Notas de Aula de Matemática Discreta

Prof. Dr. Rodrigo Geraldo Ribeiro

12 de Abril de 2016

Conteúdo

Prefácio

Esta apostila consiste em notas de aulas de Matemática Discreta para os cursos de Engenharia de Computação e Sistemas de Informação da Universidade Federal de Ouro Preto. Este material foi desenvolvido a partir de diversas fontes bibliográficas, as quais cito abaixo:

1. Discrete Mathematics and its Applications, Rosen [6].
2. How to Prove It: A Structured Approach, Velleman [9],
3. Matemática Concreta, Knuth et. al. [2].
4. Logic and Structure, Van Dalen [8].
5. Notas de aulas do professor prof. Dr. Newton Vieira, DCC- UFMG.

Como grande parte da bibliografia utilizada pela disciplina encontra-se em língua inglesa e "espalhada" por vários livros, o principal objetivo desta apostila é fornecer um fonte bibliográfica unificada em língua portuguesa.

Vários alunos colaboraram com a elaboração deste material, seja por fazerem parte de projetos pró-ativa, monitoria ou sugerindo correções. Listo o nome de alguns: Deivisson Felipe, Pâmela Monique, Arthur Miranda, Patrick Dantas, Rafael Duarte, Marcelo Melo, Carla Neres, Natalie Bravo, Arthur Felipe, Guilherme Baumgratz, Raquel Conceição e possivelmente muitos outros cujos nomes não me recordo. A todos estes alunos (mencionados ou não), dedico um sincero obrigado.

Todo e qualquer erro encontrado neste material é de responsabilidade (e culpa) do autor.

Parte I

Lógica Formal

1

Conceitos Preliminares

> "Comece pelo começo", disse o Rei de maneira severa, "e continue até chegar ao fim: então, pare."
>
> Lewis Carroll, Alice no País das Maravilhas.

Lógica é um *sistema formal* para descrever e raciocinar sobre o mundo. Para isso, precisamos de uma *linguagem formal* e de um conjunto de *regras de inferência* ou regras de dedução, que possibilitem inferir conclusões, a partir de um dado conjunto de *hipóteses*, ou premissas. O objetivo deste primeiro capítulo é esclarecer o significado desses conceitos, que serão fundamentais ao longo de todo o texto.

1.1 Linguagens Formais

Uma *linguagem formal* é um conjunto (possivelmente infinito) de sentenças, definidas sobre um dado *alfabeto*, ou conjunto finito de símbolos. Cada palavra da linguagem é simplesmente uma sequência finita de símbolos do alfabeto. As sentenças de uma linguagem formal são também chamadas de *termos*, ou *fórmulas*, da linguagem.

A *sintaxe* de uma linguagem formal é especificada por um conjunto finito de regras bem definidas, que determinam a estrutura de suas sentenças. Para compreender o significado dessas sentenças, é necessário definir também a *semântica* da linguagem, que é usualmente especificada com base nas regras de definição da sintaxe da linguagem.

Nas seções a seguir apresentamos alguns exemplos de definição da sintaxe e semântica de algumas linguagens simples, com o objetivo de esclarecer melhor esses conceitos.

1.1.1 Sintaxe

Uma linguagem finita pode ser especificada simplesmente pela enumeração de todas as suas sentenças. Entretanto, isso não é muito prático, se a linguagem tem um grande número de sentenças, e certamente não é possível, no caso de

linguagens infinitas. A sintaxe de uma linguagem formal é, portanto, usualmente definida na forma de um conjunto finito de regras, que especificam como podem ser construídas as sentenças dessa linguagem.

Definição 1 (**Valores Booleanos**)**.** O conjunto de valores lógicos, ou conjunto Booleano, contém apenas dois elementos – $\{T, F\}$ – que representam, respectivamente, verdade e falsidade. O nome *Booleano* é uma homenagem ao matemático e lógico inglês George Boole (1815–1864), quem primeiro desenvolveu a Lógica Booleana, que constitui a base para os circuitos dos atuais computadores digitais. O conjunto $\mathcal{B}$, dos termos que denotam valores booleanos, pode ser definido pelas seguintes regras:

$$T \in \mathcal{B}$$
$$F \in \mathcal{B}$$

■

Como o conjunto dos valores Booleanos é finito, sua definição por meio de regras é imediata: basta enumerar uma regra que especifique a pertinência de cada um dos seus elementos.

Definição 2 (**Números Naturais**)**.** O conjunto $\mathcal{N}$, dos termos que representam números naturais, pode ser definido pelas seguintes regras:

$$zero \in \mathcal{N}$$
$$\text{se } n \in \mathcal{N} \text{ então } suc\, n \in \mathcal{N}$$

■

As regras acima constituem uma definição *indutiva*, ou *recursiva*, do conjunto $\mathcal{N}$, no sentido de que alguns elementos do conjunto $\mathcal{N}$ são definidos em termos de elementos do próprio conjunto $\mathcal{N}$. A primeira regra especifica que o termo *zero* pertence ao conjunto $\mathcal{N}$. A segunda regra especifica como são construídos elementos de $\mathcal{N}$, a partir de elementos do próprio conjunto $\mathcal{N}$: se n é um termo pertencente ao conjunto $\mathcal{N}$, então o termo $suc\, n$ também pertence a $\mathcal{N}$. Desse modo, os termos do conjunto $\mathcal{N}$ são:

$$\{zero,\ suc\, zero,\ suc\, (suc\, zero),\ suc\, (suc\, (suc\, zero)),\ ...\}$$

Definição 3 (**Listas**)**.** O conjunto $List\, \mathcal{T}$, dos termos que representam listas de elementos de um dado conjunto $\mathcal{T}$, pode ser definido, indutivamente, pelas seguintes regras:

$$[\,] \in List\, \mathcal{T}$$
$$\text{se } t \in \mathcal{T} \text{ e } ts \in List\, \mathcal{T} \text{ então } t :: ts \in List\, \mathcal{T}$$

■

A primeira regra especifica que o termo $[\,]$ pertence ao conjunto $List\, \mathcal{T}$. O termo $[\,]$ denota a lista vazia, ou seja, que não possui nenhum elemento. A segunda regra especifica como formar termos que representam listas não vazias: dada uma lista $ts \in List\, \mathcal{T}$ e um elemento $t \in \mathcal{T}$, o termo $t :: ts$ representa a lista cujo primeiro elemento, ou *cabeça* da lista, é t, e cujo restante, ou *cauda* da lista, é a lista ts.

Note que listas são definidas de maneira independente do conjunto dos termos que constituem seus elementos – dizemos que a definição de listas é *polimórfica* em relação ao conjunto de seus elementos. De acordo com a definição acima, temos, por exemplo, que as seguintes listas pertencem ao conjunto $List\, \mathcal{B}$:

- $[\,]$, que representa a lista vazia.
- $T :: [\,]$, que representa a lista contendo apenas o elemento T, isto é, a lista cuja cabeça é T e cuja cauda é a lista $[\,]$. Esta lista é também representada, abreviadamente, como $[T]$.
- $F :: T :: [\,]$, representada abreviadamente como $[F, T]$, é a lista cuja cabeça é F e cuja cauda é a lista $T :: [\,]$ (ou seja, $[T]$).

Como elemento do conjunto $\mathit{List}\,\mathcal{N}$ temos, por exemplo, a lista $\mathit{zero} :: (\mathit{suc}\ \mathit{zero}) :: [\,]$, ou seja, $[\mathit{zero} :: (\mathit{suc}\ \mathit{zero})]$.

Definição 4 (Expressões Aritméticas). A linguagem $\mathcal{E}$ é o conjunto dos termos que representam expressões ariméticas sobre números naturais, envolvendo apenas os operadores de adição e multiplicação, representados, respectivamente, pelos termos *plus* e *times*. A linguagem $\mathcal{E}$ pode ser definida, indutivamente, pelas seguintes regras:

$$\begin{array}{l}\text{se } n \in \mathcal{N}\text{, então } \mathit{const}\, n \in \mathcal{E}\\ \text{se } e_1 \in \mathcal{E} \text{ e } e_2 \in \mathcal{E} \text{ então } \mathit{plus}\, e_1\, e_2 \in \mathcal{E}\\ \text{se } e_1 \in \mathcal{E} \text{ e } e_2 \in \mathcal{E} \text{ então } \mathit{times}\, e_1\, e_2 \in \mathcal{E}\end{array}$$

Informalmente, as regras acima nos dizem que todo elemento de $\mathcal{N}$ constitui uma constante em $\mathcal{E}$, e que os operadores *plus* e *times* podem ser usados para construir termos que representam expressões aritméticas mais complexas, a partir de quaisquer dois termos $e_1, e_2 \in \mathcal{E}$. ■

Exemplo 1. Os exemplos a seguir ilustram termos da linguagem $\mathcal{E}$, mostrando a interpretação intuitiva de cada termo. Uma semântica para essa linguagem será definida formalmente na seção 1.1.2.

- $\mathit{const}(\mathit{suc}\ \mathit{zero})$ é um termo que representa número natural 1.
- $\mathit{plus}\ (\mathit{const}(\mathit{suc}\ \mathit{zero}))\ (\mathit{const}(\mathit{suc}\ \mathit{zero}))$ é um termo que representa a expressão $1 + 1$.

■

As definições anteriores especificam formalmente a sintaxe de quatro linguagens: $\mathcal{B}$, $\mathcal{N}$, $\mathit{List}\,\mathcal{T}$ e $\mathcal{E}$. Entretanto, o significado dos termos dessas linguagens apenas foi apresentado de maneira informal. Na seção a seguir, mostramos como atribuir significado aos termos de uma linguagem formalmente.

1.1.2 Semântica

A sintaxe de uma linguagem apenas define o conjunto de termos da linguagem, especificando a estrutura desses termos, mas não atribui a eles qualquer significado. Para isso, é preciso também definir a *semântica* da linguagem.

A semântica de uma linguagem L pode ser definida na forma de uma função, cujo domínio é L e cujo contra-domínio é alguma estrutura matemática S apropriada.[1]. Tal função é usualmente representada na forma $[\![\,]\!] : L \to S$. Essa maneira de definir semântica é ilustrada a seguir, por meio de exemplos que provêm definições semânticas para as linguagens $\mathcal{B}$, $\mathcal{N}$, $\mathcal{E}$ e $\mathit{List}\,\mathcal{T}$, cujas sintaxes foram definidas na seção 1.1.1.

[1] Assumimos que o leitor tem familiaridade com os conceitos de domínio e contra-domínio de uma função, que serão, entretanto, revisados no capítulo 8.

Definição 5 (**Valores Booleanos**)**.** Uma possível semântica para termos do conjunto $\mathcal{B}$ é dada pela função $[\![\,]\!] : \mathcal{B} \to \{0, 1\}$, definida como:

$$\begin{array}{rcl} [\![T]\!] & = & 1 \\ [\![F]\!] & = & 0 \end{array}$$

■

Essa não é a única possível maneira de interpretar os termos de $\mathcal{B}$. Outra possível interpretação é dada pela função $[\![\,]\!] : \mathcal{B} \to \mathbb{Z}$, definida a seguir:

$$\begin{array}{rcl} [\![T]\!] & = & \{k \in \mathbb{Z} \,|\, k \neq 0\} \\ [\![F]\!] & = & \{0\} \end{array}$$

Essa interpretação associa, ao termo T, o conjunto de todos números inteiros diferentes de 0; e associa o valor 0 ao termo F. Essa interpretação é tal como usado nas linguagens de programação C/C++.

