato consegue mudar seu estado interno q para um subconjunto de estados X sem precisar acessar a memória.

■ **Exemplo 10.21** O grafo de transição representado na Figura 10.10 a seguir é uma representação para o λ -AFN do Exemplo 10.20.

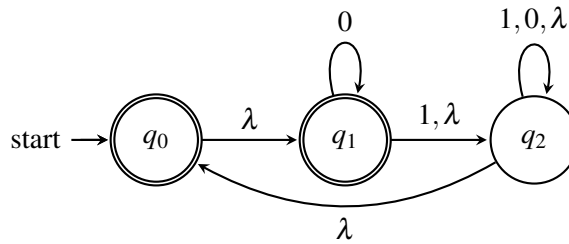


Figura 10.10: Grafo de transição do λ -AFN do Exemplo 10.20.

Note porém que a definição da função de transição δ_N garante que as transições em um λ -AFN acontecem apenas em duas situações, a primeira em relação símbolos individuais do alfabeto Σ e a

segunda com relação a palavra vazia, assim não existe uma forma de computar uma palavra w de forma que $|w| > 1$. A saída para contorna esse fato é estender a função de transição do autômato, similarmente ao que é feito para os AFD e AFN, para isso entretanto, é necessária algumas definições adicionais.

Definição 10.12 — Função δ_λ . Seja $A = \langle Q, \Sigma, \underline{\delta}_N, q_0, F \rangle$ um λ -AFN, então a função $\delta_\lambda : Q \rightarrow \wp(Q)$ é definida como,

$$\delta_\lambda(q) = \bigcup_{i=0}^n \lambda\text{-fecho}^i(q) \quad (10.17)$$

onde $n = \#Q - 1$ e

$$\lambda\text{-fecho}^0(q) = \{q\} \quad (10.18)$$

$$\lambda\text{-fecho}^i(q) = \bigcup_{q' \in \lambda\text{-fecho}^{i-1}(q)} \delta_N(q', \lambda) \quad (10.19)$$

■ **Exemplo 10.22** Considere o λ -AFN da Figura 10.10 tem-se para o estado q_1 que,

$$\lambda\text{-fecho}^2(q_1) = \bigcup_{q' \in \lambda\text{-fecho}^1(q_1)} \delta_N(q', \lambda) \quad (10.20)$$

desenvolvendo $q' \in \lambda\text{-fecho}^1(q_0)$ tem-se que,

$$\lambda\text{-fecho}^1(q_1) = \bigcup_{q' \in \lambda\text{-fecho}^0(q_1)} \delta_N(q', \lambda)$$

mas,

$$\lambda\text{-fecho}^0(q_1) = \{q_1\}$$

assim,

$$\lambda\text{-fecho}^1(q_1) = \{q_2\}$$

substituindo tal resultado na Equação 10.20 tem-se que,

$$\begin{aligned} \lambda\text{-fecho}^2(q_1) &= \bigcup_{q' \in \{q_2\}} \delta_N(q', \lambda) \\ &= \{q_2, q_0\} \end{aligned}$$

logo $\delta_\lambda(q_1) = \{q_0, q_1, q_2\}$.

Uma interpretação semântica para a função δ_λ é que ela representa a resposta ao questionamento:

“Estando no estado q e executando n λ -transições qual subconjunto de estados a unidade central do autômato irá assumir?”. Assim como acontecer com as funções de transição a função δ_λ pode ser estendida, a seguir é exposto tal extensão.

Definição 10.13 — Função $\widehat{\delta_\lambda}$. Seja $A = \langle Q, \Sigma, \underline{\delta_N}, q_0, F \rangle$ um λ -AFN, então a função $\widehat{\delta_\lambda} : \wp(Q) \rightarrow \wp(Q)$ é definida como,

$$\widehat{\delta_\lambda}(X) = \bigcup_{q \in X} \delta_\lambda(q) \quad (10.21)$$

■ **Exemplo 10.23** Considere o λ -AFN da Figura 10.10 tem-se para o conjunto $\{q_1, q_2\}$ que,

$$\begin{aligned} \widehat{\delta_\lambda}(\{q_1, q_2\}) &= \bigcup_{q \in \{q_1, q_2\}} \delta_\lambda(q) \\ &= \delta_\lambda(q_1) \cup \delta_\lambda(q_2) \\ &= \{q_0, q_1, q_2\} \cup \{q_0, q_2, q_1\} \\ &= \{q_0, q_2, q_1\} \end{aligned}$$

Agora usando as definições de δ_λ e $\widehat{\delta_\lambda}$ pode-se apresentar a extensão da função de transição dos λ -AFN.

Definição 10.14 — λ -Transição não-determinística estendida. Seja $A = \langle Q, \Sigma, \underline{\delta_N}, q_0, F \rangle$ um λ -AFN a função $\underline{\delta_N}$ é estendido para a função $\widehat{\delta_N} : Q \times \Sigma^* \rightarrow \wp(Q)$ definida pela seguinte recursão:

$$\widehat{\delta_N}(q, \lambda) = \delta_\lambda(q) \quad (10.22)$$

$$\widehat{\delta_N}(q, wa) = \bigcup_{q' \in \widehat{\delta_N}(q, w)} \widehat{\delta_\lambda}(\underline{\delta_N}(q', a)) \quad (10.23)$$

■ **Exemplo 10.24** Considere o λ -AFN da Figura 10.10 tem-se a seguinte computação para a palavra “10”:

$$\begin{aligned} \widehat{\delta_N}(q_0, 10) &= \bigcup_{q' \in \widehat{\delta_N}(q_0, 1)} \widehat{\delta_\lambda}(\underline{\delta_N}(q', 0)) \\ &= \bigcup_{q' \in \widehat{\delta_N}(q_0, 1)} \widehat{\delta_\lambda}(\underline{\delta_N}(q', 0)) \end{aligned} \quad (10.24)$$

mas,

$$\begin{aligned}
 \widehat{\delta_N}(q_0, 1) &= \bigcup_{q' \in \widehat{\delta_N}(q_0, \lambda)} \widehat{\delta_\lambda}(\delta_N(q', 1)) \\
 &= \bigcup_{q' \in \delta_\lambda(q_0)} \widehat{\delta_\lambda}(\delta_N(q', 1)) \\
 &= \bigcup_{q' \in \{q_0, q_1, q_2\}} \widehat{\delta_\lambda}(\delta_N(q', 1)) \\
 &= \widehat{\delta_\lambda}(\{q_1, q_2\}) \\
 &= \{q_0, q_1, q_2\}
 \end{aligned} \tag{10.25}$$

substituindo o valor da Equação (10.25) na Equação (10.24) tem-se que,

$$\begin{aligned}
 \widehat{\delta_N}(q_0, 10) &= \bigcup_{q' \in \{q_0, q_1, q_2\}} \widehat{\delta_\lambda}(\delta_N(q', 0)) \\
 &= \{q_0, q_1, q_2\}
 \end{aligned}$$

Assim como para o caso dos AFN uma palavra qualquer $w \in \Sigma^*$ será dita aceita por um λ -AFN quando a computação da palavra w para em pelo menos um estado final, ou seja, w é reconhecida pelo λ -AFN sempre que $\widehat{\delta_N}(q_0, w) \cap F \neq \emptyset$, e assim pode-se definir formalmente a noção de linguagem para os λ -AFN.

Definição 10.15 — Linguagem de um λ -AFN. Seja $A = \langle Q, \Sigma, \delta_N, q_0, F \rangle$ um λ -AFN a linguagem aceita por A , denotado por $\mathcal{L}(A)$, corresponde ao seguinte conjunto.

$$\mathcal{L}(A) = \{w \in \Sigma^* \mid \widehat{\delta_N}(q_0, w) \cap F \neq \emptyset\} \tag{10.26}$$

Os aspectos relacionados a mostrar que um λ -AFN reconhece uma linguagem L são similares ao mesmo aspectos com respeito aos AFN.

■ **Exemplo 10.25** O λ -AFN representado pelo grafo de transição da Figura 10.11 a seguir reconhece a linguagem $L = \{w \in \{1, 2, 3\}^* \mid w = 1^i 2^j 3^k \text{ com } i, j, k \in \mathbb{N}\}$.

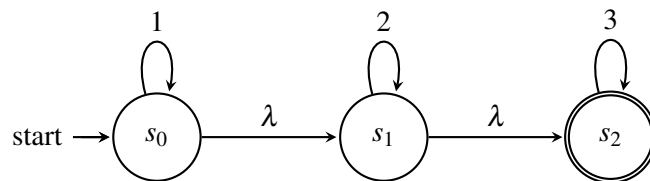


Figura 10.11: Grafo de transição do λ -AFN do Exemplo 10.25.

■ **Exemplo 10.26** O λ -AFN representado pelo grafo de transição esboçado pela Figura 10.12, aceita a linguagem $L = \{uv \mid u \in \{a\}^*, |u|_a = 2k \text{ ou } |u|_a = 3k, v = bx, x \in \{a, b\}^*, k \in \mathbb{N}\}$.

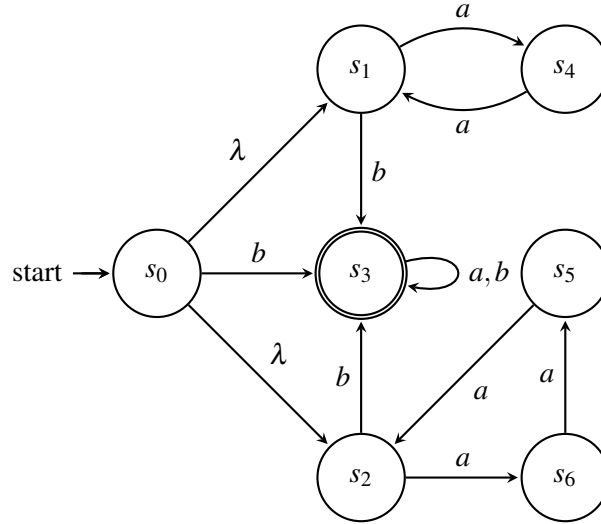


Figura 10.12: Grafo de transição do λ -AFN do Exemplo 10.26.

Teorema 10.3 — Transformação λ -AFN-AFN. Se $L = \mathcal{L}(A)$ para algum λ -AFN A , então existe um AFD A' tal que $L = \mathcal{L}(A')$.

Demonstração. Suponha que $L = \mathcal{L}(A)$ para algum λ -AFN $A = \langle Q, \Sigma, \delta_N, q_0, F \rangle$ agora defina o seguinte o autômato $A' = \langle \wp(Q), \Sigma, \delta, \delta_\lambda(q_0), F' \rangle$ onde para todo $X \in \wp(Q)$ tem-se que $X \in F'$ se, e somente se, $X \cap F \neq \emptyset$, e além disso, para todo $X \in \wp(Q)$ e $a \in \Sigma$ tem-se:

$$\delta(X, a) = \bigcup_{q \in X} \widehat{\delta_\lambda}(\delta_N(q, a)) \quad (10.27)$$

por essa construção obviamente esse autômato é um AFD⁶. Agora será mostrado por indução sobre o tamanho de $w \in \Sigma^*$ que:

$$\widehat{\delta}(\delta_\lambda(q_0), w) = \widehat{\delta_N}(q_0, w)$$

• **Base da indução:**

Quando $|w| = 0$ isto é $w = \lambda$ tem-se trivialmente pela definição das funções de transição estendidas que,

$$\begin{aligned} \widehat{\delta}(\delta_\lambda(q_0), \lambda) &= \delta_\lambda(q_0) \\ &= \widehat{\delta_N}(q_0, \lambda) \end{aligned}$$

⁶A prova desse fato fica como exercício ao leitor.

- **Hipótese indutiva (HI):**

Suponha que para todo $w \in \Sigma^*$ com $|w| \geq 0$ tem-se que $\widehat{\delta}(\delta_\lambda(q_0), w) = \widehat{\delta_N}(q_0, w)$.

- **Passo indutivo:**

Dado $w = ua$ com $u \in \Sigma^*$, $|u| \geq 0$ e $a \in \Sigma$ tem-se que,

$$\begin{aligned}
 \widehat{\delta}(\delta_\lambda(q_0), w) &= \widehat{\delta}(\delta_\lambda(q_0), ua) \\
 &= \delta(\widehat{\delta}(\delta_\lambda(q_0), u), a) \\
 &\stackrel{(HI)}{=} \delta(\widehat{\delta_N}(q_0, u), a) \\
 &\stackrel{Eq.(10.27)}{=} \bigcup_{q \in \widehat{\delta_N}(q_0, u)} \widehat{\delta}_\lambda(\delta_N(q, a)) \\
 &= \widehat{\delta_N}(q_0, ua) \\
 &= \widehat{\delta_N}(q_0, w)
 \end{aligned}$$

Portanto, pode-se concluir que $w \in \mathcal{L}(A)$ se, e somente se, $w \in \mathcal{L}(A')$, ou seja, $L = \mathcal{L}(A')$ o que completa a prova. \square

Teorema 10.4 — Transformação AFD- λ -AFN. Se $L = \mathcal{L}(A)$ para algum AFD A , então existe um λ -AFN A' tal que $L = \mathcal{L}(A')$.

Demonstração. Trivial e ficará como exercício ao leitor. \square

Imediatamente a estes resultados pode-se enunciar uma nova caracterização para as linguagens regulares, esta caracterização se baseado na ideia de λ -AFN é apresentada pelo corolário a seguir.

Corolário 10.2 Uma linguagem L é regular se, e somente se, existe um λ -AFN A tal que $L = \mathcal{L}(A)$.

Demonstração. (\Rightarrow) Assuma que L é regular, assim por definição existe um AFD A' tal que $L = \mathcal{L}(A')$, entretanto, pelo Teorema 10.4 existe um λ -AFN A tal que $L = \mathcal{L}(A)$. (\Leftarrow) Suponha que $L = \mathcal{L}(A)$ para algum λ -AFN A , agora pelo Teorema 10.3 existe um AFN A' tal que $L = \mathcal{L}(A')$, e portanto, L é regular. \square

Como destacado por Hopcroft *et. al.* em [49], uma importante interpretação sobre o Corolário 10.2 é que ele estabelece que λ -transições não aumentam o poder computacional dos autômatos finitos, isto é, autômatos finitos sem essa comodidade (as λ -transições) ainda são capazes de reconhecer exatamente a mesma classe de linguagens.

Observação 10.4 Note que o Corolário 10.2 estabelece que a existência de λ -transições não aumenta o poder de computação dos autômatos finitos.

Assim como para o caso da transformação de AFN em AFD apresentada por Rabin e Scott em [92], o processo de usar a construção do conjunto das partes no Teorema 10.3 possui a desvantagem de gera estados inacessíveis. Mas como discutido em [12, 13, 49, 61], algumas simples modificações no método exposto na demonstração do Teorema 10.3 gera o Algoritmo 1 (apresentado a seguir), tal algoritmo possui a capacidade de remover as λ -transições e aos mesmo tempo não são produzidos os estados inacessíveis, a seguir é apresentado este novo algoritmo.

■ **Exemplo 10.27** Aplicando o Algoritmo 2 ao λ -AFN do Exemplo 10.26 é obtido como saída o AFD $D = \langle \{A_0, A_1, A_2, A_3, A_4, A_5, A_6, A_7, \emptyset\}, \{a, b\}, \delta, \{s_0, s_1, s_2\}, F' \rangle$ onde tem-se

$$\begin{aligned} A_0 &= \{s_0, s_1, s_2\} \\ A_1 &= \{s_4, s_6\} \\ A_2 &= \{s_3\} \\ A_3 &= \{s_1, s_5\} \\ A_4 &= \{s_2, s_4\} \\ A_5 &= \{s_1, s_6\} \\ A_6 &= \{s_4, s_5\} \\ A_7 &= \{s_1, s_2\} \end{aligned}$$

sendo $F' = \{A_2\}$ e com a função δ descrito pela tabela a seguir.

δ	a	b
A_0	A_1	A_2
A_1	A_3	\emptyset
A_2	A_2	A_2
A_3	A_4	A_2
A_4	A_5	A_2
A_5	A_6	A_2
A_6	A_7	\emptyset
A_7	A_1	A_2
\emptyset	\emptyset	\emptyset

Observação 10.5 Argumentações sobre a corretude e a completude do Algoritmo 2 podem ser consultadas em [49].

Algoritmo 2: Algoritmo para remoção de λ -transições de um λ -AFN.

Entrada: Um λ -AFN $A = \langle Q, \Sigma, \delta_N, q_0, F \rangle$
Saída: Um AFD $A' = \langle Q', \Sigma, \delta, \delta_\lambda(q_0), F' \rangle$

```

1  início
2  | Inicialize o conjuntos  $Q_u$  com um estado rotulado por  $\delta_\lambda(q_0)$ 
3  | Inicialize o conjuntos  $Q'$  com um estado rotulado por  $\delta_\lambda(q_0)$ 
4  | Inicialize o conjunto  $F'$  como sendo vazio
5  | repita
6  | | Selecione um estado  $X \in Q_u$ 
7  | | para cada  $a \in \Sigma$  faça
8  | | | Determine o conjunto  $Y = \widehat{\delta_\lambda} \left( \bigcup_{q \in X} \delta_N(q, a) \right)$ 
9  | | | se  $Y \notin Q'$  então
10 | | | | Adicione um estado rotulado por  $Y$  em  $Q'$ 
11 | | | | Adicione um estado rotulado por  $Y$  em  $Q_u$ 
12 | | | | Defina a transição  $\delta(X, a) = Y$ 
13 | | | senão
14 | | | | Defina a transição  $\delta(X, a) = Y$ 
15 | | | fim
16 | | fim
17 | | Remova  $X$  de  $Q_u$ 
18 | até  $Q_u = \emptyset$ ;
19 | para cada  $X \in Q'$  faça
20 | | se  $X \cap F \neq \emptyset$  então
21 | | | Adicione  $X$  ao conjunto  $F'$ 
22 | | fim
23 | fim
24 | retorna  $A' = \langle Q', \Sigma, \delta, \delta_\lambda(q_0), F' \rangle$ 
25 fim
  
```



Nota 10.3 Nestas últimas seções foram usados os símbolos δ , δ_N e δ_λ para denotar as funções de transições dos AFD, AFN e λ -AFN respectivamente, entretanto, isso foi feito apenas para tornar o texto mais didático e ajudar na conversão entre os tipos de autômatos, mas é comum encontrar na literatura (ver [15, 49, 61]) que independente do tipo de autômato sua função de transição é denotada apenas por δ .

10.4 Teorema Myhill-Nerode e a Minimização de AFD

Até agora este manuscrito se preocupou com a tarefa de saber se uma linguagem pode ou não ser reconhecida por um autômato finito, seja ele determinístico ou não-determinístico. Nesta seção será apresentada ao leitor a questão de eficiência no reconhecimento de linguagens em relação aos autômatos finitos, aqui será mostrado que o problema de encontrar um menor AFD que reconhece uma linguagem L é decidível.

Na teoria dos autômatos quando se usa a palavra “menor”, se está querendo dizer simplesmente aquele com o menor número possível de estados, ou seja, o AFD mínimo. Mais adiante será aqui provado, que esse AFD mínimo é único a menos de isomorfismo, ou seja, se dois AFD reconhecem a mesma linguagem, cada um tendo o menor número possível de estados, então eles são isomórficos. Isso significa que cada linguagem regular está associada com um AFD mínimo.

O resultado que estabelece a existência de um AFD mínimo recebe o nome de **Teorema Myhill-Nerode**, em homenagem aos matemáticos John Myhill⁷ (1923-1987) e Anil Nerode (1932-), que o provaram na Universidade de Chicago em 1958 no artigo [85], de forma geral tal resultado fornece condições suficientes e necessárias para que uma linguagem L seja regular, para construir tal resultado antes é necessário considerar algumas definições básicas e alguns resultados auxiliares.

Definição 10.16 — A família \mathcal{H}_L . Seja L uma linguagem qualquer^a sobre o alfabeto Σ , para qualquer palavra w é definido o conjunto $L_w = \{x \mid wx \in L\}$. A família $\{L_w \mid w \in \Sigma^*\}$ construída sobre L será denotada por \mathcal{H}_L , ou seja, $\mathcal{H}_L = \{L_w \mid w \in \Sigma^*\}$

^aNão necessariamente regular.

Com respeito aos conjuntos L_w o leitor mais atento pode notar que $L_\lambda = L$, além disso, os conjuntos L_w também apresentam a seguinte propriedade básica.

Proposição 10.1 Dado $L \subseteq \Sigma^*$. Se $L_w = L_{w'}$, então $L_{wa} = L_{w'a}$ para todo $a \in \Sigma$.