Definição 6 (**Números Naturais**)**.** A linguagem $\mathcal{N}$ pode ser interpretada, de modo natural, sobre o conjunto dos números naturais $\mathbb{N}$: o número 0 é associado ao termo *zero*, e o número k ao termo que contém k ocorrências da constante *suc*. Essa semântica é especificada pela função $[\![\,]\!] : \mathcal{N} \to \mathbb{N}$, a seguir. Note que essa função é definida recursivamente, sobre a estrutura da sintaxe de $\mathcal{N}$.

$$\begin{array}{rcl} [\![zero]\!] & = & 0 \\ [\![suc\ n]\!] & = & [\![n]\!] + 1 \end{array}$$

■

Para que uma definição recursiva determine uma função $f : D \to C$, essa definição deve satisfazer aos dois seguintes critérios:

1. *Totalidade*: Isso significa que a definição recursiva deve associar, a cada elemento do domínio D, algum elemento do contradomínio C. Uma maneira de garantir totalidade de uma definição recursiva da semântica de uma linguagem L é especificar uma equação de definição para cada regra de construção de termos da sintaxe de L.

2. *Terminação*: Para garantir terminação, a definição recursiva deve incluir *casos base*, ou seja, casos em que a definição é dada diretamente, e não em termos de si própria. Além disso, em todos os demais casos – ditos *recursivos*, ou *indutivos* – a definição é especificada em termos valores que se aproximam cada vez mais dos casos base. No caso da definição recursiva de uma semântica de uma linguagem L, a terminação pode ser garantida se a semântica de qualquer termo de L é definida apenas em função da semântica de seus subtermos.

A definição 6 determina, de fato, a função semântica $[\![\,]\!] : \mathcal{N} \to \mathbb{N}$: 1) satisfaz ao critério de totalidade, uma vez que inclui uma equação de definição da semântica para cada regra da sintaxe de $\mathcal{N}$; 2) satisfaz o critétio de terminação, pois inclui um caso base, que corresponde à definição do significado do termo *zero*, e, no outro caso, define o significado de um termo com k ocorrências da constante *suc*, em termos do significado do subtermo com $(k - 1)$ ocorrências de *suc*, ou seja, uma ocorrência a menos de *suc* do que no termo original.

A linguagem $List\,\mathcal{T}$ pode também ser interpretada sobre alguma estrutura matemática apropriada. Uma possível definição da semântica de $List\,\mathcal{T}$ será apresentada mais adiante. Por enquanto, vamos tomar como base nossa interpretação informal dos termos dessa linguagem como listas. Vamos mostrar, a seguir, por meio de alguns exemplos, como podemos definir funções sobre listas representadas por esses termos. A definição de cada uma dessas funções é baseada na estrutura sintática dos termos de $List\,\mathcal{T}$.

Definição 7 (**Comprimento de uma lista**)**.** A função $length : List\,\mathcal{T} \to \mathbb{N}$, recebe como argumento uma lista, representada como um termo de $List\,\mathcal{T}$, e retorna o número de elementos, ou seja, o comprimento, da lista dada. Essa função pode ser definida recursivamente, sobre a estrutura sintática dos termos de $List\,\mathcal{T}$, do seguinte modo:

$$\begin{array}{lcll} length\ [\,] & = & 0 & (1) \\ length\ (t :: ts) & = & 1 + length\ ts & (2) \end{array}$$

A definição acima determina a função $length : List\,\mathcal{T} \to \mathbb{N}$, pois: 1) a definição é total, já que existe uma equação para cada regra da definição da sintaxe de $List\,\mathcal{T}$; e 2) termina sempre, pois inclui um caso base (a definição de *length* para a lista vazia) e, no caso recursivo, a definição para uma lista da forma $(t :: ts)$ é dada apenas em termos da sublista ts, que possui um elemento a menos do que a lista original $(t :: ts)$. ■

Exemplo 2. A definição acima pode ser usada para calcular o número de elementos de uma lista, conforme mostra, passo a passo, o exemplo a seguir, que ilustra o cálculo de $length\ (T :: (F :: (T :: [\,])))$:

$$\begin{array}{lll} & length\ (T :: (F :: (T :: [\,]))) & \\ = & 1 + length\ (F :: (T :: [\,])) & \{\text{pela equação (2)}\} \\ = & 1 + (1 + length\ (T :: [\,])) & \{\text{pela equação (2)}\} \\ = & 1 + (1 + (1 + length\ [\,])) & \{\text{pela equação (2)}\} \\ = & 1 + (1 + (1 + 0)) & \{\text{pela equação (1)}\} \\ = & 3 & \end{array}$$

■

Cada passo do cálculo acima simplesmente reescreve a expressão, de acordo com as equações de definição de *length*. Por exemplo, a expressão $length\ (T :: [\,])$ pode ser reescrita como $1 + length\,[\,]$, de acordo com a equação (2).

Definição 8 (**Concatenação de listas**)**.** A concatenação de duas listas xs e ys, denotada como $xs +\!\!+\ ys$, resulta em uma nova lista, que consiste da segunda lista justaposta ao final da primeira. Por exemplo, $(T :: F :: [\,]) +\!\!+ (F :: F :: [\,])$ resulta na lista $(T :: F :: F :: F :: [\,])$. A operação de concatenação de listas pode ser definida, recursivamente, do seguinte modo:

$$\begin{array}{lcll} [\,] +\!\!+\ ys & = & ys & (1) \\ (x :: xs) +\!\!+\ ys & = & x :: (xs +\!\!+\ ys) & (2) \end{array}$$

Note que a recursão é definida sobre a estrutura sintática da lista dada como primeiro parâmetro. A equação (1) especifica que, se o primeiro parâmetro é a lista vazia, então o resultado da concatenação é a segunda lista. A equação

(2) apenas se aplica se o primeiro parâmetro é uma lista não vazia, ou seja, é da forma $(x :: xs)$. Neste caso, o resultado é obtido inserindo-se o primeiro elemento x, da lista $(x :: xs)$, no início da lista resultante de se concatenar a cauda xs, da primeira lista, com a segunda lista ys. ■

Exemplo 3. O cálculo do valor de $(T :: F :: [\,]) +\!+ (F :: F :: [\,])$ é ilustrado, passo a passo, a seguir:

$$\begin{array}{lll} & (T :: F :: [\,]) \mathbin{+\!+} (F :: F :: [\,]) & \\ = & T :: ((F :: [\,]) \mathbin{+\!+} (F :: F :: [\,])) & \text{pela equação (2)} \\ = & T :: (F :: ([\,] \mathbin{+\!+} (F :: F :: [\,]))) & \text{pela equação (2)} \\ = & T :: (F :: (F :: F :: [\,])) & \text{pela equação (1)} \end{array}$$

■

Definição 9 (Expressões aritméticas). Uma maneira natural de interpretar a linguagem $\mathcal{E}$ é ver cada termo como uma expressão arimética sobre números naturais, envolvendo os operadores de adição $(+)$ e multiplicação $(\times)$. Com essa interpretação em mente, vamos definir uma função $eval : \mathcal{E} \to \mathbb{N}$, que mapeia cada termo de $\mathcal{E}$ no valor resultante da avaliação da expressão aritmética correspondente. Essa função pode ser definida recursivamente, sobre a estrutura sintática dos termos de $\mathcal{E}$, tal como mostrado a seguir. A definição usa da função $[\![\,]\!] : \mathcal{N} \to \mathbb{N}$, definida anteriormente.

$$\begin{array}{lcl} eval\,(const\,n) & = & [\![n]\!] \\ eval\,(plus\,e_1\,e_2) & = & eval\,e_1 + eval\,e_2 \\ eval\,(times\,e_1\,e_2) & = & eval\,e_1 \times eval\,e_2 \end{array}$$

■

A função *eval* determina o valor da expressão aritmética representada por um termo da linguagem $\mathcal{E}$, conforme ilustrado pelo exemplo a seguir:

$$\begin{array}{ll} & eval\,(times\,(plus\,(const(suc(suc\,zero)))\,(const(suc\,zero)))\,(const(suc(suc\,zero)))) \\ = & eval\,(plus\,(const(suc(suc\,zero)))(const(suc\,zero))) \times eval\,(const(suc(suc\,zero))) \\ = & (eval\,(const(suc(suc\,zero))) + eval\,(const(suc\,zero))) \times eval\,(const(suc(suc\,zero))) \\ = & ([\![suc(suc\,zero)]\!] + [\![suc\,zero]\!]) \times [\![suc(suc\,zero)]\!] \\ = & (2+1) \times 2 \\ = & 6 \end{array}$$

Todas as definições apresentadas nesta seção podem ser implementadas em uma linguagem de programação de uso geral. Em linguagens funcionais, a implementação seria praticamente uma transcrição das definições apresentadas.

1.1.3 Exercícios

1. Apresente a execução passo a passo das seguintes expressões:
   (a) $[\![suc\,(suc\,(suc\;zero))]\!]$
   (b) $length\,(zero :: zero :: [\,])$
   (c) $(zero :: (suc\;zero) :: [\,]) \mathbin{+\!+} ((suc\;zero) :: zero :: [\,])$
2. A operação de concatenação de duas listas foi definida recursivamente na Definição 8. Justifique que essa definição é uma função, mostrando que ela satisfaz os critérios de totalidade e terminação.

3. Dê uma definição recursiva para a função $nsubtermos : \mathcal{E} \to \mathbb{N}$ que, dada uma expressão aritmética, representada como um termo de $\mathcal{E}$, retorna o número de subtermos dessa expressão. Por exemplo,

   $nsubtermos\ (const\ zero)$ retorna 1
   $nsubtermos\ (times\ (plus\ (const(suc\ zero))\ (const(zero)))\ (const zero))$ retorna 5

   Justifique que sua definição é uma função, mostrando que ela satisfaz os critérios de totalidade e terminação.

4. O objetivo deste exercício é que você compreenda porque definições recursivas que não atendem os critérios de totalidade e terminação não podem ser consideradas funções. Apresente exemplos de definições recursivas que: 1) não atendem o critério de totalidade e 2) não atendem o critério de terminação. Por que essas definições não podem ser funções?

5. Considere o conjunto de expressões aritméticas (Definição 4). Apresente uma funções recursivas para desempenhar as seguintes tarefas:

   (a) Calcular o número de constantes presentes em uma expressão.

   (b) Calcular o número de operações de soma em uma expressão.

6. Considerando o conjunto de números naturais na notação de Peano (Definição 2), desenvolva as seguintes operações:

   (a) Desenvolva uma função para realizar a subtração de dois números naturais. Considere que se $n \leq m$, $n - m = zero$.

   (b) Desenvolva uma função para realizar a multiplicação de dois números naturais. *Dica*: Use a função de adição.

   (c) Desenvolva uma função para realizar a exponenciação de dois números naturais.

   (d) Desenvolva a função $even : \mathcal{N} \to \mathcal{B}$, que retorna T caso o número natural na notação de Peano seja par e F caso contrário.

   (e) Desenvolva uma função $eq : \mathcal{N} \to \mathcal{N} \to \mathcal{B}$, que a partir de dois números naturais na notação de Peano, retorne T se estes forem iguais e F caso contrário.

   (f) Desenvolva uma função $leq : \mathcal{N} \to \mathcal{N} \to \mathcal{B}$, que a partir de dois números naturais na notação de Peano, retorne T se o primeiro for menor ou igual ao segundo e F, caso contrário.

7. Anteriormente, apresentamos uma definição recursiva para o conjunto dos números naturais, $\mathcal{N}$. Outra maneira de representar o conjunto $\mathbb{N}$ é utilizando sequências finitas (listas) de bits. Tal representação consiste na conversão de números de base 10 para base 2 e vice-versa.

   (a) Apresente uma função recursiva que, a partir de um número natural $n \in \mathbb{N}$, retorne uma lista de booleanos correspondente a sequência de bits que representa n em notação binária.

   (b) Apresente uma função recursiva que, a partir de uma lista de valores booleanos $l \in List\mathcal{B}$, retorne o número natural $n \in \mathbb{N}$ correspondente ao número binário l na base 10.

1.2 Notas Bibliográficas

Conceitos de sintaxe e semântica são pervasivos em Ciência da Computação, e são abordados mais detalhadamente em livros sobre semântica formal de linguagens de programação [10], teoria de linguagens formais [7, 4] e construção de compiladores [1].

2

Lógica Proposicional

> "Uma vez uma pessoa me disse: Me convença de que a lógica é útil. — Você deseja que eu prove isso?, respondi. — Sim, ele respondeu. — Então, eu devo produzir um argumento que comprove este fato? — Ele concordou — Então, como você saberá que eu não produzi um argumento falacioso? — Ele nada disse — Veja, você acaba de se convencer de que a lógica é necessária, uma vez que sem ela você não é capaz de saber se esta é ou não necessária."
>
> Epicteto, Discursos.

2.1 Motivação

A lógica provê um ferramental para o raciocínio sobre matemática, algoritmos e circuitos digitais. Sua aplicabilidade em computação permeia diversas áreas, entre elas:

- **Engenharia de Software**: considera-se uma boa prática especificar um sistema antes de iniciar a sua codificação. Várias técnicas de especificação de software são baseadas em asserções lógicas.
- **Aplicações de Missão Crítica**: dizemos que uma aplicação é crítica se a ela está relacionado algum risco (de vida, de elevados prejuízos financeiros etc.). Como a utilização de testes não é, em geral, suficiente para garantir o funcionamento adequado de um programa, o que se espera é uma prova da sua corretude, isto é, uma demonstração de que ele comporta-se de acordo com sua especificação, em todas as situações possíveis. A lógica é a fundamentação matemática de demonstrações de correção de programas.

- **Recuperação de informação**: em máquinas de busca para Web, utiliza-se lógica para especificar propriedades que classificam uma determinada página como relevante ou não com base em seu conteúdo.

- **Circuitos Digitais e Arquitetura de Computadores:** lógica é a linguagem utilizada para descrever sinais produzidos e recebidos como entrada por componentes eletrônicos. Um problema comum no projeto de circuitos eletrônicos é determinar uma versão equivalente, porém mais eficiente, de um circuito. Técnicas para solução desse problema são baseadas em algoritmos eficientes para o processamento de fórmulas lógicas.

- **Bancos de dados**: um recurso fundamental de qualquer sistema gerenciador de bancos de dados é uma linguagem simples e expressiva para recuperar informações nele armazenadas. Lógica é a chave para a expressividade de linguagens para consultas a bancos de dados.

Além das áreas citadas anteriormente, a lógica é fundamental no estudo e no projeto de linguagens de programação e da teoria de computabilidade.

Neste capítulo, discutiremos as dificuldades presentes na utilização do Português para expressar raciocínio lógico e como contornar essas dificuldades, utilizando lógica formal. Existem diversos tipos de lógicas formais, cada uma com uma aplicação específica. Vamos começar considerando uma lógica bem simples, chamada Lógica Proposicional. Primeiramente, vamos definir a sintaxe da linguagem da lógica proposicional e depois vamos considerar três sistemas matemáticos para raciocínio sobre fórmulas da lógica proposicional: tabelas verdade, dedução natural e álgebra Booleana.

Tabelas verdade definem o significado dos conectivos lógicos e como eles podem ser utilizados para calcular os valores de expressões lógicas e provar que duas proposições são logicamente equivalentes. Como tabelas verdade expressam diretamente o siginificado de propoposições, dizemos que essa é uma abordagem baseada em semântica para lógica. Tabelas verdade são de simples entendimento, porém não são úteis na solução de problemas reais, devido ao seu tamanho.

Dedução Natural é uma formalização de princípios básicos do raciocínio lógico utilizado no cotidiano. A dedução natural provê um conjunto de regras de inferência que especificam exatamente quais fatos podem ser deduzidos a partir de um conjunto de fatos dados, ou hipóteses. Em dedução natural não há a noção de 'valor lógico' de proposições, já que tudo no sistema está encapsulado em suas regras de inferência. Conforme veremos posteriormente, essas regras são baseadas na estrutura das proposições envolvidas – a dedução natural é uma abordagem puramente sintática para a lógica. Diversas técnicas utilizadas em pesquisas na área de linguagens de programação são baseadas em sistemas lógicos que são, de alguma maneira, relacionados à dedução natural.

Álgebra Booleana é uma abordagem para formalização da lógica baseada em um conjunto de equações — as leis da álgebra Booleana — para especificar que certas proposições são equivallente a outras. A álgebra Booleana é uma abordagem axiomática, similar á da álgebra elementar ou da geometria, pois provê um conjunto de leis para manipular proposições. Técnicas algébricas para a lógica são fundamentais para o projeto de circuitos digitais.

2.2 Introdução à Lógica Formal

A lógica formal foi inicialmente concebida na grécia antiga, onde filósofos desejavam ser capazes de analisar argumentos em linguagem natural. Os gregos eram fascinados pela idéia de que alguns argumentos eram sempre verdadeiros e outros sempre falsos. Porém, eles rapidamente perceberam que o raciocínio lógico é difícil de ser analisado usando-se linguagens naturais, como o Grego ou o Português. Isso se deve, principalmente, às *ambiguidades* inerentes às linguagens naturais. Uma das maneiras de evitar essas dificuldades é o uso de variáveis que denominaremos *variáveis proposicionais*.

Suponha que um conhecido lhe diga 'O dia está ensolarado e estou feliz'. Aparentemente essa frase possui interpretação óbvia, mas, ao observá-la com cuidado, percebe-se que o seu significado não é tão evidente. Talvez essa pessoa goste de dias ensolarados e fique contente quando esse fato ocorre. Note que existe uma conexão entre as duas partes da sentença e, neste caso, a palavra 'e' presente na frase 'O dia está ensolarado e estou feliz' significa 'e, portanto'. Porém, essa análise depende de nossa experiência em relacionar o clima com a felicidade das pessoas. Considere agora o seguinte exemplo: 'Gatos são peludos e elefantes pesados'. Essa sentença possui a mesma estrutura do exemplo anterior, mas ninguém irá tentar relacionar o peso de elefantes com a quantidade de pelos de gatos. Neste caso, a palavra 'e' significa 'e, também'. Pode-se perceber que a palavra 'e' possui diversos significados sutis, e escolhemos o significado apropriado usando nosso conhecimento do mundo à nossa volta.

As duas frases simples consideradas como exemplo ilustram as dificuldades de interpretação que podem surgir ao se utilizar uma linguagem natural. As dificuldades de dar um significado preciso a frases em linguagem natural não se restringem apenas a como intepretar a palavra 'e'. O estudo preciso da semântica de sentenças expressas em linguagem natural é objeto de estudo da linguística e da filosofia.

Ao invés de tentar o impossível — expressar, de maneira precisa, raciocínio lógico em linguagem natural — vamos nos ater à estrutura lógica de um argumento, separando-a de todas as conotações que possa ter na língua portuguesa. Faremos isso utilizando **proposições**, que são definidas a seguir.

Definição 10 (Proposição)**.** Definimos por proposição qualquer sentença passível de possuir um dos valores lógicos: verdadeiro ou falso. ■

Exemplo 4. Quais das seguintes sentenças podem ser consideradas proposições?

1. Hoje é segunda-feira.
2. $10 < 7$
3. $x + 1 = 3$
4. Como está você?
5. Ela é muito talentosa
6. Existe vida em outros planetas.

Neste exemplo, temos que a sentença 1 é uma proposição, pois o dia de hoje pode ser ou não segunda-feira, tornando essa frase verdadeira ou falsa. A sentença 2 é uma proposição, pois temos que 10 não é menor do que 7. Logo, o valor lógico dessa sentença é falso. A sentença 3 não é uma proposição, pois seu valor lógico depende do valor atribuído à variável x. Se $x = 2$, temos que a sentença 3 é verdadeira. A mesma sentença é falsa para qualquer outro valor de x. Logo, como não é possível determinar de maneira única o valor lógico da sentença 3, ela não é considerada uma proposição. A sentença 4 não é uma proposição, pois não é possível atribuir um valor verdadeiro ou falso para uma pergunta. A sentença 5 não é uma proposição, pois "ela" não está especificada. Portanto, o fato de "ela" ser talentosa ou não depende de quem é "ela". Logo, essa sentença não é uma proposição. A sentença 6 é uma proposição, pois o fato de existir vida em outros planetas pode ser verdadeiro ou falso. ■

No conceito de proposição estão implícitas duas propriedades fundamentais da lógica clássica:

- O princípio da não contradição: Nenhuma proposição é verdadeira e falsa simultaneamente.
- O princípio do terceiro excluído: Toda proposição é verdadeira ou é falsa.

Não é difícil perceber que existem proposições que são compostas por outras proposições menores. Considere a seguinte frase: "Gatos são peludos e elefantes pesados". Esta é formada por duas proposições distintas, a saber: 1) Gatos são peludos; 2) Elefantes são pesados. Podemos classificar proposições como sendo simples ou compostas, conforme definido a seguir.

Definição 11 (Proposição simples e composta)**.** Dizemos que uma proposição é simples se ela não pode ser decomposta em proposições menores. Por sua vez, uma proposição é composta caso possa ser dividida em duas ou mais proposições. ■

Exemplo 5. Classifique as seguintes proposições como simples ou compostas. Caso a proposição em questão seja composta, identifique as proposições simples que a compõem.

1. Diógenes é carteiro.
2. Joãozinho não conta mentiras.
3. O bandido é francês.
4. Se Cléber ganhar eleição, então os impostos serão reduzidos.
5. O processador é rápido mas a impressora é lenta.
6. Se João correr vai ficar cansado.

A primeira proposição é simples, pois não pode ser dividida em proposições menores. Isto é, não possível decompor a frase em "pedaços" de maneira que estes possam ter valores lógicos verdadeiro ou falso. A proposição 2) é composta, pois possui como componente a proposição "Joãozinho conta mentiras". A proposição 3) é simples. A proposição 4) é composta , sendo formada pelas

Conectivo Lógico	**Expressão em Português**
Conjunção	A e B; A mas B; A também B ; A além disso B
Disjunção	A ou B
Condicional	Se A, então B A implica B A logo, B A só se B A somente se B B segue de A A é uma condição suficiente para B basta A para B B é uma condição necessária para A
Bicondicional	A se e somente se B A é condição necessária e suficiente para B
Negação	não A É falso que A Não é verdade que A

Tabela 2.1: Relacionando palavras do português com conectivos lógicos

seguintes proposições simples: "Cléber ganhou a eleição" e "Os impostos serão reduzidos". A proposição 5) também é composta, e é formada pelas proposições: "O processador é rápido" e "A impressora é lenta". Finalmente, a proposição 6) é também composta, sendo formada por "João corre" e "João fica cansado". ■

Proposições simples podem ser combinadas utilizando-se conectivos. Embora exista uma infinidade de conectivos lógicos possíveis, vamos nos ater aqui apenas aos conectivos usualmente utilizados na lógica matemática. : *negação* (não), *conjunção* (ê), *(*disjunção) (ou), *implicação* (se então) e *bi-implicação* (se e somente se).