Demonstração. Suponha que $L_w = L_{w'}$, assim para todo $a \in \Sigma^*$ tem-se que:

$$\begin{aligned} x \in L_{wa} &\iff wax \in L \\ &\iff ax \in L_w \\ &\stackrel{Hip.}{\iff} ax \in L_{w'} \\ &\iff x \in L_{w'a} \end{aligned}$$

concluindo a prova. □

Um fato importante sobre AFD que será usado a seguir e que não foi mencionado diretamente até agora é o exposto pelo resultado a seguir.

Proposição 10.2 Se $A = \langle Q, \Sigma, \delta, q_0, F \rangle$ é um AFD, então $\widehat{\delta}(q_0, uv) = \widehat{\delta}(\widehat{\delta}(q_0, u), v)$ para todo $u, v \in \Sigma^*$.

Demonstração. A prova é por indução sobre o tamanho da palavra uv e ficará como exercício ao leitor. □

⁷O professor Myhill também é conhecido por seu Teorema de isomorfismo[82], que pode ser visto como um análogo dentro da teoria da computabilidade ao teorema de Cantor–Bernstein–Schroeder e pelo famoso pelo Teorema de Rice–Myhill–Shapiro, mais comumente conhecido como Teorema de Rice [15, 93].

O lema a seguir mostra que $\#\mathcal{H}_L$ é na verdade um limite inferior para o número de estados em um AFD.

Lema 10.1 Se $L = \mathcal{L}(A)$ para algum AFD $A = \langle Q, \Sigma, \delta, q_0, F \rangle$, então $\#\mathcal{H}_L \leq \#Q$.

Demonstração. Suponha que $L = \mathcal{L}(A)$ para algum AFD $A = \langle Q, \Sigma, \delta, q_0, F \rangle$, agora para todo $q \in Q$ defina um novo AFD A_q igual ao anterior em todos os aspectos menos no estado inicial pois este será o estado q , ou seja, $A_q = \langle Q, \Sigma, \delta, q, F \rangle$. Agora para toda palavra $w \in \Sigma^*$ suponha que $\widehat{\delta}(q_0, w) = q$, por definição note que,

$$\begin{aligned}
 x \in L_w & \stackrel{\text{Def. 10.16}}{\iff} wx \in L \\
 & \iff wx \in \mathcal{L}(A) \\
 & \iff \widehat{\delta}(q_0, wx) \in F \\
 & \stackrel{\text{Prop. 10.2}}{\iff} \widehat{\delta}(\widehat{\delta}(q_0, w), x) \in F \\
 & \stackrel{\text{Hip.}}{\iff} \widehat{\delta}(q, x) \in F \\
 & \iff x \in \mathcal{L}(A_q)
 \end{aligned}$$

Dessa forma tem-se que $\mathcal{L}(A_q) = L_w$, e obviamente $\mathcal{L}(A_{q_0}) = L$. Desde que A é fixo, tem-se que L_w depende apenas do estado obtido quando a computação w começa em q_0 , e assim o número de L_w distintos, ou seja, os elementos de \mathcal{H}_L não pode ser maior que o número de estados em A , portanto, $\#\mathcal{H}_L \leq \#Q$. \square

O próximo lema estabelece que para alguma linguagem L no caso \mathcal{H}_L ser finito, então sua cardinalidade será o limite superior no número de estados em um AFD capaz de reconhecer L .

Lema 10.2 Seja $L \subseteq \Sigma^*$. Se \mathcal{H}_L é finito, então existe um AFD A_L tal que $L = \mathcal{L}(A_L)$ e A_L possui exatamente $\#\mathcal{H}_L$ estados.

Demonstração. Dado $L \subseteq \Sigma^*$ assumamos que \mathcal{H}_L é finito, dito isto pode-se construir o seguinte AFD $A_L = \langle \mathcal{H}_L, \Sigma, \delta, q_0, F \rangle$ onde:

$$q_0 = L \tag{10.28}$$

e para todo $L_w \in \mathcal{H}_L$ e $a \in \Sigma$ tem-se

$$\delta(L_w, a) = L_{wa} \tag{10.29}$$

e

$$F = \{L_w \in \mathcal{H}_L \mid \lambda \in L_w\} \quad (10.30)$$

sobre a definição de A_L por indução sobre o tamanho de $w \in \Sigma^*$ pode-se facilmente verificar que,

$$\widehat{\delta}(q_0, w) = L_w \quad (10.31)$$

além disso, claramente A_L possui exatamente $\#\mathcal{H}_L$ estados, dito isto, note que:

$$\begin{aligned} w \in L & \stackrel{\text{Def. 10.16}}{\iff} \lambda \in L_w \\ & \stackrel{\text{Eq. (10.30)}}{\iff} L_w \in F \\ & \stackrel{\text{Eq. (10.31)}}{\iff} \widehat{\delta}(q_0, w) \in F \\ & \iff w \in \mathcal{L}(A_L) \end{aligned}$$

e portanto, $L = \mathcal{L}(A_L)$ concluindo assim a prova. \square

Definição 10.17 — Dimensão de AFD. Seja $A = \langle Q, \Sigma, \delta', q_0, F \rangle$ um AFD a dimensão de A , denotado por $\dim(A)$, é igual a quantidade de estados em A , isto é, $\dim(A) = \#Q$.

Definição 10.18 — AFD Mínimo. Seja L uma linguagem regular tal que $L = \mathcal{L}(A)$ para algum um AFD A . O AFD A será dito ser mínimo se, e somente se, e para todo outro AFD B tal que $L = \mathcal{L}(B)$ tem-se que $\dim(A) \leq \dim(B)$.

O próximo resultado mostra que não existe nenhum autômato como menos estados que o AFD construído na prova do Lema 10.2, ou seja, o método de construção mostrado na na prova do Lema 10.2 gera o AFD mínimo de qualquer linguagem.

Lema 10.3 Seja L uma linguagem regular, assim o AFD A_L construído no Lema 10.2 é o único (a menos de isomorfismo) mínimo AFD que aceita L .

Demonstração. Suponha que existe outro AFD mínimo $N = \langle Q, \Sigma, \delta', q'_0, F' \rangle$ tal que $L = \mathcal{L}(N)$ diferente do AFD $A_L = \langle \mathcal{H}_L, \Sigma, \delta, L, F \rangle$ construído na prova do Lema 10.2. Agora será definida uma função $f : Q \rightarrow \mathcal{H}_L$ definida simplesmente como:

$$f(q) = \begin{cases} L, & \text{se } q = q_0 \\ \mathcal{L}(A_q), & \text{senão} \end{cases}$$

para todo $q \in Q$, onde $\mathcal{L}(A_q)$ é da mesma forma que na prova do Lema 10.1 onde o AFD fixo é exata-

mente A_L . Por esta definição é claro que:

- (1) Se $q_i \neq q_j$, então tem-se que $f(q_i) \neq f(q_j)$, conseqüentemente a função f é injetora.
- (2) f preserva a condição de estado inicial.
- (3) Para $a \in \Sigma$, $w \in \Sigma^*$ e $q, p \in Q$ assumamos que $\delta(q, a) = p$ e $f(q) = \mathcal{L}(A_q) = L_w$ assim para qualquer $u \in \Sigma^*$ tem-se que

$$\begin{aligned}
 u \in f(p) &\iff u \in \mathcal{L}(A_p) \\
 &\iff \widehat{\delta}(p, u) \in F \\
 &\iff \widehat{\delta}(\widehat{\delta}(q, a), u) \in F \\
 &\iff \widehat{\delta}(q, au) \in F \\
 &\iff au \in \mathcal{L}(A_q) \\
 &\iff au \in f(q) \\
 &\iff au \in L_w \\
 &\iff wau \in L \\
 &\iff u \in L_{wa}
 \end{aligned}$$

portanto, $f(p) = L_{wa}$, ou seja, f preserva transições⁸.

- (4) Agora note que pela definição de estado final e pela construção de A_L tem-se que

$$q \in F' \iff \widehat{\delta}(q, \lambda) \in F' \iff \lambda \in \mathcal{L}(A_q) \iff \mathcal{L}(A_q) \in F \iff f(q) \in F$$

portanto, a função f preserva estados finais.

- (5) Agora dado $w \in \Sigma^*$ assumamos que $\widehat{\delta}(q_0, w) = q$, assim pela prova do Lema 10.1 para todo $L_w \in \mathcal{H}_L$, ou seja, para todo $L_w \in \text{Ima}(f)$ tem-se que $L_w = \mathcal{L}(A_q)$, e portanto, pela definição de f tem-se que existe $q \in \text{Dom}(f)$ tal que $L_w = f(q)$, ou seja, f é sobrejetora.

Desde que f é injetora e sobrejetora tem-se que f é uma bijeção, e portanto, $\#Q = \#\mathcal{H}_L$, logo $\dim(N) = \dim(A_L)$. Agora desde que f preserva o estado inicial, os estados finais e as transições tem-se que f é um isomorfismo do AFN N para o AFD A_L , e portanto, eles são o mesmo AFD se diferenciando apenas pela rotulação dos seus estados. \square

Pode-se agora finalmente enunciar o Teorema Myhill-Nerode que estabelece uma caracterização para as linguagens regulares.

⁸Isto é o mesmo que dizer que $f(\delta'(q, a)) = \delta(f(q), a)$.

Teorema 10.5 — Teorema Myhill-Nerode. Uma linguagem $L \subseteq \Sigma^*$ será regular se, e somente se, \mathcal{H}_L é finito e existe um AFD mínimo A com exatamente $\#\mathcal{H}_L$ estados tal que $\mathcal{L}(A) = L$.

Demonstração. Direto dos Lemas 10.1, 10.2 e 10.3. \square

Apesar de extremamente elegante o resultado exposto pelo Teorema Myhill-Nerode apresentado anteriormente, não é um algoritmo explícito (no sentido de um conjunto finito de instruções) para construir um AFD mínimo equivalente a outro AFD dado, o que o teorema faz é construir para uma linguagem regular o autômato mínimo que a reconhece. Por outro lado o processo de obter um AFD mínimo a partir de um outro AFD dado é chamado de minimização [15], e o mesmo é baseado na ideia de relação de equivalência entre estados e do espaço quociente de estados em um AFD. Primeiro será apresentado a formalização matemática da construção do AFD quociente e depois o pseudo-código de tal construção.

Definição 10.19 — Estados Equivalentes. Seja $A = \langle Q, \Sigma, \delta, q_0, F \rangle$ um AFD. A relação de equivalência entre dois estados $q, q' \in Q$ será denotado por $q \equiv q'$ e será verdadeira quando para todo $w \in \Sigma^*$ tem-se que $\widehat{\delta}(q, w) \in F \iff \widehat{\delta}(q', w) \in F$.

O resultado a seguir diz que se dois estados são equivalentes, então seus sucessores também o são.

Lema 10.4 Dado um AFD $A = \langle Q, \Sigma, \delta, q_0, F \rangle$. Se $q \equiv q'$, então $\delta(q, a) \equiv \delta(q', a)$ com $a \in \Sigma$.

Demonstração. Dado $a \in \Sigma$, por contrapositiva será mostrado que:

$$\text{Se } \delta(q, a) \not\equiv \delta(q', a), \text{ então } q \not\equiv q'.$$

Inicialmente assuma que $\delta(q, a) \not\equiv \delta(q', a)$, assim pela Definição 10.19 existe um $w \in \Sigma^*$ tal que um dos dois casos a seguir acontece:

- (1) $\widehat{\delta}(\delta(q, a), w) \in F$ e $\widehat{\delta}(\delta(q', a), w) \notin F$ ou
- (2) $\widehat{\delta}(\delta(q, a), w) \notin F$ e $\widehat{\delta}(\delta(q', a), w) \in F$.

agora fazendo $w = \lambda$, para o primeiro caso (a prova é similar para o caso (2)) tem-se pela Definição 10.2 que:

$$\widehat{\delta}(\delta(q, a), w) \in F \text{ e } \widehat{\delta}(\delta(q', a), w) \notin F \iff \delta(q, a) \in F \text{ e } \delta(q', a) \notin F$$

e desde que $a \in \Sigma^*$ pela Definição 10.19 tem-se que $q \not\equiv q'$, o que conclui a prova da contrapositiva. Desde que a contrapositiva é verdadeira tem-se que afirmação original “Se $q \equiv q'$, então $\delta(q, a) \equiv \delta(q', a)$ ” é também verdadeira. \square

Usando a definição anterior, a seguir será apresentado a definição de AFD (ou colapso) quociente sobre um determinado AFD dado.

Definição 10.20 — AFD quociente. Seja $A = \langle Q, \Sigma, \delta, q_0, F \rangle$ um AFD. O AFD (ou colapso) quociente de A é o AFD $A_{/\equiv} = \langle Q_{/\equiv}, \Sigma, \delta^*, [q_0], F_{/\equiv} \rangle$ e para todo $q \in Q$ e $a \in \Sigma$ tem-se que $\delta^*([q], a) = [\delta(q, a)]$ e $F_{/\equiv} = \{[q] \mid q \in F\}$.

Teorema 10.6 Seja $A_{/\equiv} = \langle Q_{/\equiv}, \Sigma, \delta^*, [q_0], F_{/\equiv} \rangle$ o AFD quociente obtido a partir de um AFD $A = \langle Q, \Sigma, \delta, q_0, F \rangle$. Então para todo $w \in \Sigma^*$ tem-se que $\widehat{\delta^*}([q], w) = [\widehat{\delta}(q, w)]$.

Demonstração. Por indução sobre o tamanho de $w \in \Sigma^*$ será mostrado a seguinte igualdade $\widehat{\delta^*}([q], w) = [\widehat{\delta}(q, w)]$.

- **Base da indução:**

Quando $|w| = 0$ ($w = \lambda$) tem-se trivialmente que $\widehat{\delta^*}([q], \lambda) = [q] = [\widehat{\delta}(q, \lambda)]$.

- **Hipótese indutiva (HI):**

Suponha que para todo $w \in \Sigma^*$ com $|w| \geq 0$ tem-se que $\widehat{\delta^*}([q], w) = [\widehat{\delta}(q, w)]$

- **Passo indutivo:**

Dado $w = ua$ com $u \in \Sigma^*$, $|u| \geq 0$ e $a \in \Sigma$ tem-se que,

$$\begin{aligned}
 \widehat{\delta^*}([q], w) &= \widehat{\delta^*}([q], ua) \\
 &= \delta^*(\widehat{\delta^*}([q], u), a) \\
 &\stackrel{(HI)}{=} \delta^*([\widehat{\delta}(q, u)], a) \\
 &= [\widehat{\delta}(\widehat{\delta}(q, u), a)] \\
 &= [\widehat{\delta}(q, ua)] \\
 &= [\widehat{\delta}(q, w)]
 \end{aligned}$$

O que completa a prova. □

Teorema 10.7 Dado $A_{/\equiv} = \langle Q_{/\equiv}, \Sigma, \delta^*, [q_0], F_{/\equiv} \rangle$ o AFD quociente obtido a partir de um AFD $A = \langle Q, \Sigma, \delta, q_0, F \rangle$ tem-se que $q \in F \iff [q] \in F_{/\equiv}$

Demonstração. (\Rightarrow) Trivial pela própria definição de $F_{/\equiv}$. (\Leftarrow) É suficiente mostrar que se $q \equiv p$ e $q \in F$, então $p \in F$. Para provar isto, suponha que $q \equiv p$ e $q \in F$, mas note que $q \in F \iff \widehat{\delta}(q, \lambda) \in F$, mas como $q \equiv p$, por definição tem-se para todo $w \in \Sigma^*$ que $\widehat{\delta}(q, w) \in F \iff \widehat{\delta}(p, w) \in F$, assim no particular quando $w = \lambda$ tem-se que $\widehat{\delta}(p, \lambda) \in F$, mas $\widehat{\delta}(p, \lambda) = p$, portanto, $p \in F$. □

O próximo resultado mostra que dado um AFD A qualquer o AFD quociente A' de A tem a propriedade de reconhecer a mesma linguagem de A , ou seja, $\mathcal{L}(A) = \mathcal{L}(A')$, em outras palavras pode-se dizer simplesmente que, o processo de minimização preserva a linguagem do AFD.

Teorema 10.8 Dado $A_{/\equiv} = \langle Q_{/\equiv}, \Sigma, \delta^*, [q_0], F_{/\equiv} \rangle$ o AFD quociente obtido a partir de um AFD $A = \langle Q, \Sigma, \delta, q_0, F \rangle$ tem-se que $\mathcal{L}(A_{/\equiv}) = \mathcal{L}(A)$.

Demonstração. Basta notar que para qualquer $w \in \Sigma^*$ tem-se que:

$$\begin{aligned}
 w \in \mathcal{L}(A_{/\equiv}) &\iff \widehat{\delta^*}([q_0], w) \in F_{/\equiv} \\
 &\stackrel{\text{Teo. 10.6}}{\iff} [\widehat{\delta}(q_0, w)] \in F_{/\equiv} \\
 &\stackrel{\text{Teo. 10.7}}{\iff} \widehat{\delta}(q_0, w) \in F \\
 &\iff w \in \mathcal{L}(A)
 \end{aligned}$$

logo $\mathcal{L}(A_{/\equiv}) = \mathcal{L}(A)$. □

Agora é aceitável que o leitor possa se questionar sobre o que acontece quando se tenta colapsar um AFD que já é um AFD quociente. A resposta para esse fato é simples, não acontece nada, isto é, se $A_{/\equiv}$ é o AFD quociente, então o colapso dele é ele próprio, isto é, o AFD quociente é um ponto fixo⁹ com respeito ao processo de minimização dos AFD, a prova disto pode ser vista em [69].

Um outro ponto importante a ser destacado sobre o AFD quociente é que ele é mínimo a menos de isomorfismo [69], ou seja, o AFD mínimo produzido pela minimização será exatamente o mesmo AFD gerado pelo Teorema Myhill-Nerode (para detalhes veja [69]).

O pseudo-código de construção do AFD quociente, ou seja, o código capaz de realizar a minimização de autômatos está descrito no Algoritmo 3 a seguir, é importante que o leitor note que como $\delta^*([q], a) = [\delta(q, a)]$ é um caso particular do resultado obtido no Teorema 10.6, e pela condição suficiente e necessária apresentada no Teorema 10.7, tem-se que a corretude do Algoritmo 3 se resume a demonstrar o teorema a seguir.

Teorema 10.9 — Corretude do Algoritmo de Construção do Quociente. Considerando um AFD $A = \langle Q, \Sigma, \delta, q_0, F \rangle$, dado o Algoritmo 3 e os índices $j > i$ tem-se que, $M(i, j)$ está marcado com X se, e somente se, $q_i \neq q_j$.

Demonstração. A demonstração fica como exercício ao leitor. □

O exemplo a seguir ilustra a aplicação do Algoritmo 3 para minimizar um AFD dado.

⁹ Isso é o mesmo que dizer que $MIN(A_{/\equiv}) = A_{/\equiv}$, onde MIN é a minimização de autômatos.

Algoritmo 3: Algoritmo para construção do AFD quociente.

Entrada: Um AFD $A = \langle Q, \Sigma, \delta, q_0, F \rangle$
Saída: O AFD quociente $A_{/\equiv} = \langle Q_{/\equiv}, \Sigma, \delta^*, [q_0], F_{/\equiv} \rangle$

```

1  início
2  Defina uma matriz  $M$  de dimensão  $\#Q$  por  $\#Q$ 
3  para cada  $(i, j) \in \{0, \dots, \#Q - 1\} \times \{0, \dots, \#Q - 1\}$  tal que  $j > i$  faça
4      se  $q_i \in F, q_j \notin F$  ou  $q_i \notin F, q_j \in F$  então
5          Marque a posição  $M(i, j)$  com um  $X$ 
6      fim
7  fim
8  para cada  $(i, j) \in \{0, \dots, \#Q - 1\} \times \{0, \dots, \#Q - 1\}$  tal que  $j > i$  faça
9      se  $M(i, j)$  não foi marcado então
10         para cada  $a \in \Sigma$  faça
11             se  $\delta(q_i, a) \in F, \delta(q_j, a) \notin F$  ou  $\delta(q_i, a) \notin F, \delta(q_j, a) \in F$  então
12                 Marque a posição  $M(i, j)$  com um  $X$ 
13             fim
14         fim
15     fim
16 fim
17 Defina  $T = \{0, \dots, \#Q\}$ 
18 Defina  $Q_{/\equiv}$  como sendo vazio
19 enquanto  $T \neq \emptyset$  faça
20     Selecione o menor  $i \in T$ 
21     se  $\nexists [q] \in Q_{/\equiv}$  tal que  $q_i \in [q]$  então
22         Defina uma classe de equivalência  $[q_i] = \{q_i\}$ 
23         para todo  $j \in T$  tal que  $i \neq j$  faça
24             se  $M(i, j)$  não estiver marcado então
25                 Adicione  $q_j$  na classe de equivalência  $[q_i]$ 
26                 Remova  $j$  de  $T$ 
27             fim
28         fim
29     fim
30     Remova  $i$  de  $T$ 
31 fim
32 para cada  $([q], a) \in Q_{/\equiv} \times \Sigma$  faça
33     Defina  $\delta^*([q], a) = [\delta(q, a)]$ 
34 fim
35 Considere inicialmente  $F_{/\equiv}$  como vazio
36 para cada  $[q] \in Q_{/\equiv}$  faça
37     se  $q \in F$  então
38         Adicione  $[q]$  ao conjunto  $F_{/\equiv}$ 
39     fim
40 fim
41 retorna  $A_{/\equiv} = \langle Q_{/\equiv}, \Sigma, \delta^*, [q_0], F_{/\equiv} \rangle$ 
42 fim

```

■ **Exemplo 10.28** Aplicando o Algoritmo 3 ao AFD representado pelo grafo de transições esboçado pela Figura 10.13 a seguir.

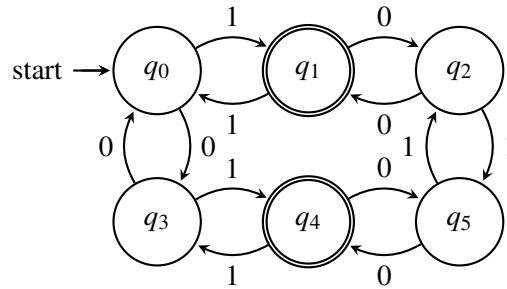


Figura 10.13: Grafo do AFD do Exemplo 10.25.

Inicialmente é gerada a matriz representada na Tabela 10.3 a seguir, em que a i -ésima linha (coluna) representa o estado q_{i-1} , e os espaços marcados com * são aqueles não considerados pelo algoritmo, as posições com \checkmark representam os estados não equivalentes, e as posições (i, j) em branco representam os estados equivalentes.

*	\checkmark	\checkmark		\checkmark	\checkmark
*	*	\checkmark	\checkmark		\checkmark
*	*	*	\checkmark	\checkmark	
*	*	*	*	\checkmark	\checkmark
*	*	*	*	*	\checkmark
*	*	*	*	*	*

Tabela 10.3: Matriz de equivalência M para o AFD 10.13.

A partir desta Tabela 10.3 o Algoritmo 3 gera as seguintes classes de equivalência: $[q_0] = \{q_0, q_3\}$, $[q_1] = \{q_1, q_4\}$ e $[q_2] = \{q_2, q_5\}$, depois define as transições e o conjunto de estados finais como esboçadas pelo grafo de Transição na Figura 10.14.

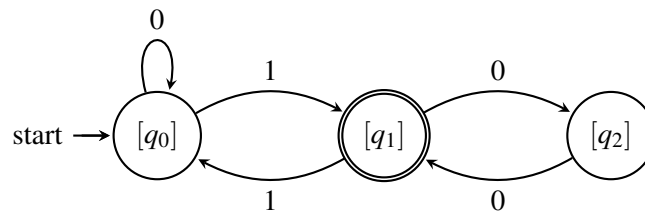


Figura 10.14: Grafo do AFD do Exemplo 10.25.

10.5 Máquinas de Mealy

Uma classe particularmente interessante de autômatos finitos são os chamados **autômatos com saída**, dos quais se destacam as máquinas (ou autômatos) de Mealy [73] e Moore [77], além de serem alicerces fundamentais da teoria dos autômatos finitos, tais máquina tem uma grande gama de aplicações práticas. Nesta seção será apresentada ao leitor uma formalização das máquinas de Mealy¹⁰ e será aqui

¹⁰Para leitores interessados nas máquinas de Moore é recomendado a leitura de [74].

trabalhado a ideia de usar tais máquinas como tradutoras, além disso, será apresentado o relacionamento de tais máquinas com os circuitos sequências.

Como dito em [52], diferente dos autômatos sem saída vistos nas seções 10.1, 10.2 e 10.3 as máquinas de Mealy são capazes de apresentar saídas mais complexas que as típicas saídas booleanas (aceita/rejeita) dos AFD, AFN e λ -AFN como será visto a seguir.

Em termos informais uma autômato com saída pode ser descrito como um máquina que possui duas memórias, a primeira memória (somente de leitura) é chamada de memória de entrada, e assim como nos autômatos das seções 10.1, 10.2 e 10.3, ela serve para guarda a palavra de entrada usada para o funcionamento da máquina. Já segunda memória (apenas de escrita) é chamada de **memória de saída**, sendo esta responsável por armazenar a palavra de saída, produzida pela máquina de Mealy. Formalmente uma máquina de Mealy é como se segue.

Definição 10.21 — Máquina de Mealy. Uma máquina (ou autômato) de Mealy é uma estrutura $A = \langle Q, \Sigma, \Lambda, \delta, \psi, q_0, F \rangle$ onde Q, Σ, q_0 e F são da mesma forma que na Definição 10.1, a função $\delta : Q \times \Sigma \rightarrow Q$ é uma função parcial, Λ é um alfabeto chamado **alfabeto de saída** e $\psi : Q \times \Sigma \rightarrow \Lambda^*$ é a função parcial de tradução da máquina de Mealy, de forma que $\psi(q, a)$ está definida se, e somente se, $\delta(q, a)$ está definida.

Dado qualquer máquina de Mealy tem-se que a função δ é responsável por acessar (ou usar) a memória de entrada (da mesma que em qualquer AFD), por outro lado, a função ψ é o mecanismo responsável por acessar (ou usar) a memória de saída.

■ **Exemplo 10.29** A estrutura $M = \langle \{q_0, q_1\}, \{a, b\}, \{0, 1, 2\}, \delta, \psi, q_0, \{q_0\} \rangle$ onde a função de transição é representada pela Tabela 10.4 e a função de tradução é representada pela Tabela 10.5, é uma máquina de Mealy.

$Q \backslash \Sigma$	a	b
	q_1	q_1
q_0	q_1	q_1
q_1	q_0	q_1

Tabela 10.4: Tabela da Função de transição da Máquina de Mealy do Exemplo 10.29.

$Q \backslash \Sigma$	a	b
	1	0
q_0	1	2
q_1	1	2

Tabela 10.5: Tabela da Função de tradução da Máquina de Mealy do Exemplo 10.29.

Pode-se utilizar a ideia de grafo de transição para representar visualmente as máquinas de Mealy a única diferença que esta representação terá em relação aos grafos que representam os AFD, é que as

arestas que ligam dois vértices q_i e q_j serão rotuladas com pares $(a, b) \in \Sigma \times \Lambda$ representando assim simultaneamente as funções $\delta(q_i, a) = q_j$ e $\psi(q_i, a) = b$.

■ **Exemplo 10.30** A representação do grafo de transição da máquina de Mealy do Exemplo 10.29 pode ser consultada na Figura 10.15.

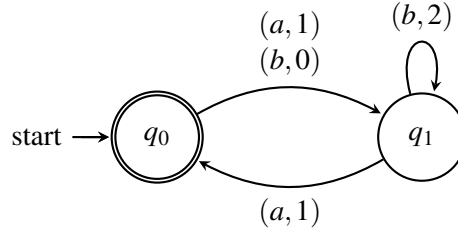


Figura 10.15: Representação da máquina de Mealy do Exemplo 10.29.

Note que a máquina de Mealy traduz (ou associa) cada transição da máquina em um símbolo do alfabeto de saída, ou seja, cada movimento no processo de computação da máquina é traduzido na saída da mesma. Obviamente $\hat{\delta}$ é definido da mesma forma que para os AFD, sendo assim basta agora estender a função ψ para que a máquina de Mealy possa de fato ser vista como um tradutor finito, e isto será feito a seguir.

Definição 10.22 — Extensão da função ψ . Seja $A = \langle Q, \Sigma, \Lambda, \delta, \psi, q_0, F \rangle$ uma máquina de Mealy a função ψ é estendida para a função $\hat{\psi} : Q \times \Sigma^* \rightarrow \Lambda^*$ usando recursividade como se segue.

$$\hat{\psi}(q, \lambda) = \lambda \quad (10.32)$$

$$\hat{\psi}(q, wa) = \hat{\psi}(q, w) \psi(\hat{\delta}(q, w), a) \quad (10.33)$$

■ **Exemplo 10.31** Considerando a máquina de Mealy do Exemplo 10.29 a tradução da palavra *bab* a partir do estado q_1 é dada seguinte forma,

$$\begin{aligned}
 \hat{\psi}(q_1, bab) &= \hat{\psi}(q_1, ba) \psi(\hat{\delta}(q_1, ba), b) \\
 &= \left(\hat{\psi}(q_1, b) \psi(\hat{\delta}(q_1, b), a) \right) \psi(\hat{\delta}(q_1, ba), b) \\
 &= \left(\left(\hat{\psi}(q_1, \lambda) \psi(q_1, b) \right) \psi(\hat{\delta}(q_1, b), a) \right) \psi(\hat{\delta}(q_1, ba), b) \\
 &= \lambda 210 \\
 &= 210
 \end{aligned}$$

Diferentemente dos AFD, AFN e λ -AFN o conjunto saída de qualquer máquina de Mealy não é visto como uma linguagem aceita pela máquina, em vez disso, a saída original de uma máquina de Mealy é exatamente a palavra escrita na memória de saída após a máquina termina a execução [52].

Definição 10.23 — Linguagem Traduzida - Máquina de Mealy. Dado uma máquina de Mealy $A = \langle Q, \Sigma, \Lambda, \delta, \psi, q_0, F \rangle$ a linguagem traduzida de A , denotado por $TRAD(A)$, é o conjunto de todas as traduções das palavras $w \in \Sigma^*$ realizadas a partir do estado inicial q_0 , ou seja, tem-se que:

$$TRAD(A) = \{ \hat{\psi}(q_0, w) \mid \hat{\delta}(q_0, w) \in F \} \quad (10.34)$$

■ **Exemplo 10.32** Usando indução sobre o tamanho das palavras de entrada não é difícil verificar que a máquina de Mealy representado pela Figura 10.16 traduz a linguagem $\{1^{2i+1}0^{2j+1}1 \mid i, j \geq 1\}$ para a linguagem $\{0^{4k_1+2}1^{3k_2+5} \mid k_1, k_2 \in \mathbb{N}\}$.

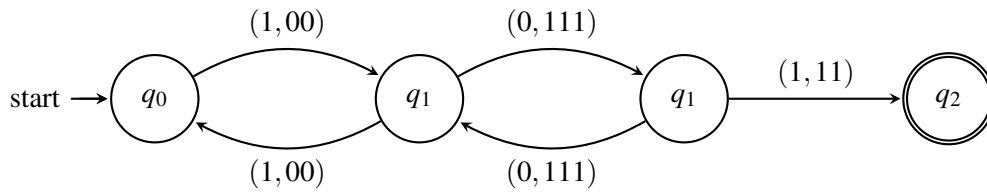


Figura 10.16: Representação da máquina de Mealy do Exemplo 10.32.

■ **Exemplo 10.33** Não é difícil verificar que a máquina de Mealy representado pela Figura 10.17 traduz a linguagem $\{a, b\}^*$ para a linguagem $\{101, 11\}^*$.

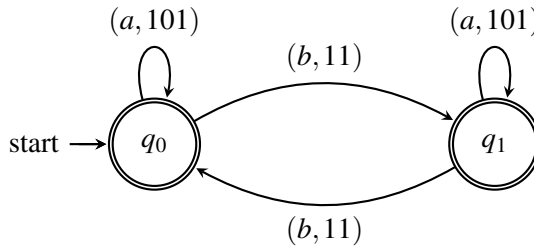


Figura 10.17: Representação da máquina de Mealy do Exemplo 10.33.

Observação 10.6 Um leitor mais atento pode notar que um AFD A pode ser encarado como sendo uma máquina de Mealy cuja função de tradução está desabilitada, ou seja, e neste caso a linguagem traduzida corresponderá exatamente ao conjunto vazio, isto é, $TRAD(A) = \emptyset$.

O resultado a seguir garante que máquinas de Mealy traduzem linguagens regulares em linguagens regulares, tal resultado diz que uma máquina de Mealy é um computador equipado com um compilador que compila a linguagem regular $\mathcal{L}(A)$ na linguagem regular $TRAD(A)$, em uma visão mais abstrata e algébrica uma máquina de Mealy computa um homomorfismo entre duas linguagens regulares.

Teorema 10.10 Se A é uma máquina de Mealy, então $TRAD(A)$ é uma linguagem regular.

Demonstração. Suponha que $A = \langle Q, \Sigma, \Lambda, \delta, \psi, q_0, F \rangle$ é uma máquina de Mealy, agora defina um AFD

$A' = \langle Q \cup \{q_z\}, \Lambda, \delta', q_0, F \rangle$ onde $q_z \notin Q$ e a função de transição δ' é definida pelas regras:

$$\delta'(q, a) = \begin{cases} \delta(q, a), & \text{se } q \in Q \text{ e } \psi(q, a) = a \\ q_z, & \text{senão} \end{cases}$$

claramente δ' está bem definida e é total. Além disso, por indução não é difícil verificar¹¹ que $\widehat{\delta'}(q_0, w) \in F$ se, e somente se, $\widehat{\delta}(q_0, u) \in F$ e $\widehat{\psi}(q_0, u) = w$ com $w \in \Lambda^*$, $u \in \Sigma^*$ e $q_f \in F$. Assim claramente $\mathcal{L}(A') = \text{TRAD}(A)$ e, portanto, $\text{TRAD}(A)$ é uma linguagem regular. \square

Em seu seminal artigo Mealy [73], propõe o uso de seu modelo¹² de autômato com saída como uma formalização para descrever matematicamente os circuitos sequências síncronos. O Exemplo 10.34 mostra como usar máquinas de Mealy para resolver problemas e a partir de tais máquinas produzir um circuito para resolver o problema.

■ **Exemplo 10.34** Considere o problema de detectar se uma palavra sobre o alfabeto $\{0, 1\}$ termina com uma sequência da forma 1010. Esse problema é solucionado pela máquina de Mealy A representada pela Figura , sobre está máquina é claro que $\widehat{\psi}(q_0, w) = v0001$, se e somente se, tem-se que $w = u1010$ para algum $v, u \in \{0, 1\}^*$, ou seja, o último caractere da palavra de saída será 1 apenas no caso da palavra de entrada terminar em 1010.

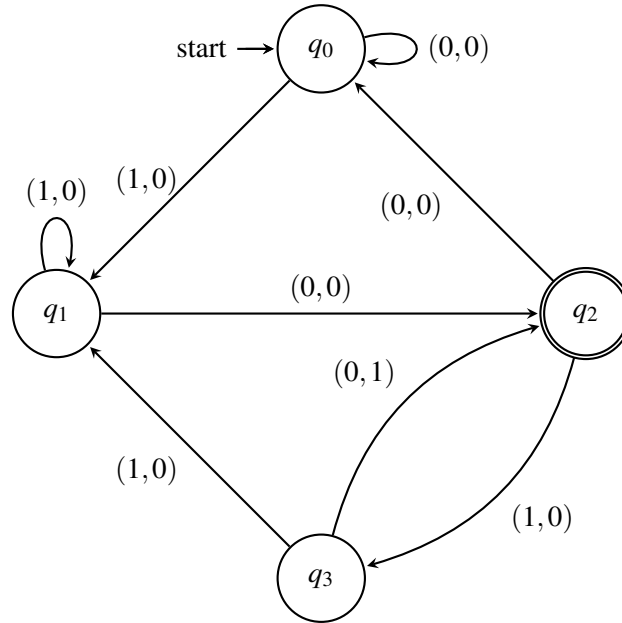


Figura 10.18: Representação da máquina de Mealy do Exemplo 10.34.

Agora para construir uma implementação física de tal máquina¹³ primeiro deve-se usar variáveis de estado para poder codificar os estados, aqui será usado as variáveis A e B com a seguinte codificação:

¹¹Este é um bom exercício ao leitor.

¹²Outro modelo que é equivalente as máquinas de Mealy nesse sentido são as máquinas de Moore [77].

¹³Um implementação física aqui diz respeito a projeto e construir o circuito que executa tal máquina.

Estado	Variáveis	
	A	B
q_0	0	0
q_1	0	1
q_2	1	0
q_3	1	1

Tabela 10.6: Codificação da máquina de Mealy da Figura 10.18.

Para a implementação física de tal máquina de Mealy será usado dois *Flip-Flops*¹⁴ do tipo D, para exibir o circuito antes serão codificadas todas as informações da máquina usando a Tabela 10.6 com referência, essa codificação pode ser vista na Tabela 10.7, ressaltando que será usado os símbolos A e B para as variáveis de estado e o símbolo Y para a saída.

Estado	Próximo Estado		Saídas		Flip-flop			
	X = 0	X = 1	X = 0	X = 1	X = 0	X = 1	X = 0	X = 1
AB	AB	AB	Y	Y	D_A	D_B	D_A	D_B
00	00	01	0	0	0	0	0	1
01	10	01	0	0	1	0	0	1
11	10	01	1	0	1	0	0	1
10	00	11	0	0	0	0	1	1

Tabela 10.7: Tabela de transição.

Da tabela anterior são montados os mapas de Karnaugh¹⁵ [48, 50, 65] a seguir.

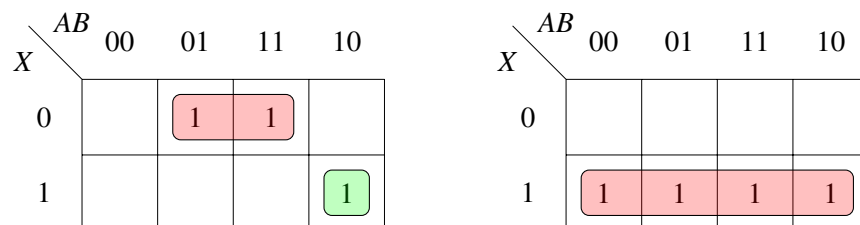
(a) Flip-flop D_A .(b) Flip-flop D_B .

Figura 10.19: Mapas de Karnaugh para o Circuito da Figura 10.7.

A partir dos mapas de Karnaugh das Figuras 10.19a e 10.19b e da Tabela 10.7 são obtidas as seguintes equações de excitação:

$$D_A = B\bar{X} + A\bar{B}X \quad (10.35)$$

e

$$D_B = X \quad (10.36)$$

¹⁴Em eletrônica e circuitos digitais, um *flip-flop*, é um circuito digital que pulsado pode ser usado como uma memória de um bit.

¹⁵Para leitores sem familiaridade com eletrônica digital é recomendado a leitura das obras [48, 50, 65].

e a equação do valor de saída:

$$Y = AB\bar{X} \quad (10.37)$$

Portanto, no circuito que implementa tal máquina tem-se que o flip-flop D_A recebe como entrada o “pulso¹⁶” produzido por uma porta lógica **OR** que combina a saída de duas portas lógicas **AND**, sendo que a primeira combina os valores de B e \bar{X} e a segunda combina os valores A, \bar{B} e X , ou seja, a segunda é uma porta para três entradas. Por sua vez, o flip-flop D_B recebe com entrada o valor de pulso de X . Vale salientar que os valores complementares dos pulsos são obtidos negando o sinal na entrada das portas, assim não são necessário esboçar explicitamente as portas **NOT** no diagrama do circuito. Por fim, o circuito também deverá conter uma porta **AND** de três entradas que combina A, B e \bar{X} para gerar a saída do circuito (Equação 10.37), o diagrama de tal circuito pode ser visto na Figura

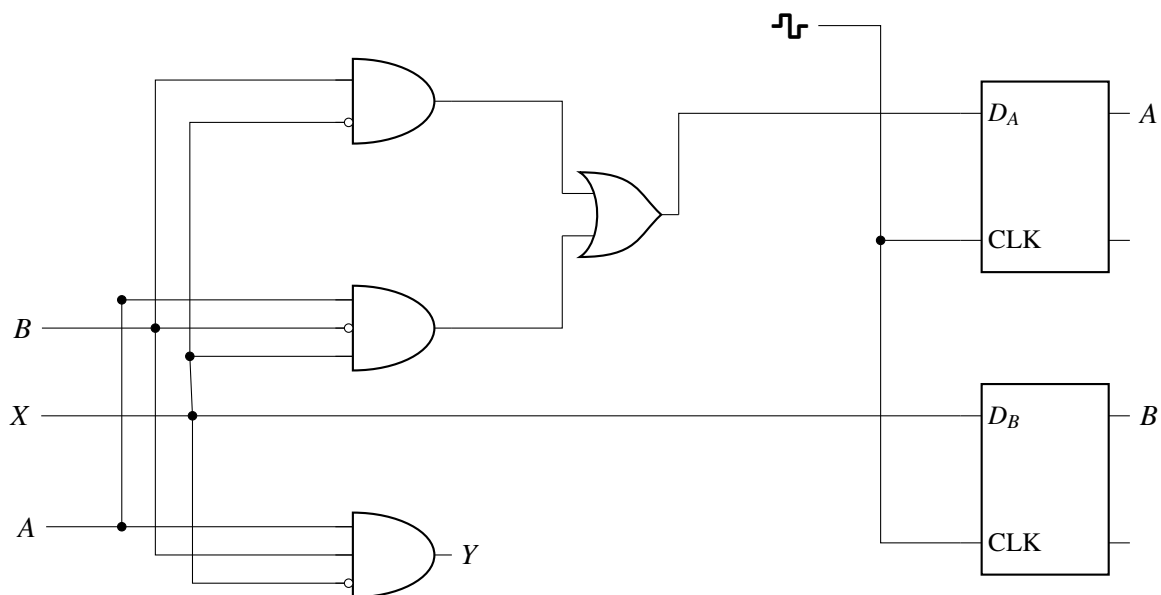


Figura 10.20: Diagrama do circuito que implementa a máquina de Mealy da Figura 10.18.