Algumas palavras da língua portuguesa são frequentemente utilizadas em proposições para denotar conectivos. A tabela 2.1 apresenta algumas destas palavras e quais conectivos estas representam. Nesta tabela utilizamos as variáveis A e B para denotar proposições quaisquer.

Exemplo 6. Quais são os conectivos presentes nas seguintes proposições compostas?

1. Joãozinho não conta mentiras.
2. Se Cléber ganhar eleição, então os impostos serão reduzidos.
3. O processador é rápido mas a impressora é lenta.
4. Amanhã irei à praça ou ao supermercado.
5. Pagarei todas minhas dívidas se e somente se meu salário sair.

Neste exemplo, temos que o conectivo presente na proposição 1) é a negação e a proposição 2 é formada por um condicional. A proposição 3) é formada pelo conectivo de conjunção. Por sua vez, a proposição 4) é formada pelo conectivo de disjunção e a proposição 5 pelo bicondicional. ■

Apesar da tabela 2.1 ser um guia útil na identificação de conectivos, certamente ela não é exaustiva. Além disso, diversas sentenças da língua portuguesa não podem ser representadas utilizando apenas esses tipos de composição. Usualmente elementos que não possuem uma correspondência direta com a lógica proposicional podem ser "despresados" durante a modelagem em questão. Outro ponto referente a modelagem utilizando lógica proposicional é que o conceito de proposição simples e composta é relativo ao problema a ser representado. Por exemplo, considere a seguinte proposição: *5 não é um número par*. Esta proposição pode ser considerada composta — formada pela negação de *5 é um número par* — ou considerada uma proposição simples, indivisível. A tarefa de determinar a "granularidade" do que deve ser considerado como proposição simples varia de problema para problema. Visando tornar esse tipo de conceito uniforme, nesta apostila adotaremos como convenção que uma proposição simples é uma proposição que não pode ser dividida em proposições menores. Desta maneira, a proposição *5 não é um número par* será considerada uma proposição composta.

2.2.1 Exercícios

1. Para cada uma das sentenças a seguir, apresente as proposições simples que a compõe e os conectivos nela envolvidos.

   (a) João é político, mas é honesto.

   (b) João é honesto, mas seu irmão não é.

   (c) Virão a festa João ou sua irmã, além da mãe.

   (d) A estrela do espetáculo não canta, dança nem representa.

   (e) Sempre que o trem apita, João sai correndo.

   (f) Caso João não perca dinheiro no jogo, ele vai a festa.

   (g) João vai ser multado, a menos que diminua a velocidade ou a rodovia não tenha radar.

   (h) Uma condição suficiente para que um número natural n seja primo é que este seja ímpar.

   (i) João vai ao teatro somente se estiver em cartaz uma comédia.

   (j) Se você for Brasileiro, gosta de futebol a menos que torça para o Tabajara ou Íbis.

   (k) A propina será paga exatamente nas situações em que o deputado votar como instruído por João.

   (l) Roberto estava com ciúmes de Ivone ou não estava de bom humor.

   (m) Se o barômetro descer, então vai chover ou nevar.

   (n) Se houver uma requisição, então ela deverá finalmente ser levada em consideração ou o processo requerido nunca poderá proseguir.

   (o) Se João encontrou Maria ontem, eles tomaram uma xícara de café juntos ou passearam no parque.

   (p) Se os juros subirem, o preço das ações abaixará.

   (q) Se João instalou o aquecimento central, então ele vendeu seu carro ou não pagou a hipoteca.

2.2.2 Formalizando Sentenças

Considere as seguintes proposições compostas:

1. O dia está lindo, embora nublado.
2. O dia está ensolarado e José está feliz.

Ao observarmos estas duas proposições, podemos dizer que estas possuem estrutura equivalente, pois ambas são formadas por duas proposições simples e pelo conectivo de conjunção. Desta forma, podemos representar estas proposições compostas de maneira mais compacta substituindo as proposições simples que as compõe por variáveis. A tabela seguinte apresenta a variável associada a uma determinada proposição simples para as frases anteriores.

Variável	Proposição Simples
A	O dia está lindo
B	O dia está nublado
C	O dia está ensolarado
D	José está feliz

Utilizando a tabela anterior, as sentenças em questão podem ser representadas da seguinte maneira:

1. A e B
2. C e D

Apesar do uso de variáveis ter eliminado grande parte dos detalhes que não são relevantes para estrutura das proposições em questão, ainda utilizamos o português para representar os conectivos lógicos utilizados em proposições compostas. Visando tornar a notação para representação de proposições uniforme, adotaremos os seguintes símbolos para conectivos lógicos, em que A e B denotam proposições quaisquer:

Conectivo	Símbolo
Negação	$\neg A$
Conjunção	$A \wedge B$
Disjunção	$A \vee B$
Condicional	$A \rightarrow B$
Bicondicional	$A \leftrightarrow B$

Tabela 2.2: Notação para conectivos lógicos

Utilizando a notação presente na tabela 2.2, temos que as sentenças anteriores seriam representadas pelas seguintes fórmulas $A \wedge B$ e $C \wedge D$.

2.2.3 Exercícios

1. Escreva cada uma das proposições compostas a seguir utilizando a notação simbólica introduzida nesta seção.

   (a) Se Jane vencer ou perder, irá ficar cansada.

(b) Rosas são vermelhas ou violetas são azuis.

(c) Se elefantes podem subir em árvores, 3 é um número irracional.

(d) É proibido fumar cigarros ou charutos.

(e) Não é verdade que se $\pi > 0$ se e somente se $\pi > 1$.

(f) Se as laranjas são amarelas, então os morangos são vermelhos.

(g) É falso que se Montreal é a capital do Canadá, então a próxima copa será realizada no Brasil.

2. Represente utilizando notação simbólica as proposições do exercício 1 da seção 2.2.1.

2.3 Sintaxe da Lógica Proposicional

Tanto no Português quanto na matemática e nas linguagens de programação, existem regras que determinam quando uma determinada sentença é ou não válida na linguagem em questão. Como exemplo, em linguagens de programação, a expressão "(2+3" é considerada sintaticamente inválida, devido à falta do símbolo ")" no final desta expressão. Em liguagens de programação há a necessidade de verificação sintática, pois estamos interessados no signficado (execução) das sentenças (programas) em questão e, formalmente, não há como atribuir semântica a sentenças sintaticamente incorretas.

Para definir quais sentenças da lógica proposicional são passíveis de atribuirmos um significado preciso, iremos definir o conjunto de *fórmulas bem formadas* da lógica proposicional. Neste texto o termo fórmula (da lógica proposicional) denotará fórmulas bem formadas, a menos que seja explicitamente dito o contrário.

Definição 12 (Fórmulas Bem Formadas)**.** O conjunto $\mathcal{F}$ de fórmulas bem formadas da lógica proposicional é definido recursivamente da seguinte maneira:

1. As constantes lógicas $\top, \bot \in \mathcal{F}$ e denotam verdadeiro e falso respectivamente.
2. Seja $\mathcal{V}$ o conjunto (infinito) de variáveis proposicionais. Então $\mathcal{V} \subseteq \mathcal{F}$.
3. Se $\alpha, \beta \in \mathcal{F}$, então:
   (a) $\neg\alpha \in \mathcal{F}$.
   (b) $\alpha \circ \beta \in \mathcal{F}$, em que $\circ \in \{\wedge, \vee, \rightarrow, \leftrightarrow\}$.
   (c) $(\alpha) \in \mathcal{F}$.

Todos os elementos de $\mathcal{F}$ podem ser construídos pelas regras anteriores. ■

Apresentaremos alguns exemplos de fórmulas da lógica e como estas podem ser construídas utilizando a definição 7.

Exemplo 7. Considere as seguintes fórmulas da lógica proposicional:

1. $\neg(A \vee \top)$
2. $A \rightarrow \neg A$

A fórmula 1) pode ser construída da seguinte maneira: Primeiramente, pelas regras 1 e 2 temos que a variável A e a constante $\top$ são fórmulas da lógica e, portanto, pela regra 3-b temos que $A \vee \top$. Uma vez que $A \vee \top$ é uma fórmula, temos, pela regra 3-a, temos que $\neg(A \vee \top)$.

Por sua vez, a fórmula $A \rightarrow \neg A$ pode ser formada da seguinte forma: Pela regra 2, temos que a variável A é uma fórmula. Pela regra 3-a, temos que $\neg A$ é uma fórmula e, finalmente, por 3-b, temos que $A \rightarrow \neg A$.

Porém, as seguintes expressões não podem ser consideradas fórmulas pois, não podem ser construídas de acordo com a definição : $A \vee \neg B \wedge$ e $A \rightarrow \neg$. A primeira não pode ser considerada uma fórmula pois, pela regra 3-b), o operador $\wedge$ precisa de dois parâmetro. O mesmo problema ocorre com a segunda fórmula, pois de acordo com a regra 3-a), o operador $\neg$ precisa de um parâmetro. ■

Em matemática, é usual o uso de parênteses para impor uma ordem de avaliação sobre expressões. Como exemplo, o resultado da expressão $(2 + 3) \times 5$ é obtido calculando-se primeiro a soma para só então efetuarmos a multiplicação. Em fórmulas da lógica proposicional, parênteses podem ser utilizados para determinar a ordem de avaliação de uma certa expressão. Porém, para permitir uma melhor legibilidade, utilizaremos uma ordem de precedência entre os conectivos para evitar o excesso de parênteses. O conectivo de maior precedência é o de negação ($\neg$). O próximo conectivo de maior precedência é a conjunção ($\wedge$) seguido da disjunção ($\vee$). Finalmente, os dois conectivos de menor precedência são o condicional ($\rightarrow$) e o bicondicional ($\leftrightarrow$), sendo o último o de menor precedência. Usando essa ordem de precedência, temos que a fórmula $A \wedge B \rightarrow C$ deve ser entendida como $(A \wedge B) \rightarrow C$, uma vez que a conjunção possui maior precedência que o condicional ($\rightarrow$).

Outra maneira de evitar o excesso de parênteses é a utilização de critérios de associatividade de operadores. Neste texto vamos considerar que os operadores de conjunção e disjunção associam à esquerda, isto é, temos que $A \wedge B \wedge C \wedge D$ denota a mesma expressão que $((A \wedge B) \wedge C) \wedge D$. Por sua vez, os conectivos condicional e bicondicional associam à direita, logo, temos que $A \rightarrow B \rightarrow C \rightarrow D$ representa a mesma expressão que $A \rightarrow (B \rightarrow (C \rightarrow D))$.

2.3.1 Exercícios

1. Para cada uma dos termos a seguir, use a definição de fórmulas bem formadas (definição 7) para justificar o porquê estes podem ser consideradas fórmulas bem formadas.

   (a) $\neg A \wedge B \rightarrow C$

   (b) $(A \rightarrow B) \wedge \neg(A \vee B \rightarrow C)$

   (c) $A \rightarrow B \rightarrow C \leftrightarrow \bot$

   (d) $A \wedge \neg A \rightarrow B$

   (e) $A \vee B \wedge C$

2. Para cada umas das fórmulas seguintes, acrescente parênteses de maneira que não seja necessário utilizar as regras de precedência entre os conectivos da lógica proposicional.