10.6 A Notação Matricial

Escrever depois...

Observação 10.7 Para esse questionário sempre que $w \in \Sigma^*$ e $c \in \Sigma$ a notação $|w|_c$ irá representar o número de c 's que existem na palavra w .

10.7 Questionário

■ **Exercício 10.1** Considere o autômato na Figura 10.21 e responda o que é solicitado.

¹⁶O termo pulso aqui é usado como sinônimo para a ideia de nível de sinal da corrente elétrica.

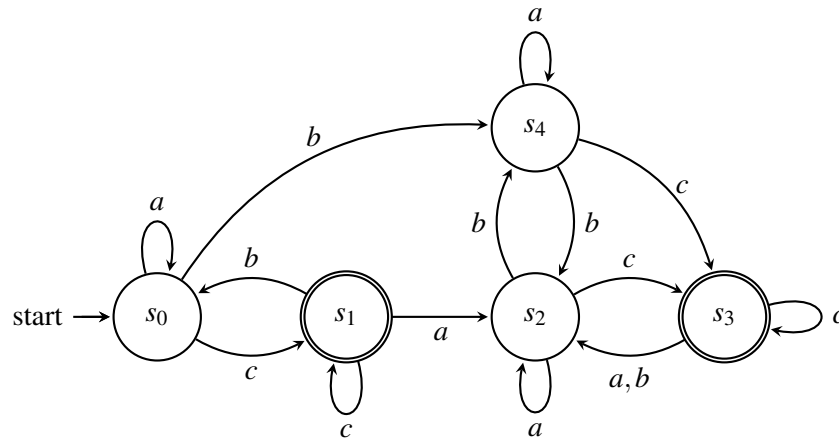


Figura 10.21: Autômato para o exercício 1.

- (a). Verifique se as palavras: $abccaaabacab$, $ccccbacabacbb$, $bbacabb$, $aaccca$, $aaabb$, $acacabb$, $bbacac$ e $ccabbbacac$ são aceitas ou não pelo autômato.
- (b). λ é aceito por este autômato?
- (c). Traduza o grafo de transição para a notação algébrica.

■ **Exercício 10.2** Construa um AFD que compute cada linguagem a seguir.

- (a). $L_1 = \{\lambda, 00101111, 11001101, 1\}$.
- (b). $L_2 = \{w10 \in \{0,1\}^* \mid w = (001)^n 11 \text{ com } n \in \mathbb{N}\}$.
- (c). $L_3 = \{w \in \{a,b\}^* \mid w = a^{2m}b^{2n+1} \text{ ou } w = aab^{3m+3}b^n \text{ com } m,n \in \mathbb{N}\}$.
- (d). $L_4 = \{w \in \{a,b,c,d\}^* \mid w \text{ não contém as subcadeias } ab \text{ e } cd\}$.
- (e). $L_5 = \{w \in \{0,1\}^* \mid (\forall n \in \mathbb{N})[|w|_1 \neq 3n]\}$.
- (f). $L_6 = \{w \in \{0,1\}^* \mid w = x_1 \cdots x_n \text{ e } x_i = 0 \text{ se } i \text{ for par, senão } x_i = 1, \text{ sendo } n \in \mathbb{N}\}$.
- (g). $L_7 = \{w \in \{x,y,z\}^* \mid \text{Se } w \text{ contém a sub-palavra } zz, \text{ então à direita da sub-palavra } zz \text{ não ocorre a sub-palavra } y\}$.
- (h). $L_8 = \{aaa, aab, aba, abb, baa, bab, bba, bbb\} \cup \{bbca, ccab, ccab, baba\}$.
- (i). $L_9 = \{uv \in \{0,1\}^* \mid u = 1^m 0111, v = 0100^p 1 \text{ com } m,p \in \mathbb{N}\}$.
- (j). $L_{10} = \{w \in \{0,1\}^* \mid w \text{ é um número binário múltiplo de } 3\}$.

■ **Exercício 10.3** Considerando o alfabeto $\Sigma = \{c,d\}$ para cada uma das linguagens definidas pelas propriedades a seguir construa um AFD que a compute.

- (a). w possui exatamente um único símbolo d , e este não aparece no final das palavras.

- (b). w tem apenas uma única sub-palavra dd .
- (c). w não contém três d 's seguidos.
- (d). w possui pelo menos um c .
- (e). w possui 6 ou menos símbolos, e não existe as sub-palavras cc e dd .
- (f). w tem mais que 4 símbolos c .
- (g). w é da forma db^4wb^5d com $w \in \Sigma^*$.
- (h). w possui tamanho n tal que $n \bmod 3 \neq 0$.
- (i). $|w| = n$ com $n \geq 3$ e $|w|_c \bmod 2 > 1$.
- (j). w é qualquer palavra tal que $3 \leq |w|_c \leq 6$.
- (k). w tem a forma $c^m d^n$ tal que m ou n não divisível por 2.
- (l). w tem a forma $c^m d^n c^p$ tal que $mnp > 6$ e mnp seja ímpar.
- (m). w é tal que $|w|_d = 2$ ou $|w|_c = 1$.
- (n). w tem a forma $c^m d^n$ tal que $m + n > mn$.
- (o). w possui a sub-palavra $ddcd$ e $|w|_d + |w|_c > 6$.

■ **Exercício 10.4** Considerando o alfabeto $\Sigma = \{2, 3, 5\}$, demonstre usando AFD que as linguagens definidas pelas propriedades a seguir são regulares.

- (a). $L_{pp} = \{w \in \Sigma^* \mid |w|_2, |w|_5 \text{ são ambos pares}\}$.
- (b). $L_{pi} = \{w \in \Sigma^* \mid |w|_2 \cdot |w|_3 \text{ é par e } |w|_5 \text{ é ímpar}\}$.
- (c). $L_{pip} = \{w \in \Sigma^* \mid |w|_2 \text{ é par, } |w|_3 \text{ é ímpar e } |w|_5 \text{ é par}\}$.
- (d). $L_{pu} = \{w \in \Sigma^* \mid w = x_1 \cdots x_n, x_1 = x_n \text{ com } n \in \mathbb{N}\}$.
- (e). $L_{pud} = \{w \in \Sigma^* \mid w = x_1 \cdots x_n, x_1 \neq x_n \text{ com } n \in \mathbb{N}\}$.
- (f). $L_{dif} = \{wuv \in \Sigma^* \mid w, v \in \{3, 5\}^*, |u| \geq 3\}$.
- (g). $L_{inv} = \{wuv \in \Sigma^* \mid w \in \{2\}^*, v \in \{3, 5\}^+, u \in \{3\}^*\}$.

■ **Exercício 10.5** Dado um AFD $A = \langle Q, \Sigma, \delta, q_0, F \rangle$ qualquer, mostre que para todo $u, v \in \Sigma^*$ tem-se que $\widehat{\delta}(q_0, uv) = \widehat{\delta}(\widehat{\delta}(q_0, u), v)$.

- **Exercício 10.6** Considerando os AFN das Figuras 10.5, 10.7 e 10.8 execute as computações das palavras *aabbababa*, *bbababba*, *bbbbabaabb*, *bababaaa* e *ababababb*.
- **Exercício 10.7** Considerando o AFN da Figura 10.9 construa as árvores de computação para as palavras 010101 e 11001101.
- **Exercício 10.8** Converta os AFN das Figuras 10.5, 10.7, 10.8 e 10.9 para a notação algébrica.
- **Exercício 10.9** Encontre os AFD equivalentes aos AFN das Figuras 10.5, 10.7 e 10.8.
- **Exercício 10.10** Seja $L = \{01, 012\}$, construa um AFN com 4 estados (ou menos) que aceite a linguagem L^* .
- **Exercício 10.11** Dado a linguagem $L = \{0101^m \mid m \geq 1\} \cup \{010^n \mid n \in \mathbb{N}\}$ construa um AFN com 5 ou menos estados que aceite a linguagem L .
- **Exercício 10.12** Dado os AFN que você construiu nos Exercícios 10 e 11, encontre os AFD equivalentes a eles.
- **Exercício 10.13** Considerando o AFN representado pelo grafo de transição na Figura 10.22 responda o que é solicitado.

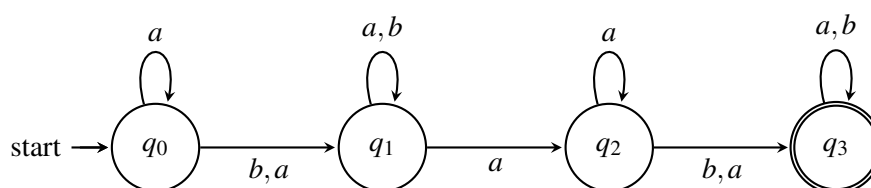


Figura 10.22: Autômato para o exercício 12.

- (a). Realize a computação para as palavras *aaba* e *baba*.
 - (b). Esboce a árvore de computação para a palavra *abbaabaab*.
 - (c). Converta o AFN da Figura 10.22 para a notação algébrica.
 - (d). Encontre um AFD equivalente ao AFN da Figura 10.22.
- **Exercício 10.14** Considerando o AFN representado pelo grafo de transição na Figura 10.23 responda o que é solicitado.
- (a). Realize a computação para as palavras 01110 e 1001011.
 - (b). Esboce a árvore de computação para a palavra 110110101.
 - (c). Converta o AFN da Figura 10.23 para a notação algébrica.

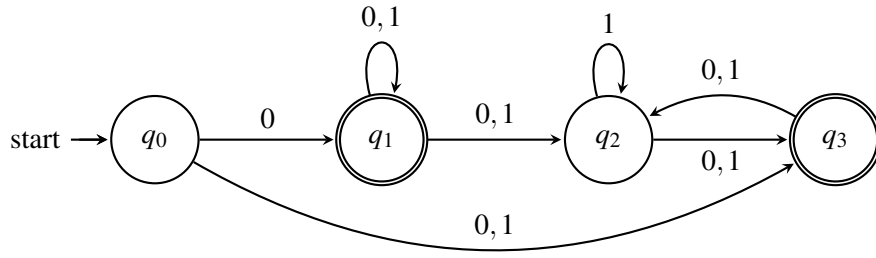


Figura 10.23: Autômato para o exercício 14.

(d). Encontre um AFD equivalente ao AFN da Figura 10.23.

■ **Exercício 10.15** Considerando o λ -AFN do Exemplo 10.25 responda o que é solicitado.

(a). Calcule $\delta_\lambda(q_0)$, $\delta_\lambda(q_1)$ e $\delta_\lambda(q_2)$.

(b). Compute $\widehat{\delta_N}(q_0, 11233)$.

(c). Apresente um AFD equivalente ao λ -AFN.

■ **Exercício 10.16** Considerando $\Sigma_1 = \{0, 1\}$ e $\Sigma_2 = \{2, 3\}$ construa um λ -AFN com um único estado final que aceita a linguagem $\{w \in \Sigma_1^* \mid |w|_1 = 2i, i \in \mathbb{N}\} \cup \{w \in \Sigma_2^* \mid |w|_3 = 2j + 1, j \in \mathbb{N}\}$.

■ **Exercício 10.17** Considere o alfabeto $\Sigma = \{a, b, c\}$ e as linguagens $L_1 = \{w \in \Sigma^* \mid |w|_a \geq 0, |w|_b > 0, |w|_c = 2k, k \in \mathbb{N}\}$ e $L_2 = \{abaccc, ababc, abacab, acbcc, bacbaaa, abb, bba, aaaabbbb, cac, ccba, caccabcac\}$. Para cada uma das linguagens $L_1 \cup L_2$, L_1L_2 e L_2L_1 construa um λ -AFN A que as aceite.

■ **Exercício 10.18** Considerando o λ -AFN $A = \langle Q, \Sigma, \underline{\delta_N}, q_0, \{q_2\} \rangle$ onde $\underline{\delta_N}$ é especificada pela Tabela 10.8, responda o que é solicitado.

$Q' \backslash \Sigma \cup \{\lambda\}$	λ	a	b	c
q_0	\emptyset	$\{q_0\}$	$\{q_1\}$	$\{q_2\}$
q_1	$\{q_0\}$	$\{q_1\}$	$\{q_2\}$	\emptyset
q_2	$\{q_1\}$	$\{q_2\}$	\emptyset	$\{q_0\}$

Tabela 10.8: Tabela da função de transição do λ -AFN do Exercício 18.

(a). Para cada estado do autômato calcule δ_λ .

(b). Esboce todos os w pertencentes a linguagem do autômato tal que $|w| \leq 4$.

(c). Encontre um AFD equivalente A .

■ **Exercício 10.19** Considerando o λ -AFN $S = \langle Q, \Sigma, \underline{\delta_N}, s_0, \{s_2\} \rangle$ onde $\underline{\delta_N}$ é especificada pela Tabela 10.9, responda o que é solicitado.

$Q' \backslash \Sigma \cup \{\lambda\}$	λ	0	1	2
s_0	$\{s_1, s_2\}$	\emptyset	$\{s_1\}$	$\{s_2\}$
s_1	\emptyset	$\{s_0\}$	$\{s_2\}$	$\{s_0, s_2\}$
s_2	\emptyset	\emptyset	\emptyset	\emptyset

Tabela 10.9: Tabela da função de transição do λ -AFN do exercício 19.

- (a). Para cada estado do autômato calcule δ_λ .
- (b). Esboce todos os w pertencentes a linguagem do autômato tal que $|w| \leq 3$.
- (c). Encontre um AFD equivalente S .

■ **Exercício 10.20** Para as linguagens especificadas pelos enunciados a seguir construa λ -AFN que as aceite.

- (a). A linguagem de todas as palavras que começam e terminam com qualquer letra do alfabeto $\{a, b, c\}$, porém não existe nas palavras dois a 's e dois c 's seguidos.
- (b). A linguagem de todas as palavras sobre o alfabeto $\{0, 1\}$ que iniciam com o prefixo $(01)^n$ e termina com o sufixo 10 ou começam com o prefixo $(101)^n$ e terminam com o sufixo 00, sendo $n \in \mathbb{N}$ tal que $n \geq 1$.
- (c). A linguagem de todas as palavras sobre o alfabeto $\{x, y\}$ tal que no mínimo uma das quatro últimas letras da palavra é um x .
- (d). A linguagem de todas as palavras sobre o alfabeto $\{x, y, z\}$ tal que no máximo uma das três primeiras letras da palavra é um y ou as duas primeiras são z .

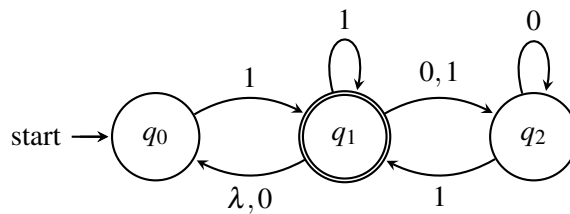


Figura 10.24: Autômato para o Exercício 10.26.

- **Exercício 10.21** Encontre o AFD quociente do AFD apresentado no Exercício 10.1.
- **Exercício 10.22** Encontre o AFD quociente equivalente ao AFN apresentado no Exercício 10.13.
- **Exercício 10.23** Encontre o AFD quociente equivalente ao AFN apresentado no Exercício 10.14.
- **Exercício 10.24** Encontre o AFD quociente equivalente ao λ -AFN apresentado no Exercício 10.18.

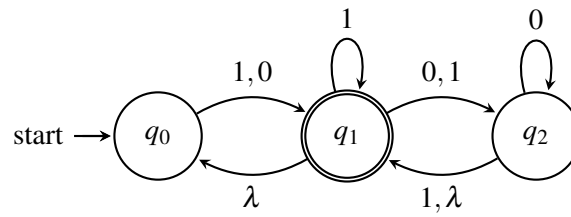


Figura 10.25: Autômato para o Exercício 10.27.

- **Exercício 10.25** Encontre o AFD quociente equivalente ao λ -AFN apresentado no Exercício 10.19.
- **Exercício 10.26** Encontre o AFD quociente equivalente ao λ -AFN da Figura 10.24 .
- **Exercício 10.27** Encontre o AFD quociente equivalente ao λ -AFN da Figura 10.25.

Capítulo 11

Expressões e Gramáticas Regulares

Uma teoria matemática não deve ser considerada completa até que a tenhamos feito tão clara que a possamos explicar ao primeiro homem que encontramos na rua.

David Hilbert

No Capítulo 10 a noção de linguagens regulares foi formalizado utilizado pela das máquinas de computação, isto é, sobre a perspectiva dos autômatos finitos, em tal perspectiva as linguagens são vistas como sendo conjuntos de palavras sobre os quais as máquinas tinha a tarefa de reconhecer seus elementos, ou seja, decidir quando um dada palavra pertencia ou não a linguagem, agora neste capítulo serão apresentada as linguagens regulares através da ótica de formalizamos geradores.

11.1 Expressões Regulares

Neste seção será apresentada uma nova visão de aspecto mais algébrico para as linguagens regulares, essa nova visão foi introduzida por Kleene em seu seminal *paper* “*Representation of events in nerve nets and finite automata*” [54], tal perspectiva consiste em um sistema formal (com sintaxe e semântica) chamado de expressões regulares, a seguir é formalizado a sintaxe das expressões regulares.

Definição 11.1 — Conjunto das Expressões Regulares – Sintaxe. Seja Σ uma alfabeto, o conjunto de todas as expressões regulares sobre Σ , denotado por Exp_{Σ} , é o conjunto indutivamente gerado pelas seguintes regras.

(B)ase: \emptyset, λ e cada $a \in \Sigma$, são expressões regulares^a.

(P)asso indutivo: Se $r_1, r_2 \in Exp_{\Sigma}$, então $r_1 + r_2, r_1 \cdot r_2, r_1^*, (r_1) \in Exp_{\Sigma}$.

(F)echo: Exp_{Σ} é exatamente o conjunto dos elementos obtidos a partir **(B)** ou usando-se uma quantidade finita (podendo ser nula) de aplicações de **(P)**.

^aAs expressões regulares da base costumam ser chamadas de expressões regulares primitivas.

No que diz respeito as expressões regulares é comum assumir que $+, \cdot, (,), * \notin \Sigma$, assim uma expressão regular nem sempre é uma palavra sobre Σ , em geral os símbolos $+$ e \cdot são lidos como soma e produto [22]. Além disso, como dito em [15] se $r_1, r_2 \in Exp_{\Sigma}$, então costuma-se escrever $r_1 r_2$ em vez de $r_1 \cdot r_2$.

Observação 11.1 Também é possível encontrar referência em que o termo expressão regular seja trocado para álgebra de Kleene.

■ **Exemplo 11.1** Considerando o alfabeto $\{0, 1\}$ tem-se que $(1 + 1)0, 01 \in Exp_{\Sigma}$. Essa afirmação pode ser verificada construindo tais palavras facilmente, basta pela regra **(B)** tem-se que 0 e 1 são expressões regulares primitivas, assim usando a regra **(P)** pode-se construir as expressões $1 + 1$ e 01 , agora aplicando novamente a regra **(P)** sobre a expressão $1 + 1$ pode-se gerar a expressão $(1 + 1)$, finalmente, aplicando **(P)** novamente obtem-se a expressão $(1 + 1)0$ e isso mostra que de fato $(1 + 1)0, 01 \in Exp_{\Sigma}$.

■ **Exemplo 11.2** Considerando o alfabeto $\{a, b, c\}$ tem-se que $a(\emptyset + (bc^*)) \in Exp_{\Sigma}$. Essa afirmação pode ser verificada construindo tal palavra, uma vez que, a, b e c são expressões regulares primitivas aplicando o passo **(B)** varias vezes será obtida a expressão regular $a(\emptyset + (bc^*))$.

A seguir será formalizado o conceito de semântica para as expressões regulares, sendo que tal semântica pode ser visto como uma **semântica denotacional**¹ [97], que apresenta significado as operações de soma e multiplicação.

Definição 11.2 — Semântica das Expressão Regulares. Seja Exp_{Σ} o conjunto das expressões regulares sobre Σ , a semântica (ou interpretação) de Exp_{Σ} é uma função $\mathcal{L} : Exp_{\Sigma} \rightarrow \wp(\Sigma^*)$ definida recursivamente para todo $r, r_1, r_2 \in Exp_{\Sigma}$ pelas seguintes regras.


- (i) Se $r \in \Sigma \cup \{\lambda\}$, então $\mathcal{L}(r) = \{r\}$.
- (ii) Se $r = \emptyset$, então $\mathcal{L}(r) = \emptyset$.
- (iii) Se $r = r_1 + r_2$, então $\mathcal{L}(r) = \mathcal{L}(r_1) \cup \mathcal{L}(r_2)$.
- (iv) Se $r = r_1 \cdot r_2$, então $\mathcal{L}(r) = \mathcal{L}(r_1) \cdot \mathcal{L}(r_2)$.
- (v) Se $r = r_1^*$, então $\mathcal{L}(r) = (\mathcal{L}(r_1))^*$.
- (vi) Se $r = (r_1)$, então $\mathcal{L}(r) = (\mathcal{L}(r_1))$.

Agora como explicado em [22], para a valoração de expressões regulares não primitivas, é necessário

¹Uma semântica denotacional é aquela em que as funções de valoração usadas, são funções que mapeiam palavras da linguagem para funções parciais que representam o comportamento dos programas.

que seja seguido a precedência dos operadores no momento de combinar as linguagens, sendo tal precedência no sentido de maior precedência para a menor formada pela seguinte ordem: fecho de Kleene, concatenação e união.

Observação 11.2 Vale destacar que como na aritmética convencional os parênteses mudam a precedência dos conectivos anteriores, e deve ser avaliados dos mais internos para os mais externos.

 **Nota 11.1** A valoração pode vir a gerar situações como (L) onde $L \subseteq \Sigma^*$, neste caso será escrito simplesmente L vez de (L) .

■ **Exemplo 11.3** Dado o alfabeto $\{0, 1\}$ e $(0 + 1^*)0 \in \text{Exp}_\Sigma$ tem-se que:

$$\begin{aligned}
 \mathcal{L}((0 + 1^*)0) &= \mathcal{L}((0 + 1^*))\mathcal{L}(0) \\
 &= (\mathcal{L}(0 + 1^*))\{0\} \\
 &= (\mathcal{L}(0) \cup \mathcal{L}(1^*))\{0\} \\
 &= (\mathcal{L}(0) \cup (\mathcal{L}(1))^*)\{0\} \\
 &= (\{0\} \cup \{1\}^*)\{0\} \\
 &= (\{0\} \cup \{\lambda, 1, 11, 111, 1111, \dots\})\{0\} \\
 &= \{\lambda, 0, 1, 11, 111, 1111, \dots\}\{0\} \\
 &= \{0, 00, 10, 110, 1110, 11110, \dots\}
 \end{aligned}$$

ou seja, a valoração da expressão regular $\{0, 1\}$ e $(0 + 1^*)0$ é exatamente a linguagem de todas as palavras w sobre o alfabeto $\{0, 1\}$ sendo que $w = 0^m$ ou $w = 1^n0$ com $m, n \in \mathbb{N}$ tal que $1 \leq m \leq 2$ e $n \geq 1$.

■ **Exemplo 11.4** Dado o alfabeto $\{0, 1\}$ e $(00)^* \in \text{Exp}_\Sigma$ tem-se que:

$$\begin{aligned}
 \mathcal{L}((00)^*) &= (\mathcal{L}((00)))^* \\
 &= ((\mathcal{L}(00)))^* \\
 &= ((\mathcal{L}(0)\mathcal{L}(0)))^* \\
 &= ((\{0\}\{0\}))^* \\
 &= ((\{00\}))^* \\
 &= \{00\}^* \\
 &= \{\lambda, 00, 0000, 000000, 00000000, \dots\}
 \end{aligned}$$

ou seja, a valoração da expressão regular $(00)^*$ consiste da linguagem de todas as palavras w sobre o alfabeto $\{0, 1\}$ sem nenhum 1 e que o tamanho seja par, isto é, $|w| = 2k$ para algum $k \in \mathbb{N}$.

■ **Exemplo 11.5** Dado o alfabeto $\{a, b, c\}$ e a expressão $((ab)^* + c)\emptyset \in \text{Exp}_\Sigma$ tem-se que:

$$\begin{aligned}
 \mathcal{L}(((ab)^* + c)\emptyset) &= \mathcal{L}(((ab)^* + c))\mathcal{L}(\emptyset) \\
 &= ((\mathcal{L}((ab)))^* \cup \{c\})\emptyset \\
 &= (((\mathcal{L}(ab)))^* \cup \{c\})\emptyset \\
 &= (((\mathcal{L}(a)\mathcal{L}(b)))^* \cup \{c\})\emptyset \\
 &= (((\{a\}\{b\}))^* \cup \{c\})\emptyset \\
 &= (((\{ab\}))^* \cup \{c\})\emptyset \\
 &= ((\{ab\})^* \cup \{c\})\emptyset \\
 &= (\{ab\}^* \cup \{c\})\emptyset \\
 &= (\{\lambda, ab, abab, ababab, \dots\} \cup \{c\})\emptyset \\
 &= (\{\lambda, c, ab, abab, ababab, \dots\})\emptyset \\
 &= \{\lambda, c, ab, abab, ababab, \dots\}\emptyset \\
 &= \{\lambda, c, ab, abab, ababab, \dots\}
 \end{aligned}$$

portanto, a valoração da expressão regular $((ab)^* + c)\emptyset$ é exatamente a linguagem de todas as palavras w sobre o alfabeto $\{a, b, c\}$ onde $w = c^m$ ou $w = (ab)^n$ com $m, n \in \mathbb{N}$ tal que $m \leq 1$ e $n \geq 1$.

Definição 11.3 Duas expressões regulares $r_1, r_2 \in \text{Exp}_\Sigma$ são dita equivalentes, denotado por $r_1 \equiv r_2$, sempre que $\mathcal{L}(r_1) = \mathcal{L}(r_2)$.

■ **Exemplo 11.6** Dada a expressão $00 + (1^*0)$ note que:

$$\begin{aligned}
 \mathcal{L}(00 + (1^*0)) &= \mathcal{L}(00) \cup \mathcal{L}((1^*0)) \\
 &= \mathcal{L}(0)\mathcal{L}(0) \cup ((\mathcal{L}(1))^*\mathcal{L}(0)) \\
 &= \{0\}\{0\} \cup ((\{1\})^*\{0\}) \\
 &= \{0\}\{0\} \cup (\{1\}^*\{0\}) \\
 &= \{0\}\{0\} \cup (\{\lambda, 1, 11, 111, 1111, \dots\}\{0\}) \\
 &= \{00\} \cup \{0, 10, 110, 1110, 11110, \dots\} \\
 &= \{0, 00, 10, 110, 1110, 11110, \dots\}
 \end{aligned}$$

logo, tal expressão é equivalente a expressão $(0 + 1^*)0$ apresentada no Exemplo 11.3.

Proposição 11.1 Se $r_1, r_2, r_3 \in \text{Exp}_\Sigma$, então $r_1(r_2 + r_3) \equiv r_1r_2 + r_1r_3$.

Demonstração. Assuma que $r_1, r_2, r_3 \in \text{Exp}_\Sigma$, logo tem-se que;

$$\begin{aligned}
 \mathcal{L}(r_1(r_2 + r_3)) &= \mathcal{L}(r_1)\mathcal{L}((r_2 + r_3)) \\
 &= \mathcal{L}(r_1)(\mathcal{L}(r_2 + r_3)) \\
 &= \mathcal{L}(r_1)(\mathcal{L}(r_2) \cup \mathcal{L}(r_3)) \\
 &= \mathcal{L}(r_1)(\mathcal{L}(r_2) \cup \mathcal{L}(r_3)) \\
 &= \mathcal{L}(r_1)\mathcal{L}(r_2) \cup \mathcal{L}(r_1)\mathcal{L}(r_3) \\
 &= \mathcal{L}(r_1r_2 + r_1r_3)
 \end{aligned}$$

O que conclui a prova. □

Agora será mostrado qual a classe (ou tipo) de linguagens definidas por expressões regulares, ou seja, agora será mostrado a que classe pertencem as linguagens fruto da valoração das expressões regulares.

Teorema 11.1 — Transformação de expressões regulares para AFN. Se $L = \mathcal{L}(r)$ para alguma expressão regular r , então existe um λ -AFN A tal que $L = \mathcal{L}(A)$.

Demonstração. Suponha que $L = \mathcal{L}(r)$ para alguma expressão regular r , agora por indução sobre o número de operadores em r será mostrado que existe um λ -AFN A tal que $L = \mathcal{L}(A)$.

- **Base da indução:** Para as expressões regulares r com 0 operadores, isto é, $r = \lambda$ ou $r = \emptyset$ ou $r = a$ para $a \in \Sigma$ considere os seguintes λ -AFN.

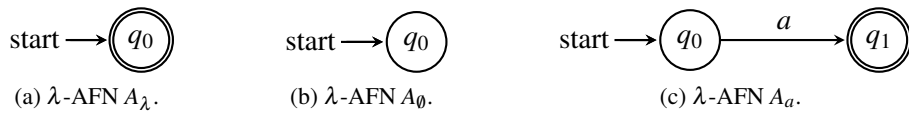


Figura 11.1: Os três λ -AFN básicos para as expressões regulares primitivas.

Claramente tem-se que $\mathcal{L}(\lambda) = \mathcal{L}(A_\lambda)$, $\mathcal{L}(\emptyset) = \mathcal{L}(A_\emptyset)$ e $\mathcal{L}(a) = \mathcal{L}(A_a)$ para todo $a \in \Sigma$.

- **Hipótese indutiva (HI):** Suponha que para toda $r \in \text{Exp}_\Sigma$ com n operadores tal que $n \geq 0$ existe um λ -AFN da forma $A = \langle Q_r, \Sigma, \delta_N^r, q_0^r, \{q_f^r\} \rangle$ tal que $\mathcal{L}(r) = \mathcal{L}(A^r)$. Para fins de representação considere que tal λ -AFN A^r tem a forma descrita pela Figura 11.2.

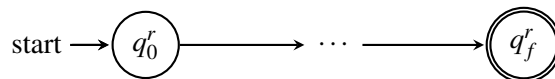


Figura 11.2: λ -AFN A^r para uma expressão r genérica com n operadores.

- **Passo indutivo:** Agora dado uma expressão regular r com $n + 1$ operadores tal que $n \geq 0$, existe três (e apenas três) casos possíveis para a forma de r .

1º Caso: $r = r_1 + r_2$, assim respectivamente r_1 e r_2 possuem k_1 e k_2 operadores tais que $k_1 + k_2 = n$, obviamente $k_1, k_2 \geq 0$ e, portanto, por **(HI)** tem-se que existem $A^{r_1} = \langle Q_{r_1}, \Sigma, \underline{\delta}_N^{r_1}, q_0^{r_1}, \{q_f^{r_1}\} \rangle$ e $A^{r_2} = \langle Q_{r_2}, \Sigma, \underline{\delta}_N^{r_2}, q_0^{r_2}, \{q_f^{r_2}\} \rangle$ tal que $\mathcal{L}(r_1) = \mathcal{L}(A^{r_1})$ e $\mathcal{L}(r_2) = \mathcal{L}(A^{r_2})$, agora pode-se criar um novo λ -AFN $A^r = \langle Q_{r_1} \cup Q_{r_2} \cup \{q_0\}, \Sigma, \underline{\delta}_N^r, q_0, \{q_f^{r_1}, q_f^{r_2}\} \rangle$ onde,

$$\underline{\delta}_N^r(q, a) = \begin{cases} \underline{\delta}_N^{r_1}(q, a), & \text{se } q \in Q_{r_1} \\ \underline{\delta}_N^{r_2}(q, a), & \text{se } q \in Q_{r_2} \\ \{q_0^{r_1}, q_0^{r_2}\}, & \text{se } q = q_0, a = \lambda \\ \emptyset, & \text{qualquer outro caso} \end{cases}$$

Para fins de representação considere que tal λ -AFN A^r tem a forma a seguir.

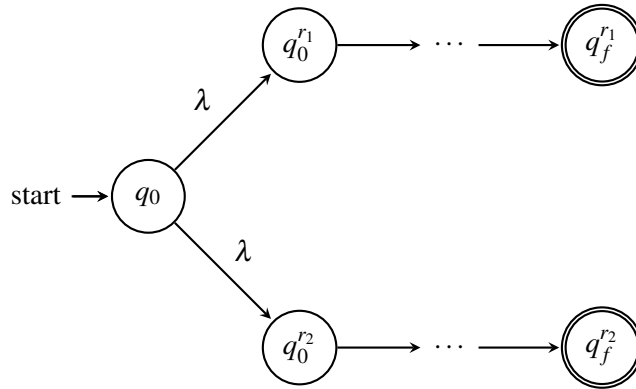


Figura 11.3: λ -AFN A^r para uma expressão $r_1 + r_2$.

Agora note que para todo $w \in \Sigma^*$ tem-se que,

$$\begin{aligned} w \in \mathcal{L}(r) &\iff w \in \mathcal{L}(r_1 + r_2) \\ &\iff w \in \mathcal{L}(r_1) \cup \mathcal{L}(r_2) \\ &\iff w \in \mathcal{L}(r_1) \text{ ou } w \in \mathcal{L}(r_2) \\ &\iff \widehat{\underline{\delta}_N^{r_1}}(q_0^{r_1}, w) \cap \{q_f^{r_1}\} \neq \emptyset \text{ ou } \widehat{\underline{\delta}_N^{r_2}}(q_0^{r_2}, w) \cap \{q_f^{r_2}\} \neq \emptyset \\ &\iff \widehat{\underline{\delta}_N}(q_0, w) \cap \{q_f^{r_1}\} \neq \emptyset \text{ ou } \widehat{\underline{\delta}_N}(q_0, w) \cap \{q_f^{r_2}\} \neq \emptyset \\ &\iff \widehat{\underline{\delta}_N}(q_0, w) \cap \{q_f^{r_1}, q_f^{r_2}\} \neq \emptyset \\ &\iff w \in \mathcal{L}(A^r) \end{aligned}$$

2º Caso: $r = r_1 r_2$, assim novamente r_1 e r_2 possuem k_1 e k_2 operadores tais que $k_1 + k_2 = n$, obviamente $k_1, k_2 \geq 0$ e, portanto, por **(HI)** tem-se que existem $A^{r_1} = \langle Q_{r_1}, \Sigma, \underline{\delta}_N^{r_1}, q_0^{r_1}, \{q_f^{r_1}\} \rangle$

e $A^{r_2} = \langle Q_{r_2}, \Sigma, \underline{\delta_N}^{r_2}, q_0^{r_2}, \{q_f^{r_2}\} \rangle$ tal que $\mathcal{L}(r_1) = \mathcal{L}(A^{r_1})$ e $\mathcal{L}(r_2) = \mathcal{L}(A^{r_2})$, agora pode-se criar um novo λ -AFN $A^r = \langle Q_{r_1} \cup Q_{r_2}, \Sigma, \underline{\delta_N}^r, q_0^{r_1}, \{q_f^{r_2}\} \rangle$ onde,

$$\underline{\delta_N}^r(q, a) = \begin{cases} \underline{\delta_N}^{r_1}(q, a), & \text{se } q \in Q_{r_1} - \{q_f^{r_1}\}, a \in \Sigma \cup \{\lambda\} \\ \underline{\delta_N}^{r_1}(q, a), & \text{se } q = q_f^{r_1}, a \in \Sigma \\ \underline{\delta_N}^{r_1}(q, a) \cup \{q_0^{r_1}\}, & \text{se } q = q_f^{r_1}, a = \lambda \\ \underline{\delta_N}^{r_2}(q, a), & \text{se } q \in Q_{r_2} - \{q_f^{r_2}\}, a \in \Sigma \cup \{\lambda\} \\ \underline{\delta_N}^{r_2}(q, a), & \text{se } q = q_f^{r_2}, a \in \Sigma \\ \underline{\delta_N}^{r_2}(q, a) \cup \{q_0^{r_2}\}, & \text{se } q = q_f^{r_2}, a = \lambda \\ \emptyset, & \text{qualquer outro caso} \end{cases}$$

E para fins de representação tal λ -AFN A^r possui a forma a seguir.

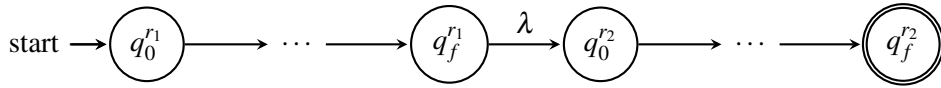


Figura 11.4: λ -AFN A^r para uma expressão $r_1 r_2$.

Agora note que para todo $w \in \Sigma^*$ tal que $w = xy$ com $x, y \in \Sigma^*$ tem-se que,

$$\begin{aligned} w \in \mathcal{L}(r) &\iff w \in \mathcal{L}(r_1 r_2) \\ &\iff xy \in \mathcal{L}(r_1 r_2) \\ &\iff xy \in \mathcal{L}(r_1) \mathcal{L}(r_2) \\ &\iff x \in \mathcal{L}(r_1) \text{ e } y \in \mathcal{L}(r_2) \\ &\iff \widehat{\underline{\delta_N}^{r_1}}(q_0^{r_1}, x) \cap \{q_f^{r_1}\} \neq \emptyset \text{ e } \widehat{\underline{\delta_N}^{r_2}}(q_0^{r_2}, y) \cap \{q_f^{r_2}\} \neq \emptyset \\ &\iff \widehat{\underline{\delta_N}}(q_0^{r_1}, x) \cap \{q_f^{r_1}\} \neq \emptyset \text{ e } \widehat{\underline{\delta_N}}(q_0^{r_2}, y) \cap \{q_f^{r_2}\} \neq \emptyset \\ &\iff \widehat{\underline{\delta_N}}(\widehat{\underline{\delta_N}}(q_0^{r_1}, x), y) \cap \{q_f^{r_2}\} \neq \emptyset \\ &\iff \widehat{\underline{\delta_N}}(q_0^{r_1}, xy) \cap \{q_f^{r_2}\} \neq \emptyset \\ &\iff w \in \mathcal{L}(A^r) \end{aligned}$$

3º Caso: $r = r_1^*$, onde r_1 tem exatamente n operadores sendo que $n \geq 0$, assim por **(HI)** tem-se que existem $A^{r_1} = \langle Q_{r_1}, \Sigma, \underline{\delta_N}^{r_1}, q_0^{r_1}, \{q_f^{r_1}\} \rangle$ tal que $\mathcal{L}(r_1) = \mathcal{L}(A^{r_1})$, agora pode-se criar um

novo λ -AFN $A^r = \langle Q_{r_1} \cup \{q_0, q_f\}, \Sigma, \delta_N^r, q_0^{r_1}, \{q_f\} \rangle$ onde,

$$\delta_N^r(q, a) = \begin{cases} \delta_N^{r_1}(q, a), & \text{se } q \in Q_{r_1} - \{q_f^{r_1}\} \\ \delta_N^{r_1}(q, a), & \text{se } q = q_f^{r_1}, a \in \Sigma \\ \delta_N^{r_1}(q, a) \cup \{q_f\}, & \text{se } q = q_f^{r_1}, a = \lambda \\ \{q_0^{r_1}, q_f\}, & \text{se } q = q_0, a = \lambda \\ \{q_0^{r_1}\}, & \text{se } q = q_f, a = \lambda \\ \emptyset, & \text{qualquer outro caso} \end{cases}$$

E para fins de representação tal λ -AFN A^r tem sua forma como descrita pela figura a seguir.

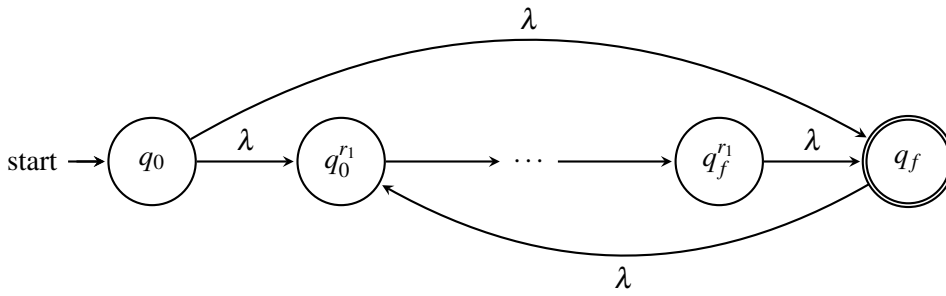


Figura 11.5: λ -AFN A^r para uma expressão r_1^* .

A prova de que $\mathcal{L}(r) = \mathcal{L}(A^r)$ ficará de exercício ao leitor. Agora os três casos anteriores permitem afirmar que sempre existe um λ -AFN A tal que $L = \mathcal{L}(A)$.

□

Uma consequência imediata deste teorema é apresentada a seguir.