   (a) $\neg A \wedge B \rightarrow C$

(b) $(A \to B) \land \neg(A \lor B \to C)$

(c) $A \to B \to C \leftrightarrow \bot$

(d) $A \land \neg A \to B$

(e) $A \lor B \land C$

3. Para cada uma das fórmulas seguintes, elimine os parênteses desnecessários.

(a) $((A \lor B) \lor (C \lor D))$

(b) $(A \to (B \to (A \land B)))$

(c) $\neg(A \lor (B \land C))$

(d) $\neg(A \land (B \lor C))$

2.4 Semântica da Lógica Proposicional

As fórmulas da lógica proposicional, descritas na seção 7, apesar de possuírem uma definição sintática, ainda não possuem um significado matematicamente preciso. Nesta seção apresentaremos a semântica de fórmulas da lógica proposicional, que foi inicialmente concebida por Alfred Tarski na primeira metade do século XX [3].

Conforme apresentado no capítulo 1, uma forma de atribuirmos semântica a linguagens formais é definindo uma função cujo domínio é o conjunto de termos da linguagem em questão e cujo contradomínio é um conjunto com significado conhecido formalmente. Para a semântica da lógica proposicional, consideraremos como domínio o conjunto de fórmulas bem formadas, $\mathcal{F}$, e contradomínio o conjunto formado apenas pelos valores verdadeiro e falso, $\{T, F\}$.

Tradicionalmente, a função que descreve a semântica de fórmulas da lógica proposicional é apresentada utilizando tabelas verdade, que descrevem o significado de conectivos em termos dos valores lógicos das subfórmulas que o compõe, ou seja, a semântica deve ser definida de acordo com a estrutura da sintaxe das fórmulas.

As próximas subseções definem o significado de cada um dos componentes da definição de fórmulas bem formadas da lógica proposicional.

2.4.1 Semântica de constantes e variáveis

A semântica das fórmulas $\top \in \mathcal{F}$ e $\bot \in \mathcal{F}$ é dada pelas constantes T e F, respectivamente. Para variáveis, a semântica deve considerar as possibilidades de valores lógicos que podem ser assumidos por esta variável. Para uma variável A qualquer, temos que seu significado pode ser um dos valores: verdadeiro ou falso. Este fato é representado pela tabela verdade seguinte:

A
F
T

2.4.2 Semântica da negação ($\neg$)

O significado de uma fórmula $\neg\alpha$, em que α é uma fórmula da lógica proposicional é dada pela seguinte tabela verdade: A primeira linha da tabela verdade

α	$\neg\alpha$
F	T
T	F

da negação diz que se uma fórmula α possui o valor falso (F) então sua negação será o valor T, verdadeiro. De maneira similar, temos que se α possuir o valor falso, $\neg\alpha$ possuirá o valor verdadeiro, conforme especificado na segunda linha da tabela verdade anterior.

2.4.3 Semântica da conjunção ($\wedge$)

Dadas duas fórmulas quaisquer, α, β, temos que $\alpha \wedge \beta$ só possuirá o valor verdadeiro quando tanto α e β forem verdadeiros. Esta interpretação para a conjunção é dada pela tabela a seguir: Note que a tabela verdade para a conjunção

α	β	$\alpha \wedge \beta$
F	F	F
F	T	F
T	F	F
T	T	T

é formada por quatro linhas que correspondem às maneira de atribuir valores verdadeiro e falso para as subfórmulas α e β.

Além disso, a tabela verdade reflete o significado informal da conjunção, a saber: 1) basta que α ou β seja falso para que $\alpha \wedge \beta$ seja falso; e 2) $\alpha \wedge \beta$ será verdadeiro apenas quando α e β também forem verdadeiros simultanemante.

2.4.4 Semântica da disjunção ($\vee$)

A fórmula $\alpha \vee \beta$ será verdadeira quando uma ou ambas das fórmulas α ou β forem verdadeiras. Disso segue que a única maneira de $\alpha \vee \beta$ possuírem o valor falso é quando tanto α quanto β forem falsos.

α	β	$\alpha \vee \beta$
F	F	F
F	T	T
T	F	T
T	T	T

2.4.5 Semântica do condicional ($\rightarrow$)

O conectivo condicional, também conhecido como implicação lógica, possui a mais peculiar semântica dentre os conectivos usuais da lógica proposicional. A peculiaridade da definição semântica da implicação lógica decorre do fato de

que este conectivo é utilizado para representar afirmativas do tipo "se-então", mas seu significado difere um pouco da interpretação cotidiana deste tipo de sentença.

Para quaisquer fórmulas α e β, temos que $\alpha \to \beta$ denota "se α então β"; α implica β ou ainda "não é o caso que α é verdadeiro e β falso". Assim, $\alpha \to \beta$ representa que não é possível que α seja verdadeiro sem que β também o seja. Em outras palavras, ou α é falso ou α e β são ambos verdadeiros.

A tabela verdade para implicação segue diretamente da discussão anterior.

α	β	$\alpha \to \beta$
F	F	T
F	T	T
T	F	F
T	T	T

A partir da tabela anterior, podemos concluir os seguintes fatos úteis sobre a implicação:

1. Se α é falso, então $\alpha \to \beta$ é verdadeiro, independente do valor de β.

2. Se β é verdadeiro, então $\alpha \to \beta$ é verdadeiro, independente do valor de α.

3. A única situação em que $\alpha \to \beta$ possui o valor falso acontece quanto α é verdadeiro e β, falso.

2.4.6 Semântica do bicondicional ($\leftrightarrow$)

Escrevemos $\alpha \leftrightarrow \beta$ para representar que α e β são ambas verdadeiras ou ambas falsas. Desta forma, temos que $\alpha \leftrightarrow \beta$ irá possuir o valor falso somente quando o valor lógico de α e β for diferente. Estes fatos são descritos formalmente na tabela verdade deste conectivo que é apresentada a seguir.

α	β	$\alpha \leftrightarrow \beta$
F	F	T
F	T	F
T	F	F
T	T	T

2.4.7 Construindo tabelas verdade para fórmulas

A construção da tabela verdade de uma fórmula α é feita calculando o valor desta de acordo com a tabela de cada um dos conectivos nela presente e os valores lógicos das variáveis nela presentes.

Considera-se uma boa prática, para construir tabelas verdade, adicionar colunas para "resultados intermediários" de uma fórmula. A noção de resultado intermediário de uma fórmula é definida de maneira precisa usando o conceito de subfórmula. Intuitivamente, o conjunto de subfórmulas é formado por todas as fórmulas bem formadas que compões um dado termo α.

Definição 13 (Subfórmula)**.** Dada uma fórmula α, o conjunto de subfórmulas de α, $sub(\alpha)$, é definido recursivamente da seguinte maneira:

$$sub(\alpha) = \begin{cases} \{\alpha\} & \text{se } \alpha = \top \text{ ou } \alpha = \bot \text{ ou } \alpha \text{ é uma variável} \\ \{\neg\beta\} \cup T & \text{se } \alpha = \neg\beta \text{ e } T = sub(\beta) \\ \{\beta \circ \rho\} \cup T \cup T' & \text{se } \alpha = \beta \circ \rho, \circ \in \{\wedge, \vee, \rightarrow, \leftrightarrow\}, \\ & T = sub(\beta) \text{ e } T' = sub(\rho) \end{cases}$$

■

Exemplo 8. De acordo com a definição 13, temos que o conjunto de subfórmulas de $A \wedge B \rightarrow \bot$ é $\{A \wedge B \rightarrow \bot, A \wedge B, \bot, A, B\}$. Evidentemente, temos que as variáveis A, B, a constante $\bot$ e $A \wedge B \rightarrow \bot$ pertecem ao conjunto de subfórmulas de $sub(A \wedge B \rightarrow \bot)$. Como $A \wedge B$ é um subtermo de $sub(A \wedge B \rightarrow \bot)$, temos que $A \wedge B$ também pertence a $sub(A \wedge B \rightarrow \bot)$. ■

Como $sub(\alpha)$ é o conjunto de subfórmulas de α, estas não possuem uma ordem. Para melhorar a leitura de tabelas verdade, podemos ordenar colunas de tabelas verdade de acordo com o tamanho de fórmulas. O conceito de tamanho de fórmulas é definido formalmente pela seguinte função recursiva:

$$tamanho(\alpha) = \begin{cases} 0 & \text{se } \alpha = \bot \text{ ou } \alpha = \top \\ 1 & \text{se } \alpha \text{ é uma variável} \\ n+1 & \text{se } \alpha = \neg\beta \text{ e } n = tamanho(\beta) \\ n+n'+1 & \text{se } \alpha = \beta \circ \rho, \circ \in \{\wedge, \vee, \rightarrow, \leftrightarrow\} \\ & n = tamanho(\beta) \text{ e } n' = tamanho(\rho) \end{cases}$$

Logo, podemos escrever o conjunto $sub(A \wedge B \rightarrow \bot)$ ordenado por tamanho da seguinte maneira: $\{\bot, A, B, A \wedge B, A \wedge B \rightarrow \bot\}$.

A tabela verdade de uma fórmula pode ser construída tomando por colunas cada uma das subórmulas ordenadas de maneira crescente de acordo com o tamanho. Desta forma, temos que a tabela verdade para $A \wedge B \rightarrow \bot$ é:

$\bot$	A	B	$A \wedge B$	$A \wedge B \rightarrow \bot$
F	F	F	F	T
F	F	T	F	T
F	T	F	F	T
F	T	T	T	F

Observe que as linhas da tabela são preenchidas da seguinte maneira: Todas as linhas para a constante $\bot$ são preenchidas com o valor F. As linhas para variáveis são formadas por todas as combinações de valores verdadeiro-falso para estas. Logo, se uma fórmula possuir n variáveis, esta terá uma tabela com 2^n linhas, representando as duas possibilidades (verdadeiro e falso) para cada uma de suas n variáveis.

Além disso, obtemos valores das linhas de acordo com a tabela verdade do conectivo da fórmula da coluna. Para facilitar, o resultado de colunas posteriores pode ser obtido a partir dos resultados de colunas anteriores.

Exemplo 9. Considere a tarefa de construir a tabela verdade para a fórmula $((A \rightarrow B) \wedge \neg B) \rightarrow \neg A$. Para isso, devemos determinar o conjunto de subfórmulas de $((A \rightarrow B) \wedge \neg B) \rightarrow \neg A$ e ordená-lo de acordo com o tamanho. Ao fazer isso, obtemos o seguinte conjunto:

$$\{A, B, \neg A, \neg B, A \rightarrow B, (A \rightarrow B) \wedge \neg B, ((A \rightarrow B) \wedge \neg B) \rightarrow \neg A\}$$

Agora basta construir a tabela verdade:

A	B	$\neg A$	$\neg B$	$A \to B$	$(A \to B) \wedge \neg B$	$((A \to B) \wedge \neg B) \to \neg A$
F	F	T	T	T	T	T
F	T	T	F	T	F	T
T	F	F	T	F	F	T
T	T	F	F	T	F	T

■

Note que a fórmula do exemplo 9 é sempre verdadeira independente do valor lógico atribuído às suas variáveis. Isso nos permite classificar fórmulas de acordo com sua tabela verdade. Esse será o assunto da próxima seção.

2.4.8 Classificando Fórmulas

O objetivo desta seção é descrever uma maneira de classificar fórmulas da lógica proposicional de acordo com o valor de sua tabela verdade.