Corolário 11.1 A linguagem (ou valoração) de qualquer $r \in \text{Exp}_\Sigma$ é uma linguagem regular.

Demonstração. Direto do Teorema 11.1 e o Corolário 10.2.

□

■ **Exemplo 11.7** Dado o alfabeto $\{a, b\}$ e $(ab)^* \in \text{Exp}_\Sigma$ pelo Teorema 11.1 são construídos os autômatos a seguir,

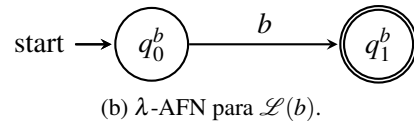
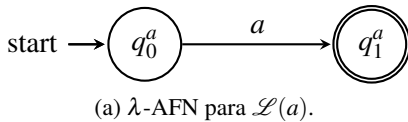
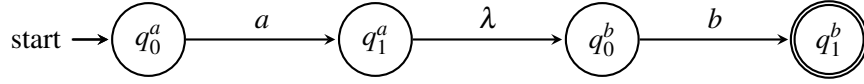
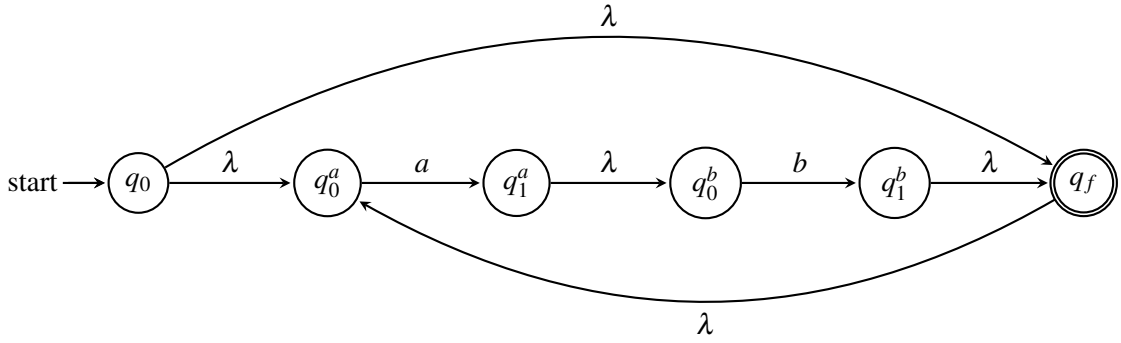


Figura 11.6: Os λ -AFN básicos para as expressões regulares primitivas a e b .

Pode-se então gerar o λ -AFN para reconhecer a concatenação ab , tal autômato será gerado pela combinação dos autômatos das Figuras 11.6a e 11.6b, gerando com resultado o autômato a seguir como se segue,

Figura 11.7: λ -AFN para as expressões regular $\mathcal{L}(ab)$.

Para finalizar é necessário agora apresentar o autômato que reconheça o fecho de Kleene da expressão ab , tal autômato é construído seguindo o Teorema 11.1, e tal autômato é representado pelo grafo de transição esboçado na Figura 11.8 a seguir.

Figura 11.8: λ -AFN para as expressões regular $\mathcal{L}((ab)^*)$.

Observação 11.3 Apenas na demonstração a seguir considere que o símbolo Σ representa o somatório enumerável, com respeito a soma (+) das expressões regulares, além disso, para fins de representação durante tal prova a letra X irá representar um alfabeto genérico qualquer.

Teorema 11.2 Se L é uma linguagem regular, então existe uma expressão regular r tal que $L = \mathcal{L}(r)$.

Demonstração. Sem perda de generalidade suponha que $L = \mathcal{L}(A)$ é uma linguagem regular reconhecida por um λ -AFN $A = \langle \{q_1, \dots, q_n\}, X, \delta, q_1, F \rangle$ em que $n \geq 1$. Agora será construído para todo $k \leq n$ a expressão regular $r_{i,j}^k$ é definida recursivamente como sendo:

$$r_{i,j}^0 = \begin{cases} \sum_{\delta(q_i, a) = q_j} a, & \text{se } i \neq j \\ \lambda + \sum_{\delta(q_i, a) = q_j} a, & \text{se } i = j \end{cases}$$

$$r_{i,j}^k = (r_{i,k}^{k-1} (r_{k,k}^{k-1})^* r_{k,j}^{k-1}) + r_{i,j}^{k-1}$$

note que $r_{i,j}^k$ é a expressão regular cuja valoração é o conjunto de todas as palavras $w \in X^*$ tal que $\widehat{\delta}(q_i, w) = q_j$ de forma que nenhum estado intermediário q_p com $p > k$ seja usado na computação (para detalhes deste fato consulte [15, 49]). Agora uma vez que A pode ter vários estados $q_f \in F$ tem-se então que cada $w \in X^*$ tal que $\widehat{\delta}(q_1, w) = q_f$ é uma palavra aceita por A , ou seja, $w \in \mathcal{L}(A)$ e, portanto, para

$n = \#Q$ a valoração da expressão regular r definida por,

$$r = \sum_{q_f \in F} r_{1,f}^n$$

é exatamente o conjunto de todas as palavras aceitas por A , ou seja, $\mathcal{L}(A) = \mathcal{L}(r)$. \square

■ **Exemplo 11.8** Considere o AFD esboçado pela Figura 11.9,

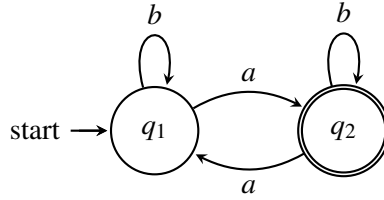


Figura 11.9: Um AFD que computa sobre o alfabeto $\{a, b\}$.

Como $n = \#Q$ tem-se que $n = 2$, e portanto, a expressão regular cuja valoração corresponde a linguagem do AFD será dada por,

$$r_{1,2}^2 = (r_{1,2}^1 (r_{2,2}^1)^* r_{2,2}^1) + r_{1,2}^1 \quad (11.1)$$

Note porém que,

$$\begin{aligned} r_{1,2}^1 &= (r_{1,1}^0 (r_{1,1}^0)^* r_{1,2}^0) + r_{1,2}^0 \\ &\equiv ((\lambda + b)(\lambda + b)^* a) + a \\ &\equiv b^* a \end{aligned} \quad (11.2)$$

e

$$\begin{aligned} r_{2,2}^1 &= (r_{2,1}^0 (r_{1,1}^0)^* r_{1,2}^0) + r_{2,2}^0 \\ &\equiv (a(\lambda + b)^* a) + (\lambda + b) \\ &\equiv ab^* a + \lambda + b \end{aligned} \quad (11.3)$$

substituindo as Equações (11.2) e (11.3) na Equação (11.1) tem-se que,

$$\begin{aligned} r_{1,2}^2 &= (b^* a (ab^* a + \lambda + b)^* ab^* a + \lambda + b) + (b^* a) \\ &\equiv b^* a (ab^* a + b)^* (ab^* a + \lambda) + b^* a \\ &\equiv b^* a (ab^* a + b)^* + b^* a \\ &\equiv b^* a (ab^* a + b)^* \end{aligned}$$

11.2 Gramática Regulares

Já foram apresentados anteriores dois formalismos para as linguagens regulares, a saber, formalismo operacional ou mecânico (os autômatos) e o formalismo denotacional (as expressões regulares). Nesta seção será apresentado um terceiro formalismo para as linguagens regulares, sendo este um formalismo gerador (ou axiomático) [74].

Definição 11.4 — Gramática Linear. Uma gramática formal $G = \langle V, \Sigma, S, P \rangle$ é dita Linear à Direita, ou simplesmente GLD, se todas as suas produções são da forma,

$$A \triangleright wB$$

e é dita Linear à Esquerda, ou simplesmente GLE, se todas as suas produções são da forma,

$$A \triangleright Bw$$

onde $A \in V, B \in V \cup \{\lambda\}$ e $w \in \Sigma^*$.

■ **Exemplo 11.9** A gramática $G_1 = \langle \{B, S, A\}, \{a, b\}, S, P \rangle$ onde P é formado pelas regras:

$$S \triangleright aaB$$

$$B \triangleright bb$$

é uma GLD.

■ **Exemplo 11.10** A gramática $G_1 = \langle \{X, S, Y\}, \{0, 1\}, S, P \rangle$ onde P é formado pelas regras:

$$S \triangleright X001$$

$$S \triangleright Y011$$

$$X \triangleright S01$$

$$Y \triangleright \lambda$$

é uma GLE.

Em uma gramática formal G quando para uma palavra w existem w_1, \dots, w_n tal que há as seguintes regras $w \triangleright w_1, \dots, w \triangleright w_n \in P$, é comum para simplificar a escrita do conjunto de regras usar a notação $w \triangleright w_1 \mid \dots \mid w_n$.

■ **Exemplo 11.11** Considere a GLE apresentada no Exemplo 11.10 o conjunto P da mesma poderia ser

escrito como:

$$S \triangleright X001 \mid Y011$$

$$X \triangleright S01$$

$$Y \triangleright \lambda$$

Definição 11.5 — Gramática Regular. Uma gramática formal G é dita regular sempre que ela for linear à esquerda ou à direita.

Um tipo mais rigoroso de gramática regular como comentado em [15, 61], são as chamadas gramática regulares unitárias definidas a seguir.

Definição 11.6 — Gramáticas Regulares Unitárias. Uma gramática regular G é dita unitária à esquerda (à direita) se ela é linear à esquerda (à direita) e toda produção é da forma $A \triangleright Bw$ ($A \triangleright wB$) com $A \in V, B \in V \cup \{\lambda\}$ e $w \in \Sigma \cup \{\lambda\}$.

■ **Exemplo 11.12** A gramática G do Exemplo 11.11 é uma gramática regular pois é uma GLE, porém não é uma gramática regular unitária.

Observação 11.4 De forma natural como o leitor deve estar pensando agora, duas gramática G_1 e G_2 serão ditas equivalentes sempre que elas gerarem a mesma linguagens.

O próximo resultado estabelece que gramática regulares e gramática regulares unitárias tem o mesmo poder de geração de linguagens.

Teorema 11.3 $L = \mathcal{L}(G)$ para alguma gramática regular G se, e somente se, existe uma gramática regular unitária G' na mesma direção (esquerda ou direita) tal que $L = \mathcal{L}(G')$.

Demonstração. (\Rightarrow) Suponha que $L = \mathcal{L}(G)$ para alguma gramática regular $G = \langle V, \Sigma, S, P \rangle$ tal que G seja linear à esquerda (a prova é similar para o caso à direita). Agora construa uma nova gramática $G' = \langle V', \Sigma, S, P' \rangle$ tal que P' é definido usando as seguintes regras:

R1: Se $A \triangleright Bw \in P$ onde $B \in V \cup \{\lambda\}, |w| \leq 1$, então $A \triangleright Bw \in P'$.

R2: Se $A \triangleright Ca_1 \cdots a_n \in P$ onde $B \in V$ e $a_i \in \Sigma$ sendo $1 \leq i \leq n$ e $n > 1$, então tem-se que $A \triangleright B_n a_n, B_n \triangleright B_{n-1} a_{n-1}, \dots, B_2 \triangleright Ca_1 \in P'$.

R3: Se $A \triangleright a_1 \cdots a_n \in P$ onde $a_i \in \Sigma$ sendo $1 \leq i \leq n$ e $n \geq 2$, então tem-se que $A \triangleright B_n a_n, B_n \triangleright B_{n-1} a_{n-1}, \dots, B_2 \triangleright B_1 a_1, B_1 \triangleright \lambda \in P'$.

Para as regras R2 e R3 todo B_i é uma nova variável existente em V' que não existe originalmente em V . Claramente a gramática G' é regular unitária à esquerda. Também não é difícil mostra por indução

sobre o tamanho das derivações que para todo $w \in \Sigma^*$ tem-se que $S \vdash_G^* w$ se, e somente se, $S \vdash_{G'}^* w$, portanto, $\mathcal{L}(G) = \mathcal{L}(G')$.

(\Leftarrow) Trivial uma vez que toda gramática regular unitária à esquerda (à direita) é um caso particular de gramática regular à esquerda (à direita). \square

■ **Exemplo 11.13** Considerando a gramática regular do Exemplo 11.11 usando o Teorema 11.3 é gerado a gramática regular unitária $G' = \langle \{S, B_3, B_2, C_3, C_2, X, D_2, Y\}, \{0, 1\}, S, P' \rangle$ onde P' é formado pelas regras:

$$\begin{aligned} S &\triangleright B_3 1 \mid C_3 1 \\ B_3 &\triangleright B_2 0 \\ B_2 &\triangleright X 0 \\ C_3 &\triangleright C_2 1 \\ C_2 &\triangleright Y 0 \\ X &\triangleright D_2 1 \\ D_2 &\triangleright S 0 \\ Y &\triangleright \lambda \end{aligned}$$

Observação 11.5 Obviamente a gramática do Exemplo 11.13 poderia ser otimizada para usar menos variáveis, porém otimização não é o foco de interesse no Teorema 11.3.

O próximo resultado estabelece o poder de geração das gramáticas regulares à direita.

Teorema 11.4 $L = \mathcal{L}(G)$ para alguma gramática regular à direita G se, e somente se, L é uma linguagem regular.

Demonstração. (\Rightarrow) Suponha que $L = \mathcal{L}(G)$ para alguma gramática regular à direita G , assim pelo Teorema 11.3 existe uma gramática regular unitária à direita $G' = \langle V, \Sigma, S, P \rangle$ tal que $L = \mathcal{L}(G')$, sem perda de generalidade² pode-se assumir que toda regra em P é da forma $A \triangleright aB$ ou $A \triangleright \lambda$ com $A, B \in V$ e $a \in \Sigma \cup \{\lambda\}$, dito isto, pode-se agora construir um λ -AFN $M = \langle V \cup \{q_f\}, \Sigma, \underline{\delta}_N, S, \{q_f\} \rangle$ tal que:

$$\begin{aligned} B \in \underline{\delta}_N(A, a) &\iff A \triangleright aB \in P \\ q_f \in \underline{\delta}_N(A, \lambda) &\iff A \triangleright \lambda \in P \end{aligned}$$

Agora será mostrado por indução sobre o tamanho das derivações em G' que se w é derivada por G' e

²Basta gerar uma nova gramática onde toda regra da forma $A \triangleright a$ com $a \in \Sigma$ foi substituída pelas regras $A \triangleright aC$ e $C \triangleright \lambda$ onde C é uma variável nova criada, obviamente a nova gramática continua equivalentes a antiga.

$w \in \Sigma^*$, então é aceita por M .

• **Base da indução:**

Quando w é derivada em G' com uma única derivação tem-se então duas situações possíveis:

- (1) Quando $w = \lambda$, obrigatoriamente existe uma regra da forma $S \triangleright \lambda$, e pela construção de M tem-se que $q_f \in \underline{\delta}_N(S, \lambda)$, logo $\widehat{\underline{\delta}_N}(S, \lambda) \cap \{q_f\} \neq \emptyset$ e, portanto, $\lambda \in L(M)$.
- (2) Quando $w = aB$, existe em P uma regra da forma $S \triangleright aB$ com $a \in \Sigma \cup \{\lambda\}$ e $B \in V$, assim pela construção de M tem-se que $B \in \underline{\delta}_N(A, a)$. Como $aB \notin \Sigma^*$ não há mais nada a fazer nesse caso.

• **Hipótese indutiva (HI):**

Suponha que $S \vdash_{G'}^* w$ em n derivação com $n \geq 1$ tal que:

- (1) Se $w \in \Sigma^*$, então $w \in \mathcal{L}(M)$.
- (2) Se $w = a_1 \cdots a_{n-1}B$ com $a_i \in \Sigma \cup \{\lambda\}$ para todo $1 \leq i \leq n-1$ e $B \in V$, então tem-se que $B \in \widehat{\underline{\delta}_N}(S, a_1 \cdots a_{n-1})$.

• **Passo indutivo:**

Agora dado que $S \vdash_{G'}^* w'$ em $n+1$ derivações, tem-se obrigatoriamente que acontece o caso (2) de **(HI)** e nesse caso duas situações são possíveis:

- (1) Se $w' \in \Sigma^*$, então $w = a_1 \cdots a_{n-1}B$ com $a_i \in \Sigma \cup \{\lambda\}$ para todo $1 \leq i \leq n-1$ e existe em P uma produção $B \rightarrow \lambda$, e assim, $w' = a_1 \cdots a_{n-1}$, nesta situação pelo caso (2) de **(HI)** tem-se que $B \in \widehat{\underline{\delta}_N}(S, a_1 \cdots a_{n-1})$ e como $B \rightarrow \lambda$ pela construção de M tem-se que $q_f \in \underline{\delta}_N(B, \lambda)$, consequentemente, $\widehat{\underline{\delta}_N}(S, a_1 \cdots a_{n-1}) \cap \{q_f\} \neq \emptyset$ e, portanto, $w' \in \mathcal{L}(M)$.
- (2) Se $w' = a_1 \cdots a_{n-1}B$ com $a_i \in \Sigma \cup \{\lambda\}$ para todo $1 \leq i \leq n-1$ e $B \in V$, então pela construção de M tem-se que $B \in \widehat{\underline{\delta}_N}(S, w)$ como $w' \notin \Sigma^*$ não há mais nada a fazer nesse caso.

Portanto, o raciocínio indutivo anterior garante que sempre que w é derivada por G' e $w \in \Sigma^*$ tem-se que $w \in \mathcal{L}(M)$ e assim pode-se afirmar pelo Corolário 10.2 que L é regular. (\Leftarrow) Suponha que L é uma linguagem regular assim por definição existe um AFD $M = \langle Q, \Sigma, \delta, q_0, F \rangle$ tal que $L = \mathcal{L}(M)$, assim construa uma gramática regular unitária à direita $G = \langle Q, \Sigma, q_0, P \rangle$ onde o conjunto P é definido usando as regras a seguir,

- (a) Se $\delta(q_i, a) = q_j$, então $q_i \triangleright aq_j \in P$.
- (b) Se $q_i \in F$, então $q_i \triangleright \lambda \in P$.

Agora note que para todo $w \in \Sigma^*$ com $w = a_1 \cdots a_n$ tem-se que,

$$\begin{aligned}
 w \in \mathcal{L}(M) &\iff \widehat{\delta}(q_0, w) \in F \\
 &\iff \widehat{\delta}(q_0, a_1 \cdots a_n) \in F \\
 &\iff (\exists q_f \in F)[\widehat{\delta}(q_0, w) = q_f] \\
 &\iff (\exists q_1, \dots, q_{n-1} \in Q, q_f \in F)[\delta(q_0, a_1) = q_1 \wedge \cdots \wedge \delta(q_{n-1}, a_n) = q_f] \\
 &\iff (\exists q_1, \dots, q_{n-1} \in Q, q_f \in F)[q_0 \triangleright a_1 q_1, \dots, q_{n-1} \triangleright a_n q_f, q_f \triangleright \lambda \in P] \\
 &\iff q_0 \vdash^* a_1 \cdots a_n \\
 &\iff q_0 \vdash^* w \\
 &\iff w \in \mathcal{L}(G)
 \end{aligned}$$

portanto, $\mathcal{L}(M) = \mathcal{L}(G)$ o que conclui a prova. \square

Lema 11.1 Se L é gerada por uma gramática à direita, então L' é gerada por uma gramática regular à direita.

Demonstração. Suponha que L é gerada por uma gramática à direita G , ou seja, $L = \mathcal{L}(G)$, assim pelo Teorema 11.4 tem-se que L é regular, logo existe um AFD $M = \langle Q, \Sigma, \delta, q_0, F \rangle$ tal que $L = \mathcal{L}(M)$, agora construa um λ -AFN $M_1 = \langle Q \cup \{q_f\}, \Sigma, \underline{\delta}_N, q_0, \{q_f\} \rangle$ tal que,

$$\underline{\delta}_N(q, a) = \begin{cases} \{\delta(q, a)\}, & \text{se } q \in Q, a \in \Sigma \\ \{q_f\}, & \text{se } q \in F, a = \lambda \\ \emptyset, & \text{qualquer outro caso} \end{cases}$$

claramente $L = \mathcal{L}(M_1)$, agora construa um novo λ -AFN $M_2 = \langle Q \cup \{q_f\}, \Sigma, \delta'_N, q_f, \{q_0\} \rangle$ onde para todo $q \in Q \cup \{q_f\}$ e $a \in \Sigma \cup \{\lambda\}$ tem-se que,

$$q_i \in \delta'_N(q_j, a) \iff q_j \in \underline{\delta}_N(q_i, a)$$

pela construção de M_2 é claro que $w \in \mathcal{L}(M_1) \iff w^r \in \mathcal{L}(M_2)$ e, portanto, $L' = \mathcal{L}(M_2)$. Desde que M_2 é um λ -AFN pelo Corolário 10.2 tem-se que L' é uma linguagem regular, consequentemente, pelo Teorema 11.4 existe uma gramática regular à direita G' tal que $L = \mathcal{L}(G')$, o que conclui a prova. \square

O próximo resultado mostra que gramática regulares à esquerda e à direita são equivalentes.

Teorema 11.5 — Mudança de direção regular. L é gerada por uma gramática à esquerda se, e somente se, L é gerada por uma gramática regular à direita.

Demonstração. (\Rightarrow) Suponha que L é gerada por uma gramática à esquerda $G = \langle V, \Sigma, S, P \rangle$, pelo Teorema 11.3 pode-se assumir que G é uma gramática regular unitária também a esquerda, assim todas as regras em P são da forma $A \triangleright Ba$ com $A \in V, B \in V \cup \{\lambda\}$ e $a \in \Sigma \cup \{\lambda\}$. Sem perda de generalidade³ pode-se construir uma nova gramática regular unitária à esquerda $G' = \langle V', \Sigma, S, P' \rangle$ onde toda regra em P' é da forma $A \triangleright Ba$ ou $A \triangleright \lambda$ com $A, B \in V'$ e $a \in \Sigma \cup \{\lambda\}$ claramente pela construção de G' tem-se que $\mathcal{L}(G) = \mathcal{L}(G')$, agora construa um λ -AFN $M = \langle V' \cup \{q_f\}, \Sigma, \underline{\delta}_N, S, \{q_f\} \rangle$ onde,

$$\begin{aligned} B \in \underline{\delta}_N(A, a) &\iff A \triangleright Ba \in P \\ q_f \in \underline{\delta}_N(A, \lambda) &\iff A \triangleright \lambda \in P \end{aligned}$$

Agora note que para todo $w = a_1 \cdots a_n \in \Sigma^*$ tem-se que,

$$\begin{aligned} w \in \mathcal{L}(G') &\iff a_1 \cdots a_n \in \mathcal{L}(G') \\ &\iff S \vdash_{G'}^* a_1 \cdots a_n \\ &\iff (\exists A_1 \cdots A_n, S \in V) \\ &\quad [S \triangleright A_n a_n, A_n \triangleright A_{n-1} a_{n-1}, \dots, A_2 \triangleright A_1 a_1, A_1 \triangleright \lambda \in P'] \\ &\iff (\exists A_1 \cdots A_n, S \in V) \\ &\quad [A_n \in \underline{\delta}_N(S, a_n), A_{n-1} \in \underline{\delta}_N(A_n, a_{n-1}), \dots, A_1 \in \underline{\delta}_N(A_2, a_1), \\ &\quad q_f \in \underline{\delta}_N(A_1, \lambda)] \\ &\iff a_n \cdots a_1 \in \mathcal{L}(M) \\ &\iff w^r \in \mathcal{L}(M) \end{aligned}$$

Logo $\mathcal{L}(M) = \mathcal{L}(G')^r$, desde que M é um λ -AFN tem-se pelo Corolário 10.2 que $\mathcal{L}(G')^r$ é regular, assim pelo Teorema 11.4 existe uma gramática regular à direita \hat{G}_1 que a gera, ou seja, $\mathcal{L}(\hat{G}_1) = \mathcal{L}(G')^r$, mas pelo Lema 11.1 irá existir outra gramática regular à direita \hat{G}_2 tal que $\mathcal{L}(\hat{G}_2) = \mathcal{L}(\hat{G}_1)^r$, mas $\mathcal{L}(\hat{G}_1)^r = (\mathcal{L}(G')^r)^r = (\mathcal{L}(G)^r)^r = (L^r)^r = L$, portanto, L é gerada por uma gramática regular à direita. (\Leftarrow) Suponha que L é gerada por uma gramática regular à direita, ou seja, que existe uma gramática regular a direita G tal que $L = \mathcal{L}(G)$, assim pelo Lema 11.1 irá existir outra gramática regular à direita $G_1 = \langle V, \Sigma, S, P \rangle$ tal que $L^r = \mathcal{L}(G_1)$, sem perda de generalidade pelo Teorema 11.3 pode-se assumir que G_1 é regular unitária à direita, logo todas as suas produções são da forma $A \triangleright aB$ com $A \in V, B \in V \cup \{\lambda\}$ e $a \in \Sigma \cup \{\lambda\}$. Dito isso construa uma nova gramática $G_2 = \langle V, \Sigma, S, P' \rangle$ onde $P' = \{A \triangleright Ba \mid A \triangleright Ba \in P\}$, claramente G_2 é unitária à esquerda. Mas pela construção de G_2 fica claro que $S \vdash_{G_1}^* w \iff S \vdash_{G_2}^* w^r$,

³Basta gerar uma nova gramática onde toda regra da forma $A \triangleright a$ com $a \in \Sigma$ foi substituída pelas regras $A \triangleright Ca$ e $C \triangleright \lambda$ onde C é uma variável nova criada.

logo $\mathcal{L}(G_1)^r = \mathcal{L}(G_2)$, mas desde que, $\mathcal{L}(G_1)^r = (L^r)^r = L$, tem-se então que L é gerada por uma gramática linear à esquerda, o que completa a prova. \square

■ **Exemplo 11.14** Dado a gramática regular à direita $G_1 = \langle \{A, B, C\}, \{a, b\}, A, P_1 \rangle$ com P_1 é formado pelas seguintes regras,

$$A \triangleright aC \mid B$$

$$B \triangleright bB \mid \lambda$$

$$C \triangleright aA$$

claramente $\mathcal{L}(G_1) = \{w \in \{a, b\}^* \mid w = a^{2m}b^n \text{ com } m, n \in \mathbb{N}\}$, agora usando a construção exposta pelo Teorema 11.5, é possível construir a gramática regular à esquerda $G_2 = \langle \{A, B, C\}, \{a, b\}, B, P_2 \rangle$ onde P_2 é formado pelas seguintes regras,

$$B \triangleright Bb \mid Ab \mid A$$

$$A \triangleright Ca \mid \lambda$$

$$C \triangleright Aa$$

e obviamente $\mathcal{L}(G_2) = \{w \in \{a, b\}^* \mid w = a^{2m}b^n \text{ com } m, n \in \mathbb{N}\}$.

11.3 Questionário

■ **Exercício 11.1** Mostre uma expressão regular cuja valoração é corresponde exatamente a linguagem $\{aabb, aaabbb, aaba, \lambda\}$.

■ **Exercício 11.2** Construa uma expressão regular para cada uma das linguagens descrita no Exercício 10.3.

■ **Exercício 11.3** Construa uma expressão regular para cada uma das linguagens descrita no Exercício 10.20.

■ **Exercício 11.4** Para cada um dos AFD A , que foram criados por você para responder o Exercício 10.2, determine uma expressão regular r tal que $\mathcal{L}(r) = \mathcal{L}(A)$.

■ **Exercício 11.5** Para cada expressão regular a seguir construa um autômato que aceita a linguagem da expressão regular.

(a). $a^*b + a$.

(b). $(ab + cd)^*$.

(c). $(a^*ab)^* + bc$.

(d). $(aa)^*(b + \lambda) + \emptyset$.

(e). $(a^*) + b(aa)^*$.

■ **Exercício 11.6** Dado o alfabeto $\{a, b\}$ demonstre as asserções a seguir.

(a). $(a + b)^* \equiv (a^*b^*)^*$.

(b). $a^* + b^* \not\equiv (a + b)^*$.

(c). $a^*b^* \not\equiv (ab)^*$.

(d). $(b + ab)^*(a + \lambda) \equiv (a + \lambda)(ba + b)^*$.

(e). $a^*a \equiv aa^*$.

(f). $\emptyset + (a + b)^* \equiv (a + b)^* + \emptyset$.

■ **Exercício 11.7** Para quais quer $r_1, r_2, r_3 \in \text{Exp}_\Sigma$ demonstre as asserções a seguir.

(a). $(r_1^*)^* \equiv r_1^*$.

(b). $(r_1 + r_2) \equiv r_2 + r_1$.

(c). $r_1 + (r_2 + r_3) \equiv (r_1 + r_2) + r_3$.

(d). $r_1 + r_1 \equiv r_1$.

(e). $(r_1 + r_2)^* \equiv (r_1^*r_2^*)^*$.

(f). $r_1r_1 \equiv r_1$.

(g). $(r_1r_2)^*r_1 \equiv r_1(r_2r_1)^*$.

(h). $r_1 + \emptyset \equiv r_1$.

■ **Exercício 11.8** Construa um autômato que aceite cada valoração a seguir.

(a). $\mathcal{L}(aa^*(b + a))$.

(b). $\mathcal{L}((aa + abb)^*(aa + \lambda + ba))$.

(c). $\mathcal{L}((ab + b)^*(a + \lambda))$.

(d). $\mathcal{L}(aa^* + aba^*b^*)$.

- (e). $\mathcal{L}((\lambda + \emptyset)^*)$.
- (f). $\mathcal{L}(aa^*bb^*aa^*)$.
- (g). $\mathcal{L}(ab + (\emptyset + a)^*b)$.
- (h). $\mathcal{L}((a^*(b(bb)^*aa^*)a)$.
- (i). $\mathcal{L}(((aa)^*)^*)$.
- (j). $\mathcal{L}((b+a)(\lambda^*)^*)$.
- (k). $\mathcal{L}(a^*(ba+ab))$.
- (l). $\mathcal{L}((bb+bab)^*(a+\lambda aa^*+ba))$.
- (m). $\mathcal{L}((b^*+b)^*(a^*+\lambda))$.
- (n). $\mathcal{L}(a^*+ba)$.
- (o). $\mathcal{L}(((\lambda)^*+\emptyset)^*)$.
- (p). $\mathcal{L}(a^*ba^*ba^*)$.
- (q). $\mathcal{L}(bab + (ab+a)b)$.
- (r). $\mathcal{L}(a(b)^*b^*)$.
- (s). $\mathcal{L}((b+a)(b+a))$.

■ **Exercício 11.9** Construa uma expressão regular cuja valoração seja exatamente a linguagem aceita pelo AFD da Figura 10.21.

■ **Exercício 11.10** Construa uma gramática regular à esquerda para cada uma das linguagens descrita no Exercício 10.3.

■ **Exercício 11.11** Construa uma gramática regular à direita para cada uma das linguagens descrita no Exercício 10.3.

■ **Exercício 11.12** Construa uma gramática regular à esquerda para cada uma das linguagens descrita no Exercício 10.20.

■ **Exercício 11.13** Construa uma gramática regular à direita para cada uma das linguagens descrita no Exercício 10.20.

■ **Exercício 11.14** Construa uma gramática regular que gera exatamente cada valoração a seguir.

(a). $\mathcal{L}(a^*b + a)$.

(b). $\mathcal{L}((ab + cd)^*)$.

(c). $\mathcal{L}((a^*ab)^* + bc)$.

(d). $\mathcal{L}((aa)^*(b + \lambda))$.

(e). $\mathcal{L}((a^*) + b(aa)^*)$.

■ **Exercício 11.15** Construa uma gramática regular que gera exatamente a linguagem aceita pelo AFD da Figura 10.21

■ **Exercício 11.16** Para as gramáticas descrita a seguir construa um AFN que aceita as linguagens geradas por tais gramáticas.

(a). $G_1 = \langle \{S, A, B\}, \{a, b\}, S, P \rangle$ onde P é definido por,

$$S \triangleright aA \mid bB \mid aaS \mid bbS$$

$$A \triangleright aA \mid \lambda$$

$$B \triangleright bB \mid b$$

(b). $G_2 = \langle \{S, A, B\}, \{a, b\}, S, P \rangle$ onde P é definido por,

$$S \triangleright abA$$

$$A \triangleright bab$$

$$B \triangleright aA \mid bb$$

(c). $G_3 = \langle \{S, A, B\}, \{a, b\}, S, P \rangle$ onde P é definido por,

$$S \triangleright aA \mid bS \mid \lambda$$

$$A \triangleright aB \mid bS \mid \lambda$$

$$B \triangleright aaS \mid bS \mid \lambda$$

(d). $G_4 = \langle \{S, A\}, \{a, b\}, S, P \rangle$ onde P é definido por,

$$S \triangleright aaB \mid b$$

$$A \triangleright bbS$$

(e). $G_5 = \langle \{S, A, B\}, \{a, b\}, S, P \rangle$ onde P é definido por,

$$S \rhd aaaA \mid bbB$$

$$A \rhd abaA \mid S$$

$$B \rhd bbS \mid aA \mid bb$$

■ **Exercício 11.17** Para cada gramática do Exercício 11.16 esboce uma gramática regular unitária à esquerda equivalente.

■ **Exercício 11.18** Esboce uma gramática regular que gere a linguagem aceita pelo AFD da Figura 11.9

Capítulo 12

Álgebra das Linguagens Regulares

Você acha que VOCÊ tem problemas?

Experimente ser um robô maníaco
depressivo...

Douglas Noël Adams, O Guia do

Mochileiro das Galáxias.

12.1 Operadores de Fecho

Neste capítulo as serão estudadas as operações que fecham algebricamente o conjunto (ou classe) de todas as linguagens regulares \mathcal{L}_{Reg} , isto é, serão apresentados aqui o operadores de fecho para as linguagens regulares. A maior parte dos resultados demonstrados aqui foi provado inicialmente no seminal *paper* [92].

Teorema 12.1 — Fecho do Complemento. Se L é uma linguagem regular, então \bar{L} também é uma linguagem regular.

Demonstração. Assuma que L seja uma linguagem regular, assim por definição existe um AFD $A = \langle Q, \Sigma, \delta, q_0, F \rangle$ tal que $\mathcal{L}(A) = L$, agora defina um novo AFD $\bar{A} = \langle Q, \Sigma, \delta, q_0, \bar{F} \rangle$ tal que $\bar{F} = Q - F$, agora para todo $w \in \Sigma^*$ tem-se que:

$$\begin{aligned} w \in \mathcal{L}(\bar{A}) &\iff \hat{\delta}(q_0, w) \in \bar{F} \\ &\iff \hat{\delta}(q_0, w) \notin F \\ &\iff w \notin \mathcal{L}(A) \end{aligned}$$

consequentemente, $\mathcal{L}(\bar{A}) = \bar{L}$, e desde que \bar{A} é um AFD, tem-se que \bar{L} também é uma linguagem regular, concluindo assim a prova. □

O AFD construído na demonstração do Teorema 12.1 é chamado de **autômato complementar**, o exemplo a seguir mostra a construção de um autômato complementar de um AFD dado.

■ **Exemplo 12.1** Considere o AFD A representado na Figura 12.1 que reconhece a linguagem de todas as palavras com a quantidade par de a 's e uma quantidade qualquer de b 's.

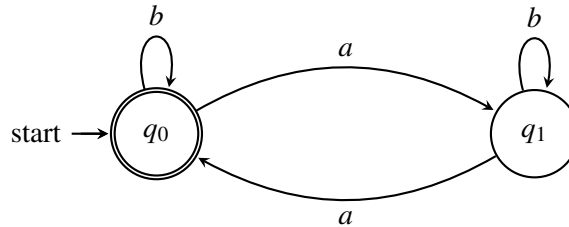


Figura 12.1: AFD A para a linguagem $\{w \in \{a,b\}^* \mid |w_a| = 2n \text{ com } n, |w|_b \in \mathbb{N}\}$.

O AFD complementar ao AFD da Figura 12.1 pode ser visto na Figura 12.2.

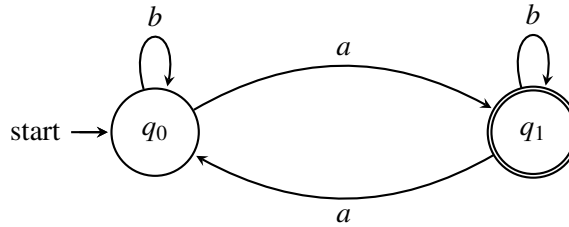


Figura 12.2: AFD A para a linguagem $\{w \in \{a,b\}^* \mid |w_a| = 2n + 1 \text{ com } n, |w|_b \in \mathbb{N}\}$.

■ **Exemplo 12.2** Seguindo a estratégia exposta na prova do Teorema 12.1 o AFD complementar ao AFD da Figura 10.2 corresponde exatamente ao AFD representado na Figura 12.3, e é claro que a linguagem de tal AFD corresponde exatamente a linguagem $\{0,1\}^*$

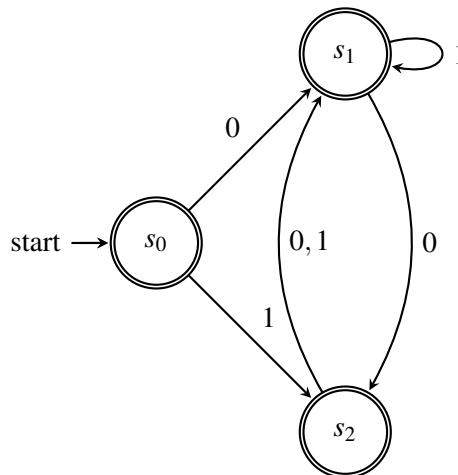


Figura 12.3: AFD complementar ao AFD da Figura 10.2.

Para os dois próximos resultados considere sempre que as linguagens são definidas sobre o mesmo alfabeto, isto é, $L_1, L_2 \subseteq \Sigma^*$.

Teorema 12.2 — Fecho do União. Se L_1 e L_2 são linguagens regulares, então $L_1 \cup L_2$ também é uma linguagem regular.

Demonstração. Assuma que L_1 e L_2 são linguagens regulares, logo existem $A_1 = \langle Q, \Sigma, \delta_1, q_0, F_1 \rangle$ e $A_2 = \langle S, \Sigma, \delta_2, s_0, F_2 \rangle$ com A_1 e A_2 sendo AFD e, além disso, $\mathcal{L}(A_1) = L_1$ e $\mathcal{L}(A_2) = L_2$, sem perda de generalidade assumamos que $Q \cap S = \emptyset$, agora construa um λ -AFN $A = \langle Q \cup S \cup \{q_{init}\}, \Sigma, \underline{\delta}_N, q_{init}, F \rangle$ tal que $q_{init} \notin Q \cup S$ e $F = F_1 \cup F_2$, além disso, tem-se que:

$$\underline{\delta}_N(q, a) = \begin{cases} \{\delta_1(q, a)\}, & \text{se } q \in Q, a \in \Sigma \\ \{\delta_2(q, a)\}, & \text{se } q \in S, a \in \Sigma \\ \{q_0, s_0\}, & \text{se } q = q_{init}, a = \lambda \\ \emptyset, & \text{qualquer outro caso} \end{cases}$$

agora para todo $w \in \Sigma^*$ tem-se que,

$$w \in \mathcal{L}(A) \iff \widehat{\underline{\delta}_N}(q_{init}, w) \cap F \neq \emptyset$$

mas pela construção de $\underline{\delta}_N$ tem-se que,

$$\begin{aligned} \widehat{\underline{\delta}_N}(q_{init}, w) \cap F \neq \emptyset &\iff \{\widehat{\delta}_1(q_0, w)\} \cup \{\widehat{\delta}_2(s_0, w)\} \neq \emptyset \\ &\iff \{\widehat{\delta}_1(q_0, w)\} \neq \emptyset \text{ ou } \{\widehat{\delta}_2(s_0, w)\} \neq \emptyset \\ &\iff (\exists q_i \in F_1)[\widehat{\delta}_1(q_0, w) = q_i] \text{ ou } (\exists s_j \in F_2)[\widehat{\delta}_2(s_0, w) = s_j] \\ &\iff w \in \mathcal{L}(A_1) \text{ ou } w \in \mathcal{L}(A_2) \\ &\iff w \in \mathcal{L}(A_1) \cup \mathcal{L}(A_2) \\ &\iff w \in L_1 \cup L_2 \end{aligned}$$

assim $\mathcal{L}(A) = L_1 \cup L_2$, e desde que A é um λ -AFN pelo Corolário 10.2 tem-se que $L_1 \cup L_2$ será uma linguagem regular, o completa a prova. \square

Teorema 12.3 — Fecho do Interseção. Se L_1 e L_2 são linguagens regulares, então $L_1 \cap L_2$ também é uma linguagem regular.