Definição 14 (Tautologia e Contradição)**.** Dizemos que uma fórmula da lógica é uma tautologia se esta é sempre verdadeira independente do valor lógico de suas variáveis. Por sua vez, uma fórmula é uma contradição se esta é sempre falsa independente do valor de suas variáveis. ■

Exemplo 10. A fórmula $A \wedge B \to A \vee B$ é uma tautologia, pois é sempre verdadeira independente dos valores das variáveis A e B, como mostrado pela tabela verdade seguinte:

A	B	$A \wedge B$	$A \vee B$	$A \wedge B \to A \vee B$
F	F	F	F	T
F	T	F	T	T
T	F	F	T	T
T	T	T	T	T

Como exemplo de uma contradição considere a fórmula $(A \to B) \wedge (A \wedge \neg B)$ e sua tabela verdade:

A	B	$\neg B$	$A \wedge \neg B$	$A \to B$	$(A \to B) \wedge (A \wedge \neg B)$
F	F	T	F	T	F
F	T	F	F	T	F
T	F	T	T	F	F
T	T	F	F	T	F

■

Definição 15 (Fórmula Satisfatível, Falseável e Contigente)**.** Uma fórmula da lógica é dita ser satisfatível se existe uma maneira de atribuir valores lógicos às suas variáveis de maneira a torná-la verdadeira. Uma fórmula é dita ser falseável se existe uma maneira de atribuir valores lógicos às suas variáveis de maneira a torná-la falsa. Finalmente, dizemos que uma fórmula é contingente se esta é satisfatível e falseável simultaneamente. ■

Exemplo 11. Para ilustrar os conceitos de fórmula satisfatível, falseável e contingente, considere a fórmula $A \vee B \rightarrow \neg A$ e sua tabela verdade:

A	B	$\neg A$	$A \vee B$	$A \vee B \rightarrow \neg A$
F	F	T	F	T
F	T	T	T	T
T	F	F	T	F
T	T	F	T	F

Com isso temos que $A \vee B \rightarrow \neg A$ é satisfazível, pois para $A = F$ e $B = F$ temos que esta fórmula é verdadeira. De maneira similar, para $A = T$ e $B = T$ temos que esta fórmula é falsa e, portanto, falseável. Como, $A \vee B \rightarrow \neg A$ é satisfazível e falseável temos que esta pode ser classificada como contingente. ■

Uma das aplicações dos conceitos anteriores é a determinar quando duas fórmulas α, β da lógica proposicional são equivalentes. Dizemos que duas fórmulas são equivalentes se estas possuem o mesmo valor lógico para a mesma atribuição de valores às suas variáveis, isto é se $\alpha \leftrightarrow \beta$ é uma tautologia.

2.4.9 Limitações de tabelas verdade

Tabelas verdade são um método simples para determinar a satisfazibilidade de fórmulas da lógica proposicional, pois estas denotam de maneira direta o significado de conectivos e fórmulas. Porém, a simplicidade de tabelas verdade possui um grande complicador: estas possuem tamanho exponencial sobre o número de variáveis em uma dada fórmula.

Em exemplos anteriores, mostramos tabelas verdade para fórmulas que possuíam duas variáveis. Todas estas tabelas possuíam 4 linhas. Considere a seguinte tabela para uma fórmula com 3 variáveis:

A	B	C	$A \wedge B$	$A \wedge B \wedge C$
F	F	F	F	F
F	F	T	F	F
F	T	F	F	F
F	T	T	F	F
T	F	F	F	F
T	T	F	T	F
T	F	T	F	F
T	T	T	T	T

esta possui 8 linhas. De forma geral, a tabela verdade para fórmulas contendo n variáveis possuirá 2^n linhas, o que limita a utilização de tabelas verdades para soluções de problemas práticos.

2.4.10 Consequência lógica

A noção de consequência lógica é um dos mais importantes conceitos no estudo de lógica. Informalmente, a consequência lógica expressa quando um argumento lógico é considerado válido. Dizemos que argumentos são válidos se sua conclusão é uma consequência de suas premissas (também chamadas de hipóteses), em que tanto a conclusão quanto as premissas são proposições. A próxima definição descreve de maneira precisa o que é uma consequência lógica.

Definição 16 (Consequência Lógica)**.** Dizemos que uma fórmula α é consequência lógica de um conjunto de fórmulas Γ, $\Gamma \models \alpha$, se, e somente se sempre que toda fórmula em Γ for verdadeira, α também o é. Isto é, se

$$\left(\bigwedge_{\varphi \in \Gamma} \varphi \right) \rightarrow \alpha$$

é uma tautologia. ■

Uma maneira para determinarmos se uma fórmula α é consequência lógica de um conjunto Γ é construir uma tabela verdade. O seguinte exemplo ilustra esse uso de tabelas verdade.

Exemplo 12. Considere as seguintes proposições:

- Se hoje for segunda-feira, irei a reunião.
- Hoje é segunda-feira.
- Hoje Irei a reunião.

Note que essas proposições podem ser modeladas pelas seguintes fórmulas:

- $A \rightarrow B$
- A
- B

em que a variável A denota "Hoje é segunda-feira" e B, "Hoje irei a reunião".

De acordo com a interpretação usual da língua portuguesa, temos que "Hoje irei a reunião" é uma consequência de "Se hoje for segunda-feira, irei a reunião" e "hoje é segunda-feira". Mas será que a definição formal de consequência lógica, coincide com a noção usual de consequências dedutivas utilizadas coloquialmente na língua portuguesa?

Considerando as fórmulas $A \rightarrow B$, A e B que modelam estas proposições citadas, temos que se B é uma consequência de $A \rightarrow B$ e A se $[(A \rightarrow B) \wedge A] \rightarrow B$ é uma tautologia, o que pode ser verificado pela tabela verdade abaixo:

A	B	$A \rightarrow B$	$(A \rightarrow B) \wedge A$	$[(A \rightarrow B) \wedge A] \rightarrow B$
F	F	T	F	T
F	T	T	F	T
T	F	F	F	T
T	T	T	T	T

Desta maneira, temos que a fórmula B é uma consequência lógica de $A \rightarrow B$ e A. ■

Evidentemente, utilizar tabelas verdade para determinar consequências lógicas possui o inconveniente de que tabelas verdade são exponenciais no número de variáveis presentes em uma fórmula, logo, mesmo para sentenças envolvendo poucas proposições, o uso de tabelas verdade para verificar a validade de argumentos é impraticável, como já apresentado na seção 2.4.9. Na seção 2.5, apresentaremos o sistema de dedução natural para lógica proposicional que permite verificar consequências lógicas sem a construção de uma tabela verdade.

2.4.11 Exercícios

1. Obtenha o conjunto de subfórmulas de cada fórmula a seguir utilizando a definição 13.
   (a) $P \vee Q \rightarrow Q \vee P$
   (b) $((P \wedge Q) \vee (P \wedge R)) \leftrightarrow (P \wedge (Q \vee R))$
   (c) $(P \rightarrow Q) \wedge P \wedge \neg Q$
   (d) $(P \rightarrow Q) \wedge \neg P \rightarrow Q$
2. Construa tabelas verdade para as fórmulas a seguir e classifique-as como sendo tautologias, contingências ou contradições:
   (a) $(A \rightarrow B) \leftrightarrow \neg A \vee B$
   (b) $(A \wedge B) \vee C \rightarrow A \wedge (B \vee C)$
   (c) $A \wedge \neg(\neg A \vee \neg B)$
   (d) $A \wedge B \rightarrow \neg A$
   (e) $(A \rightarrow B) \rightarrow [(A \vee C) \rightarrow (B \vee C)]$
   (f) $A \rightarrow (B \rightarrow A)$
   (g) $(A \wedge B) \leftrightarrow (\neg B \vee \neg A)$
3. Suponha que você possua um algoritmo que a partir de uma fórmula α da lógica responda sim se esta é satisfazível e não, caso contrário. Explique como usar esse algoritmo para determinar se uma fórmula é uma:
   (a) Tautologia
   (b) Contradição

2.5 Dedução Natural para Lógica Proposicional

Dedução natural é um sistema formal para dedução de consequências lógicas sem a necessidade de substituir variáveis por valores lógicos ou avaliar expressões. O formalismo de dedução natural é intensivamente estudado por cientistas da computação, uma vez que este é o formalismo subjacente a ferramentas para verificação de provas por computador como Coq.

De maneira simples, a dedução natural consiste de um conjunto de regras que permitem estabelecer a validade de argumentos representados como sequentes, que são definidos a seguir.

Definição 17 (Sequente)**.** Sejam $\alpha_1, ..., \alpha_n, \varphi$ fórmulas bem formadas da lógica proposicional. A notação $\alpha_1, ..., \alpha_n \vdash \varphi$ é denominada de sequente e representa que φ pode ser deduzida a partir de $\alpha_1, ..., \alpha_n$ utilizando as regras da dedução natural. ■

Como argumentos são formados por premissas e uma conclusão, temos que no sequente

$$\alpha_1, ..., \alpha_n \vdash \varphi$$

o conjunto formado pelas fórmulas α_i, $1 \leq i \leq n$, são as premissas e φ a conclusão do argumento representado.

Para determinar a validade de argumentos utilizando dedução natural, devemos ser capazes de inferir a conclusão a partir das premissas, utilizando as regras da dedução natural. Regras da dedução natural são expressas escrevendo as premissas acima de uma linha horizontal que as separam da conclusão.

$$\frac{\text{Fórmula}_1, ..., \text{Fórmula}_n}{\text{Conclusão}}$$

Esta notação expressa, intuitivamente, que se formos capazes de determinar a validade de cada uma das fórmulas Formula_i, $1 \leq i \leq n$, então a Conclusão também será verdadeira.

A maioria das regras da dedução natural podem ser divididas em duas categorias. Regras de introdução são aquelas nas quais um novo conectivo é incluído na fórmula da conclusão e são utilizadas para construir expressões mais complexas a partir de outras mais simples. Por sua vez, regras de eliminação possuem como premissa uma fórmula com um certo conectivo e este é removido da conclusão. Estas regras são utilizadas para decompor expressões complexas em expressões mais simples.

Visando simplificar a quantidade de regras para a dedução natural, utilizaremos apenas os conectivos de disjução ($\vee$), conjunção ($\wedge$), implicação ($\rightarrow$) e a constante ($\bot$). Esta convenção não compromete a expressividade da lógica pois os conectivos de negação, bicondicional e a constante $\top$ podem ser definidos da seguinte maneira:

$$\begin{array}{lcl} \neg A & \equiv & A \rightarrow \bot \\ A \leftrightarrow B & \equiv & (A \rightarrow B) \wedge (B \rightarrow A) \\ \top & \equiv & \bot \rightarrow \bot \end{array}$$

É fácil verificar, utilizando tabelas verdade, que as abreviações anteriores são realmente equivalentes. As próximas seções descrevem cada uma das regras da dedução natural apresentando exemplos de sua utilização.

2.5.1 Regra para identidade ($\{_{ID}\}$)

A primeira regra da dedução natural expressa um fato bastante óbvio: se você deseja provar que uma fórmula α é verdadeira e α é uma das fórmulas presentes no conjunto de hipóteses, então você pode concluí-la utilizando a regra $\{_{ID}\}$, que é apresentada a seguir.

$$\frac{\alpha \in \Gamma}{\Gamma \vdash \alpha}\ \{_{ID}\}$$

Porém, a utilização do conjunto de hipóteses Γ, pode ser omitida, para facilitar a leitura das deduções. Usando esta notação simplificada, podemos expressar a regra $\{_{ID}\}$, da seguinte maneira:

$$\frac{}{\alpha}\ \{_{ID}\}$$

Note que na versão "simplificada" da regra, todas as referências ao conjunto de hipóteses Γ foram removidas, porém, só podemos utilizar esta regra se a fórmula α pertencer ao conjunto de hipóteses.