Demonstração. Assuma que L_1 e L_2 são duas linguagens regulares, assim existem dois AFD A_1 e A_2 com $\mathcal{L}(A_1) = L_1$ e $\mathcal{L}(A_2) = L_2$, pelo Teorema 12.1 tem-se que existem dois AFD A'_1 e A'_2 tais que $\mathcal{L}(A'_1) = \overline{\mathcal{L}(A_1)}$ e $\mathcal{L}(A'_2) = \overline{\mathcal{L}(A_2)}$, agora pelo Teorema 12.2 tem-se que $\overline{\mathcal{L}(A_1)} \cup \overline{\mathcal{L}(A_2)}$ é uma linguagem regular, logo existe um AFD A_0 tal que $\mathcal{L}(A_0) = \overline{\mathcal{L}(A_1)} \cup \overline{\mathcal{L}(A_2)}$, e novamente pelo Teorema

12.1 tem-se que $\overline{\mathcal{L}(A_1)} \cup \overline{\mathcal{L}(A_2)}$ é uma linguagem regular, mas pelas operações sobre conjunto tem-se que $\overline{\mathcal{L}(A_1) \cup \mathcal{L}(A_2)} = \mathcal{L}(A_1) \cap \mathcal{L}(A_2)$, consequentemente, $L_1 \cap L_2$ é regular. \square

Note que o Teorema 12.3 determinar que a interseção de duas linguagens regulares ainda é uma linguagem regular, mas na prova do mesmo não é mostrado um autômato que reconhece tal linguagem, o que contrasta com a prova do Teorema 12.2, apesar disso, através do Teorema 12.3 pode-se concluir o resultado a seguir.

Corolário 12.1 Dado duas linguagem regulares L_1 e L_2 , existe um AFD A tal que $\mathcal{L}(A) = L_1 \cap L_2$.

Demonstração. Assuma que L_1 e L_2 são duas linguagens regulares, assim por definição existem dois AFD $A_1 = \langle Q, \Sigma, \delta_1, q_0, F_1 \rangle$ e $A_2 = \langle S, \Sigma, \delta_2, s_0, F_2 \rangle$ tal que $\mathcal{L}(A_1) = L_1$ e $\mathcal{L}(A_2) = L_2$, sem perda de generalidade assuma que $Q \cap S = \emptyset$, agora construa o autômato $A = \langle Q \times S, \Sigma, \delta, (q_0, s_0), F_1 \times F_2 \rangle$ onde para todo $(q, s) \in Q \times S$ e $a \in \Sigma$ tem-se que,

$$\delta((q, s), a) = (\delta_1(q, a), \delta_2(s, a))$$

desde que δ_1 e δ_2 são funções totais, tem-se que δ também será total, além disso, como $F_1 \times F_2 \subseteq Q \times S$, pode-se concluir que A é um autômato bem definido. Agora desde que A_1 e A_2 são ambos AFD e pela construção de δ é claro que A também será um AFD, além disso, por indução sobre o tamanho das palavras w tem-se para todo $(q, s) \in Q \times S$ que,

$$\widehat{\delta}((q, s), w) = (\widehat{\delta}_1(q, w), \widehat{\delta}_2(s, w))$$

por fim note que para todo $w \in \Sigma^*$ tem-se que,

$$\begin{aligned} w \in \mathcal{L}(A) &\iff \widehat{\delta}((q_0, s_0), w) \in F_1 \times F_2 \\ &\iff (\widehat{\delta}_1(q_0, w), \widehat{\delta}_2(s_0, w)) \in F_1 \times F_2 \\ &\iff \widehat{\delta}_1(q_0, w) \in F_1 \text{ e } \widehat{\delta}_2(s_0, w) \in F_2 \\ &\iff w \in \mathcal{L}(A_1) \text{ e } w \in \mathcal{L}(A_2) \\ &\iff w \in \mathcal{L}(A_1) \cap \mathcal{L}(A_2) \end{aligned}$$

portanto, tem-se que $\mathcal{L}(A) = L_1 \cap L_2$, o que conclui a prova. \square

O próximo operador de fecho para as linguagens regulares é a concatenação de linguagem, definida anteriormente (Definição 9.11) no Capítulo 9.

Capítulo 13

Linguagens Livres do Contexto

Capítulo 14

Computabilidade à Turing

“As máquinas me surpreendem muito
frequentemente.”

Alan M. Turing

Bibliografia

- [1] J. M. Abe. *Introdução à Lógica para a Ciência da Computação*. Arte & Ciência, 2002.
- [2] J. M. Abe and N. Papavero. *Teoria Intuitiva dos Conjuntos*. MAKRON Books, 1991.
- [3] U. R. Acharya, P. S. Bhat, S. S. Iyengar, A. Rao, and S. Dua. Classification of heart rate data using artificial neural network and fuzzy equivalence relation. *Pattern recognition*, 36(1):61–68, 2003.
- [4] A. V. AHO, M. S. LAM, R. SETHI, and J. D. ULLMAN. *Compiladores: Princípios, Técnicas e ferramentas*. Editora Pearson, 2ª edição edition, 2007.
- [5] W. AN and B. Russell. *Principia Mathematica*, 1910.
- [6] J. Avigad. Handbook of proof theory. In *Studies in Logic and the Foundations of Mathematics*, ch. CiteSeer, 1998.
- [7] M. Ayala-Rincón and F. L. C. de Moura. *Fundamentos da Programação Lógica e Funcional – O princípio de Resolução e a Teoria de Reescrita*. Editora UnB, 2014.
- [8] A. Bar-Hillel, T. Hertz, N. Shental, and D. Weinshall. Learning distance functions using equivalence relations. In *Proceedings of the 20th international conference on machine learning (ICML-03)*, pages 11–18, 2003.
- [9] H. P. Barendregt. *The Lambda Calculus Its Syntax and Semantics*. Elsevier, 1984.
- [10] H. P. Barendregt. *Lambda Calculi with Types*. Oxford: Clarendon Press, 1992.
- [11] J. M. Barreto, M. Roisenberg, M. A. F. Almeida, and K. Collazos. Fundamentos de Matemática Aplicada a Informática. Acessado em 06/06/2021 na página <http://www.inf.ufsc.br/~mauro.roisenberg/ine5381/leituras/apostila.pdf>, 1998.
- [12] B. Bedregal. λ -ALN: Autômatos Lineares Não-determinísticos com λ -Transições. *TEMA - Tendências em Matemática Aplicada e Computacional*, 12(3):171–182, 2011.

- [13] B. Bedregal. Nondeterministic Linear Automata and a Class of Deterministic Linear Languages. *Preliminary Proceedings LSFA*, pages 183–196, 2015.
- [14] B. Bedregal and B. M. Acióly. Introdução à Lógica Clássica para a Ciência da Computação. Notas de aula, 2007.
- [15] B. Bedregal, B. M. Acióly, and A. Lyra. *Introdução à Teoria da Computação: Linguagens Formais, Autômatos e Computabilidade*. Editora UnP, Natal, 2010.
- [16] Y. Bertot and P. Castéran. *Interactive Theorem Proving and Program Development: Coq’Art: The Calculus of Inductive Constructions*. Springer Science & Business Media, 2013.
- [17] K. Bimbó. *Combinatory Logic: Pure, Applied and Typed*,. Chapman and Hall/CRC, 2019.
- [18] K. Broda, J. Ma, G. Sinnadurai, and A. Summers. Pandora: A reasoning toolbox using natural deduction style. *Logic Journal of the IGPL*, 15(4):293–304, 2007.
- [19] S. R. Buss. *Handbook of proof theory*. Elsevier, 1998.
- [20] G. Cantor. Beiträge zur Begründung der Transfiniten Mengenlehre. *Mathematische Annalen*, 46(4):481–512, 1895.
- [21] J. Carmo, P. Gouveia, and F. M. Dionísio. *Elementos de Matemática Discreta*. College Publications, 2013.
- [22] J. Carroll and D. Long. *Theory of finite automata: with an introduction to formal languages*. Prentice Hall Upper Saddle River, NJ, New Jersey, 2^a edition edition, 1989.
- [23] G. J. Chaitin. Computers, paradoxes and the foundations of mathematics: Some great thinkers of the 20th century have shown that even in the austere world of mathematics, incompleteness and randomness are rife. *American Scientist*, 90(2):164–171, 2002.
- [24] N. Chomsky. Three models for the description of language. *IRE Transactions on information theory*, 2(3):113–124, 1956.
- [25] K. Cooper and L. Torczon. *Construindo Compiladores*, volume 1. Elsevier Brasil, 2017.
- [26] I. M. Copi. *Introdução à Lógica*. Mestre Jou, 1981.
- [27] V. S. Costa. Linguagens Lineares Fuzzy. Master’s thesis, Programa de Pós-graduação em Sistemas e Computação, Universidade Federal do Rio Grande do Norte, UFRN, Natal, RN, 2016.

- [28] V. S. Costa. *Autômatos Fuzzy Hesitantes Típicos: Teoria e Aplicações*. PhD thesis, Programa de Pós-graduação em Sistemas e Computação, Universidade Federal do Rio Grande do Norte, UFRN, Natal, RN, 2020.
- [29] N. Cressie and J. L. Davidson. Image analysis with partially ordered markov models. *Computational statistics & data analysis*, 29(1):1–26, 1998.
- [30] J. I. da Silva Filho. Lógica Paraconsistente e Probabilidade Pragmática no Tratamento de Incertezas. *Revista Seleção Documental*, (9):16–27, 2008.
- [31] E. de Alencar Filho. *Iniciação à Lógica Matemática*. NBL Editora, 2002.
- [32] C. De la Higuera. *Grammatical inference: Learning Automata and Grammars*. Cambridge University Press, London, 2010.
- [33] A. A. de Lima. *Conjuntos fuzzy multidimensionais*. PhD thesis, Programa de Pós-graduação em Sistemas e Computação, Universidade Federal do Rio Grande do Norte, UFRN, Natal, RN, 2019.
- [34] J. N. de Souza. *Lógica para Ciência da Computação e Áreas Afins*. Elsevier Brasil, 2008.
- [35] S. S. Epp. *Discrete Mathematics With Applications*. Cengage learning, 2010.
- [36] A. D. S. Farias, V. S. Costa, R. H. Santiago, and B. Bedregal. The image reduction process based on generalized mixture functions. In *2016 Annual Conference of the North American Fuzzy Information Processing Society (NAFIPS)*, pages 1–6. IEEE, 2016.
- [37] A. D. S. Farias, V. S. Costa, R. H. N. Santiago, and B. Bedregal. A Residuated Function in a Class of Mealy Type \mathcal{L} -Valued Finite Automaton. In *Fuzzy Information Processing Society (NAFIPS), 2016 Annual Conference of the North American*, pages 1–6, El Paso, TX, USA, 2016. IEEE.
- [38] P. Feofiloff. *Algoritmos em linguagem C*. Elsevier Brasil, 2009.
- [39] F. B. Fitch. Symbolic Logic, An Introduction. *American Journal of Physics*, 21(3):237–237, 1953.
- [40] J. L. Gersting. *Fundamentos Matemáticos para Ciência da Computação*. GrupoGen LTC, 2021.
- [41] D. Giri and P. Srivastava. A cryptographic key assignment scheme for access control in poset ordered hierarchies with enhanced security. *Int. J. Netw. Secur.*, 7(2):223–234, 2008.
- [42] K. Gödel. Über formal unentscheidbare Sätze der Principia Mathematica und verwandter Systeme I. *Monatshefte für mathematik und physik*, 38(1):173–198, 1931.
- [43] P. R. Halmos. *Teoria ingênua dos conjuntos*. Editora Ciência Moderna, 2001.

- [44] J. Halpern, Z. Manna, and B. Moszkowski. A hardware Semantics Based on Temporal Intervals. In *International Colloquium on Automata, Languages, and Programming*, pages 278–291. Springer, 1983.
- [45] D. Harel et al. First-order Dynamic Logic. *Lecture Notes Computer Sciences*, (9):133, 1979.
- [46] F. Hausdorff. *Grundzüge der mengenlehre*, volume 7. von Veit, 1914.
- [47] W. Hodges et al. *A Shorter Model Theory*. Cambridge university press, 1997.
- [48] B. Holdsworth and C. Woods. *Digital Logic Design*. Elsevier, 4^a edition, 2002.
- [49] J. E. Hopcroft, R. Motwani, and J. D. Ullman. *Introduction to Automata Theory, Languages and Computation*. Pearson Education India, USA, 3^a edition, 2008.
- [50] I. V. Idoeta and F. G. Capuano. *Elementos de Eletrônica Digital*. Saraiva Educação SA, 41^a edition, 2018.
- [51] S. Jaskowski. On the rules of supposition in formal logic in the series. *Studia Logica: Wydawnictwo Poswiecone Logice i jej Historii*, 1934.
- [52] R. Joosse. An Extended Algorithm for Learning Serial Compositions of Mealy Machines. Dissertação de mestrado, Radboud University, Nimega, Netherlands, July 2019.
- [53] E. Karaman, M. Soyertem, İ. Atasever Güvenç, D. Tozkan, M. Küçük, and Y. Küçük. Partial order relations on family of sets and scalarizations for set optimization. *Positivity*, 22(3):783–802, 2018.
- [54] S. C. Kleene. Representation of events in nerve nets and finite automata. Technical report, Rand Project Air Force Santa Monica CA, 1951.
- [55] P. J. Landin. The mechanical evaluation of expressions. *The computer journal*, 6(4):308–320, 1964.
- [56] P. J. Landin. Correspondence between algol 60 and church’s lambda-notation: part i. *Communications of the ACM*, 8(2):89–101, 1965.
- [57] O. Levin. *Discrete mathematics: An open introduction*. 2021.
- [58] X. Li, X. Huang, Z. Nie, and Y. Zhang. Equivalent relations between interchannel coupling and antenna polarization coupling in polarization diversity systems. *IEEE transactions on antennas and propagation*, 55(6):1709–1715, 2007.
- [59] T. Y. Lin. Data mining: Granular computing approach. In *Pacific-Asia Conference on Knowledge Discovery and Data Mining*, pages 24–33. Springer, 1999.

- [60] P. Lingras and Y. Yao. Data mining using extensions of the rough set model. *Journal of the American society for information science*, 49(5):415–422, 1998.
- [61] P. Linz. *An Introduction to Formal Languages and Automata*. Jones & Bartlett Learning, USA, 2006.
- [62] S. Lipschutz. *Topologia Geral*. McGRAW-HILL Do Brasil, LTDA/MEC, 1971. Coleção Schaum.
- [63] S. Lipschutz. *Teoria dos Conjuntos*. McGraw-Hill do Brasil - LTDA/MEC, 1978.
- [64] S. Lipschutz and M. Lipson. *Matemática Discreta*. Bookman Editora, 2013. Coleção Schaum.
- [65] A. C. d. Lourenço, E. C. A. Cruz, S. R. Ferreira, and S. C. Júnior. *Circuitos Digitais Estude e Use*. Editora Érica, 4^a edition, 1996.
- [66] C. A. Lungarzo. La Consistencia de la Lógica Intuicionista. *Tarea*, 3:119–132, 1972.
- [67] P. D. Magnus, T. Button, A. Thomas-Bolduc, R. Zach, and R. Trueman. *Forall x: Calgary. An Introduction to Formal Logic*. Fall 2020, 2020.
- [68] Z. Manna and A. Pnueli. The Modal Logic of Programs. In *International Colloquium on Automata, Languages, and Programming*, pages 385–409. Springer, 1979.
- [69] J. C. Martin. *Introduction to Languages and The Theory of Computation*. McGraw Hill, 4^a edição edition, 2003.
- [70] J. P. Martins. *Lógica e Raciocínio*. College Publications, 2014.
- [71] Y. V. Matiyasevich, J. V. Matijasevič, Ů. V. Matiâsevič, Y. V. Matiyasevich, Y. V. Matiyasevich, M. R. Garey, and A. Meyer. *Hilbert's tenth problem*. MIT press, 1993.
- [72] W. S. McCulloch and W. Pitts. A logical calculus of the ideas immanent in nervous activity. *The Bulletin of Mathematical Biophysics*, 5(4):115–133, 1943.
- [73] G. H. Mealy. A method for synthesizing sequential circuits. *The Bell System Technical Journal*, 34(5):1045–1079, 1955.
- [74] P. B. Menezes. *Linguagens Formais e Autômatos*. Sagra-Dcluzzato, 1998.
- [75] P. B. Menezes. *Matemática Discreta para Computação e Informática*, volume 2. Bookman, 2010.
- [76] T. R. B. Milfont. *Grafos fuzzy intervalares n-dimensionais*. PhD thesis, Programa de Pós-graduação em Sistemas e Computação, Universidade Federal do Rio Grande do Norte, UFRN, Natal, RN, 2021.

- [77] E. F. Moore. Gedanken-experiments on sequential machines. *Automata Studies*, 34:129–153, 1956.
- [78] J. Morgado. *Introdução à Teoria dos Reticulados, Textos de Matemática*. Instituto de Física e Matemática, Recife, 1962.
- [79] C. A. Mortari. *Introdução à Lógica*. Unesp, 2001.
- [80] L. d. Moura, S. Kong, J. Avigad, F. v. Doorn, and J. v. Raumer. The lean theorem prover (system description). In *International Conference on Automated Deduction*, pages 378–388. Springer, 2015.
- [81] V. V. Myasnikov. Description of images using a configuration equivalence relation. *Computer Optics*, 42(6):998–1007, 2018.
- [82] J. Myhill. Creative sets. *Journal of Symbolic Logic*, 22(1), 1957.
- [83] R. Nederpelt and H. Geuvers. *Type theory and formal proof: an introduction*. Cambridge University Press, 2014.
- [84] J. Neggers and H. S. Kim. *Basic posets*. World Scientific, 1998.
- [85] A. Nerode. Linear automaton transformations. *Proceedings of the American Mathematical Society*, 9(4):541–544, 1958.
- [86] G. O’Regan. *Guide to discrete mathematics*. Springer, 2021.
- [87] R. E. B. Paiva. *Uma extensão de overlaps e naBL-Álgebras para reticulados*. PhD thesis, Programa de Pós-graduação em Sistemas e Computação, Universidade Federal do Rio Grande do Norte, UFRN, Natal, RN, 2019.
- [88] B. C. Pierce, C. Casinghino, M. Gaboardi, M. Greenberg, C. Hrițcu, V. Sjöberg, and B. Yorgey. *Matemática Fundacional para Computação*. <https://softwarefoundations.cis.upenn.edu/>, University of Pennsylvania, 2007.
- [89] A. J. Pinheiro. *On algebras for interval-valued fuzzy logic*. PhD thesis, Programa de Pós-graduação em Sistemas e Computação, Universidade Federal do Rio Grande do Norte, UFRN, Natal, RN, 2019.
- [90] R. Pressman and B. Maxim. *Engenharia de Software*. McGraw Hill Brasil, 8ª edição edition, 2016.
- [91] M. O. Rabin. Probabilistic automata. *Information and Control*, 6(3):230–245, 1963.

- [92] M. O. Rabin and D. Scott. Finite automata and their decision problems. *IBM Journal of Research and Development*, 3(2):114–125, 1959.
- [93] H. G. Rice. Classes of recursively enumerable sets and their decision problems. *Transactions of the American Mathematical Society*, 74(2):358–366, 1953.
- [94] D. A. Rodrigues. *Sobre a Lógica da Verdade Pragmática em Cálculo de Sequentes*. PhD thesis, Universidade Estadual Paulista, São Paulo, Brasil, 2021.
- [95] M. Sato, T. Sakurai, Y. Kameyama, and A. Igarashi. Calculi of Meta-variables. In *International Workshop on Computer Science Logic*, pages 484–497. Springer, 2003.
- [96] E. R. Scheinerman. *Matemática Discreta - Uma Introdução*. Cengage Learning Editores, terceira edição edition, 2019.
- [97] D. S. Scott and C. Strachey. *Toward a mathematical semantics for computer languages*, volume 1. Oxford University Computing Laboratory, Programming Research Group Oxford, 1971.
- [98] I. Sergey. Programs and proofs: Mechanizing mathematics with dependent types. *Lecture notes with exercises*. Available at, 2014.
- [99] H. G. d. Silva. *A Lógica da Verdade Pragmática em um Sistema de Tableaux*. PhD thesis, Universidade Estadual Paulista, São Paulo, Brasil, 2018.
- [100] M. Sipser. *Introdução à Teoria da Computação*. Cengage Learning Edições Ltda., 2010.
- [101] I. Sommerville. *Software Engineering*. Pearson, 9ª edição edition, 2011.
- [102] M. E. Szabo et al. *The Collected Papers of Gerhard Gentzen*, volume 74. North-Holland Amsterdam, 1969.
- [103] J. L. Szwarcfiter and L. Markenzon. *Estruturas de Dados e seus Algoritmos*, volume 2. Livros Tecnicos e Cientificos, 1994.
- [104] A. Tarski. *Logic, Semantics, Metamathematics: Papers From 1923 To 1938*. Hackett Publishing, 1983.
- [105] S. Thompson. *Type theory and functional programming*. Addison Wesley, 1999.
- [106] T. Tsouanas. *Matemática Fundacional para Computação*. <http://www.tsouanas.org/fmcbook>, Universidade Federal do Rio Grande do Norte, 2017–2021. Work in progress.
- [107] A. M. Turing. On Computable Numbers, with an Application to the Entscheidungsproblem. *Proceedings of the London mathematical society*, 2(1):230–265, 1937.

- [108] D. J. Velleman. *How to prove it: A structured approach*. Cambridge University Press, 2019.