2.5.2 Regras para a conjunção ($\wedge$)

Introdução da conjunção $\{\wedge_I\}$

De maneira simples, a regra de introdução da conjunção ($\{\wedge I\}$), diz que se for possível deduzir uma fórmula α, a partir de um conjunto de hipóteses (premissas) Γ e também for possível deduzir β a partir deste mesmo conjunto de hipóteses Γ, então a partir de Γ é possível inferir $\alpha \wedge \beta$. Isto é expresso de maneira concisa pela seguinte regra:

$$\frac{\Gamma \vdash \alpha \quad \Gamma \vdash \beta}{\Gamma \vdash \alpha \wedge \beta}\ \{\wedge_I\}$$

Como já dito anteriormente, omitimos o conjunto de hipóteses Γ obtendo a seguinte versão simplificada desta regra:

$$\frac{\alpha \quad \beta}{\alpha \wedge \beta}\ \{\wedge_I\}$$

Para uma melhor compreensão de como construir demonstrações utilizando dedução natural utilizando estas regras, considere os seguintes exemplos.

Exemplo 13. Como um primeiro exemplo, considere a tarefa de determinar a validade do seguinte sequente: $P, Q \vdash P \wedge Q$. Temos que neste sequente o conjunto de hipóteses é $\Gamma = \{P, Q\}$ e a conclusão $P \wedge Q$. Este sequente pode ser provado utilizando a regra $\{\wedge_I\}$, conforme a dedução a seguir:

$$\frac{P \quad Q}{P \wedge Q}\ \{\wedge_I\}$$

A mesma dedução deixando o conjunto de hipóteses explícito é apresentada abaixo:

$$\frac{\dfrac{P \in \{P,Q\}}{\{P,Q\} \vdash P}\ \{_{ID}\} \quad \dfrac{Q \in \{P,Q\}}{\{P,Q\} \vdash Q}\ \{_{ID}\}}{\{P,Q\} \vdash P \wedge Q}\ \{\wedge_I\}$$

Note que o uso do conjunto de hipóteses torna as deduções mais verbosas e, portanto, dificulta a leitura. Por isso, vamos utilizar a versão simplificada das regras, exceto em alguns exemplos como este. ■

Exemplo 14. Considere a tarefa de demonstrar a validade do seguinte sequente: P, Q, $R \vdash (P \wedge Q) \wedge R$. Para obter a conclusão a partir das hipóteses, temos que utilizar a regra $\{\wedge_I\}$ duas vezes, uma para deduzir $P \wedge Q$ e outra para deduzir $(P \wedge Q) \wedge R$, conforme apresentado na dedução seguinte:

$$\frac{\dfrac{\overline{P}\ \{_{ID}\} \quad \overline{Q}\ \{_{ID}\}}{P \wedge Q}\ \{\wedge_I\} \quad \overline{R}\ \{_{ID}\}}{(P \wedge Q) \wedge R}\ \{\wedge_I\}$$

■

Eliminação da conjunção $\{\wedge_{EE}\}$, $\{\wedge_{ED}\}$

O conectivo de conjunção ($\wedge$) possui duas regras de eliminação. Estas regras expressam o fato de que se sabemos que $\alpha \wedge \beta$ é verdadeira, então α é verdadeira e β é verdadeiro. A regra de eliminação da conjunção à esquerda ($\{\wedge_{EE}\}$) permite concluir α a partir de $\alpha \wedge \beta$, isto é, mantemos a fórmula à esquerda. Por sua vez, a regra de eliminação à direita permite concluir a fórmula à direita do conectivo $\wedge$. Ambas as regras são apresentadas a seguir.

$$\frac{\alpha \wedge \beta}{\alpha}\ \{\wedge_{EE}\} \qquad \frac{\alpha \wedge \beta}{\beta}\ \{\wedge_{ED}\}$$

Os exemplos seguintes ilustram a utilização destas regras.

Exemplo 15. Utilizando as regras para conjunção podemos provar que $P, Q \wedge R \vdash P \wedge R$ é um sequente válido. Note que para isso, utilizaremos as regras de introdução e eliminação a esquerda para a conjunção, conforme ilustrado a seguir.

$$\frac{\dfrac{}{P}\ \{_{ID}\} \qquad \dfrac{Q \wedge R}{R}\ \{\wedge_{ED}\}}{P \wedge R}\ \{\wedge_{I}\}$$

■

Exemplo 16. Outro sequente que podemos provar utilizando as regras para conjunção é $P \wedge (Q \wedge R) \vdash (P \wedge Q) \wedge R$, cuja dedução é apresentada a seguir:

$$\frac{\dfrac{\dfrac{\dfrac{}{P \wedge (Q \wedge R)}\ \{_{ID}\}}{P}\ \{\wedge_{EE}\} \qquad \dfrac{\dfrac{}{P \wedge (Q \wedge R)}\ \{_{ID}\}}{Q \wedge R}\ \{\wedge_{ED}\}}{Q}\ \{\wedge_{EE}\}}{P \wedge Q} \qquad \dfrac{\dfrac{}{P \wedge (Q \wedge R)}\ \{_{ID}\}}{Q \wedge R}\ \{\wedge_{ED}\}}{R}\ \{\wedge_{ED}\}}{(P \wedge Q) \wedge R}\ \{\wedge_{I}\}$$

Inicialmente, utilizamos a regra $\{\wedge_I\}$ para deduzir $(P \wedge Q) \wedge R$, a partir de $P \wedge Q$ e R. A dedução de $P \wedge Q$ utiliza $\{\wedge_I\}$ e três eliminações da conjunção sobre a hipótese $P \wedge (Q \wedge R)$. Para a dedução de R, utilizamos duas eliminações da conjunção sobre $P \wedge (Q \wedge R)$. ■

2.5.3 Regras para a implicação ($\rightarrow$)

Eliminação da implicação ($\rightarrow E$)

Em nosso cotidiano, provavelmente a regra de dedução que mais utilizamos é a regra de eliminação da implicação, $\{\rightarrow_E\}$. Esta regra afirma que se conseguirmos deduzir que $\alpha \rightarrow \beta$ é verdade e que α é verdade, então, utilizando a regra $\{\rightarrow_E\}$, podemos deduzir que β possui o valor verdadeiro. Esta regra é apresentada a seguir:

$$\frac{\alpha \rightarrow \beta \quad \alpha}{\beta}\ \{\rightarrow_E\}$$

A regra de eliminação da implicação, $\{\rightarrow_E\}$, é também conhecida como *modus ponens*. O próximo exemplo apresenta uma simples aplicação desta regra.

Exemplo 17. O sequente $A \to B, B \to C, A \vdash A \wedge C$ possui a seguinte demonstração:

$$\dfrac{\dfrac{\overline{A}\ \{_{ID}\} \quad \dfrac{\overline{B \to C}\ \{_{ID}\} \quad \dfrac{\overline{A \to B}\ \{_{ID}\} \quad \overline{A}\ \{_{ID}\}}{B}\ \{\to_{E}\}}{C}\ \{\wedge_{I}\}}{A \wedge C}$$

■

Introdução da implicação ($\{\to_{I}\}$)

A regra de introdução da implicação, $\{\to_{I}\}$, especifica que para deduzirmos uma fórmula $\alpha \to \beta$, a partir de um conjunto de hipóteses Γ, devemos obter uma prova de β utilizando α como uma hipótese adicional. Esta regra é apresentada abaixo:

$$\dfrac{\Gamma \cup \{\alpha\} \vdash \beta}{\Gamma \vdash \alpha \to \beta}\ \{\to_{I}\}$$

Note que o efeito de utilizar a regra $\{\to_{I}\}$ é adicionar o lado esquerdo da implicação a ser deduzida como uma hipótese adicional. O próximo exemplo ilustra a utilização desta regra.

Exemplo 18. Considere a tarefa de deduzir que $\vdash A \wedge B \to A$. Note que de acordo com a regra $\{\to_{I}\}$, devemos transformar o sequente $\vdash A \wedge B \to A$, no sequente $A \wedge B \vdash A$. Por sua vez, o sequente $A \wedge B \vdash A$ pode ser deduzido de maneira imediata utilizando $\{\wedge_{EE}\}$. A dedução completa é apresentada a seguir.

$$\dfrac{\dfrac{\dfrac{A \wedge B \in \{A \wedge B\}}{\{A \wedge B\} \vdash A \wedge B}\ \{_{ID}\}}{\{A \wedge B\} \vdash A}\ \{\wedge_{EE}\}}{\vdash A \wedge B \to A}\ \{\to_{I}\}$$

Neste exemplo, pode-se perceber a utilidade do símbolo $\vdash$, tornar explícita a separação das hipóteses e da conclusão de um sequente. Antes de utilizarmos a regra $\{\to_{I}\}$, o conjunto de hipóteses deste sequente era vazio, isto é este sequente não possuía hipóteses. Usar a regra $\{\to_{I}\}$ nos permitiu incluir o lado esquerdo de $A \wedge B \to A$ ($A \wedge B$) no conjunto de hipóteses, possibilitanto assim, o término desta dedução. ■

Note que ao observarmos a dedução do exemplo anterior, esta permite-nos pensar que $A \wedge B$ é uma hipótese deste sequente, visto que aplicamos a regra $\{_{ID}\}$ para deduzí-la. Porém, a fórmula $A \wedge B$ é uma hipótese de "visibilidade local", cujo único propósito é possibilitar a demonstração do sequente $A \wedge B \vdash A$. Assim que obtemos a dedução desejada, a hipótese adicional pode ser "descartada", isto é, eliminada do conjunto de hipóteses do sequente em questão. Em nosso exemplo, a visibilidade da hipótese adicional $A \wedge B$ é toda a dedução acima do uso da regra $\{\to_{I}\}$.

Em deduções maiores, manter, de maneira consistente, quais hipóteses temporárias estão ou não visiíveis em um dado ponto da demonstração pode ser uma tarefa complicada. Uma solução para isso é manter o conjunto de hipóteses em

todo ponto da dedução, mas como já argumentamos diversas vezes neste texto, isso prejudica o entendimento das demonstrações. Visando facilitar a legibilidade das deduções, vamos numerar cada hipótese temporária e indicar com o mesmo número a regra que a introduziu. Isto permitirá definir a visibilidade de uma hipótese como sendo toda a dedução "acima" da regra que a introduziu. Além disso, omitiremos o conjunto de hipóteses da regra de introdução da implicação, escrevendo-a da seguinte forma simplificada:

$$\frac{\alpha \vdash \beta}{\alpha \to \beta}\ \{\to_{I}\}$$

em que a notação "$\alpha \vdash \beta$", denota "deduzir β utilizando α como hipótese adicional". Utilizando a convenção de numeração de hipóteses locais e versão simplificada da regra $\{\to_{I}\}$, a dedução do exemplo anterior, ficaria como:

$$\dfrac{\dfrac{\dfrac{\overline{A \wedge B}^{1}\ \{_{ID}\}}{A}\ \{\wedge_{EE}\}}{A \wedge B \to A}\ \{\to_{I}\}^{1}}$$

Note que a visibilidade de $A \wedge B$ é delimitada pela regra $\{\to_{I}\}$ que foi numerada com o valor 1. Este mesmo valor foi utilizado para marcar a utilização de $A \wedge B$ quando da utilização da regra $\{_{ID}\}$, para explicitar o uso de uma hipótese temporária.

A seguir apresentamos mais dois exemplos para estas regras.

Exemplo 19. Neste exemplo, mostraremos que se sabe-se que $A \to B$ e $B \to C$ são verdadeiras, então $A \to C$ também será verdadeira. Tal fato é expresso pelo seguinte sequente: $\{A \to B, B \to C\} \vdash A \to C$. A dedução é apresentada abaixo:

$$\dfrac{\dfrac{\overline{B \to C}\ \{_{ID}\} \qquad \dfrac{\overline{A \to B}\ \{_{ID}\} \quad \overline{A^{1}}\ \{_{ID}\}}{B}\ \{\to_{E}\}}{C}\ \{\to_{E}\}}{A \to C}\ \{\to_{I}\}^{1}$$

■

O próximo exemplo apresenta um resultado quase imediato utilizando a dedução anterior, este resultado é conhecido em muitos livros de lógica como *modus tollens*.

Exemplo 20. O *modus tollens* especifica que se $A \to B$ e $\neg B$ são fórmulas verdadeiras, então, $\neg A$ também deve ser verdadeira. Note que $\neg B \equiv B \to \bot$. Então, usando o resultado do exemplo anterior, temos que a partir de $A \to B$ e $B \to \bot$ podemos deduzir $A \to \bot$. Evidentemente, podemos deduzir este resultado sem apelar para o exemplo anterior. Deixamos essa dedução como um exercício para o leitor. ■

2.5.4 Regras para a disjunção ($\vee$)

Introdução da disjunção $\{\vee_{IE}\}$, $\{\vee_{ID}\}$

As regras para introdução da disjunção estabelecem condições que devem ser satisfeitas para que possamos deduzir uma fórmula contendo o conectivo $\vee$.

Caso α seja verdadeiro, temos que $\alpha \vee \beta$ e $\beta \vee \alpha$ também devem ser verdadeiros, para qualquer fórmula β. Como basta uma das fórmuas ser verdadeira para que toda a disjunção também o seja, temos duas regras para introduzir o conectivo $\vee$, apresentadas a seguir:

$$\frac{\Gamma \vdash \alpha}{\Gamma \vdash \alpha \vee \beta}\ \{\vee_{IE}\} \quad \frac{\Gamma \vdash \beta}{\Gamma \vdash \alpha \vee \beta}\ \{\vee_{ID}\}$$

Assim como em regras anteriores, omitiremos o conjunto de hipóteses Γ, obtendo as seguintes formas simplificadas das regras anteriores:

$$\frac{\alpha}{\alpha \vee \beta}\ \{\vee_{IE}\} \quad \frac{\beta}{\alpha \vee \beta}\ \{\vee_{ID}\}$$

O próximo exemplo ilustra a utilização destas regras.

Exemplo 21. Considere a tarefa de demonstrar o seguinte sequente: $\{P \wedge Q\} \vdash P \vee Q$. Como a conclusão deste sequente possui o conectivo $\vee$, podemos iniciar sua prova utilizando uma das regras de introdução da disjunção, conforme ilustrado na dedução abaixo:

$$\dfrac{\dfrac{\overline{P \wedge Q}\ \{\wedge_{ID}\}}{Q}\ \{\vee_{ID}\}}{P \vee Q}$$

■

Porém, esta não é a única maneira de se demonstrar esse sequente. Podemos deduzí-lo iniciando com a regra $\{\vee_{IE}\}$, conforme apresentado a seguir:

$$\dfrac{\dfrac{\overline{P \wedge Q}\ \{\wedge_{EE}\}}{P}\ \{\vee_{IE}\}}{P \vee Q}$$

Como existem duas demonstrações para esse sequente, qual destas seria a correta? A resposta é simples: Ambas! O fato de um sequente admitir mais de uma demonstração permite-nos "escolher" entre qualquer uma destas. Isso quer dizer, que podemos considerar diferentes deduções de um sequente como sendo "iguais". Este fato de considerar diferentes deduções de um mesmo sequente como sendo iguais é conhecido como *irrelevância de provas*[1]. Note que, como podemos considerar ambas as provas como sendo equivalentes, não há necessidade de se construir ambas ou de se escolher uma em detrimento da outra.

Eliminação da disjunção $\{\vee_E\}$

A regra de eliminação da disjunção especifica o que pode ser deduzido a partir do fato de que $\alpha \vee \beta$ é verdadeira. Note que, se $\alpha \vee \beta$ é uma fórmula verdadeira, não podemos concluir diretamente que α ou β também devem ser verdadeiras. Isto decorre do significado da disjunção. Se $\alpha \vee \beta$ é verdadeira, temos que α pode ser verdadeira ou β pode ser verdadeira ou ambas[2]!

[1] Do inglês: Proof Irrelevance.

[2] Lembre-se da tabela verdade para a disjunção!

Contudo, se sabemos que $\alpha \vee \beta$ é verdadeira e que uma fórmula γ pode ser inferida a partir de α e também de β, podemos então deduzir que γ deve ser verdadeira. Estas idéias são ilustradas pela regra de eliminação da disjunção, $\{\vee_E\}$, apresentada a seguir:

$$\frac{\Gamma \vdash \alpha \vee \beta \quad \Gamma \cup \{\alpha\} \vdash \gamma \quad \Gamma \cup \{\beta\} \vdash \gamma}{\Gamma \vdash \gamma}\ \{\vee_E\}$$

Note que, assim como a regra $\{\rightarrow_I\}$, a eliminação da disjunção permite a inclusão de novas hipóteses. Novamente, utilizaremos a convenção de numerar as hipóteses temporárias de maneira que sua visibilidade na demonstração fique evidente. Eliminando as ocorrências do conjunto de hipóteses Γ, podemos reescrever a regra $\{\vee_E\}$, da seguinte maneira:

$$\frac{\alpha \vee \beta \quad \alpha \vdash \gamma \quad \beta \vdash \gamma}{\gamma}\ \{\vee_E\}$$

Exemplo 22. Neste exemplo, considere a tarefa de demonstrar o seguinte sequente: $\{A \vee B, A \rightarrow C, B \rightarrow C\} \vdash C$. Para deduzir C, utilizaremos a regra $\{\vee_E\}$ sobre $A \vee B$ para obter hipóteses que possibilitem deduzir C a partir das implicações $A \rightarrow C$ e $B \rightarrow C$. Esta dedução é apresentada abaixo:

$$\dfrac{\dfrac{}{A \vee B}\ \{_{ID}\} \quad \dfrac{\dfrac{\overline{A \rightarrow C}\ \{_{ID}\} \quad A^1}{C}\ \{\rightarrow_E\} \quad \dfrac{\overline{B \rightarrow C}\ \{_{ID}\} \quad B^1}{C}\ \{\rightarrow_E\}}{C}\ \{\vee_E\}^1}{C}$$

■

Exemplo 23. Vamos considerar um dedução utilizando $\{\vee_E\}$ um pouco mais complexa: provar o sequente $\{(A \wedge B) \vee (A \wedge C)\} \vdash B \vee C$.

Para demonstrar o sequente anterior, utilizaremos a regra $\{\vee_E\}$ sobre a hipótese $(A \wedge B) \vee (A \wedge C)$ e utilizaremos as hipóteses introduzidas por esta regra para deduzir $B \vee C$.

$$\dfrac{\dfrac{}{(A \wedge B) \vee (A \wedge C)}\ \{_{ID}\} \quad \begin{array}{c} \dfrac{\dfrac{\overline{A \wedge B^1}\ \{_{ID}\}}{B}\ \{\wedge_{ED}\}}{B \vee C}\ \{\vee_{IE}\} \\ \dfrac{\dfrac{\overline{A \wedge C^1}\ \{_{ID}\}}{C}\ \{\wedge_{ED}\}}{B \vee C}\ \{\vee_{ID}\} \end{array}}{B \vee C}\ \{\vee_E\}^1$$

■

2.5.5 Contradição

A regra da contradição especifica que podemos deduzir *qualquer fórmula* a partir de uma dedução de $\perp$ (falso).

$$\frac{\Gamma \vdash \perp}{\Gamma \vdash \alpha}\ \{_{CTR}\}$$

Esta regra expressa a "inutilidade" de uma hipótese falsa, pois caso $\perp$ seja dedutível, então qualquer fórmula pode ser deduzida. Os próximos exemplos apresentam aplicações desta regra.

Exemplo 24. O sequente $\{A, \neg A\} \vdash B$ é provável utilizando as regras $\{_{CTR}\}$, $\{\rightarrow\ _E\}$ e $\{_{ID}\}$, conforme a dedução abaixo:

$$\dfrac{\dfrac{\overline{\neg A}\ \{_{ID}\} \quad \overline{A}\ \{_{ID}\}}{\bot}\ \{\rightarrow\ _E\}}{B}\ \{_{CTR}\}$$

Note que implicitamente esta demonstração utiliza o fato de que $\neg A$ é equivalente a $A \rightarrow \bot$ para utilizar a regra $\{\rightarrow\ _E\}$. ■

O próximo exemplo mostra uma propriedade do conectivo $\vee$: se $A \vee B$ é verdadeiro e $\neg A$ também o é, temos que necessariamente B deve ser verdadeiro.

Exemplo 25. O sequente $\{A \vee B, \neg A\} \vdash B$ possui a seguinte demonstração:

$$\dfrac{\overline{A \vee B}\ \{_{ID}\} \qquad \dfrac{\dfrac{\overline{\neg A}\ \{_{ID}\} \quad \overline{A^1}\ \{_{ID}\}}{\bot}\ \{\rightarrow\ _E\}}{B}\ \{_{CTR}\} \qquad \overline{B^1}\ \{_{ID}\}}{B}\ \{\vee\ _E\}^1$$

■

2.5.6 Reductio ad Absurdum

A regra *Reduction ad Absurdum* (redução ao absurdo) especifica que se conseguirmos deduzir $\bot$ a partir de $\neg\alpha$ então α deve ser uma fórmula verdadeira.

$$\dfrac{\Gamma \cup \{\neg\alpha\} \vdash \bot}{\Gamma \vdash \alpha}\ \{_{RAA}\}$$

A regra $\{_{RAA}\}$ é a formalização lógica de um conceito amplamente utilizando em matemática: o de prova por contradição. Se desejamos demonstrar que α é verdadeiro, basta supor que este é falso e a partir desta suposição obter um resultado absurdo (contradição).

Utilizando esta regra, podemos deduzir algumas demonstrações que não seriam possíveis utilizando outras regras da dedução natural. O seguinte exemplo, ilustra essa situação.

Exemplo 26. O seguinte sequente somente pode ser demonstrado utilizando a regra $\{_{RAA}\}$: $\vdash \neg\neg A \rightarrow A$. A demonstração deste sequente é apresentada a seguir:

$$\dfrac{\dfrac{\dfrac{\overline{\neg\neg A^1}\ \{_{ID}\} \quad \overline{\neg A^2}\ \{_{ID}\}}{\bot}\ \{\rightarrow\ _E\}}{A}\ \{_{RAA}\}^2}{\neg\neg A \rightarrow A}\ \{\rightarrow\ _I\}^1$$

Note que ao utilizarmos a regra de $\{_{RAA}\}$, adquirimos como hipótese adicional $\neg A$ que possibilita a utilização da regra $\{\rightarrow\ _E\}$, que conclui a demonstração. ■

2.5.7 Exercícios

1. Prove os seguintes sequentes usando dedução natural. Tente demonstrá-los sem utilizar a regra $\{_{RAA}\}$.

   (a) $\{(P \wedge Q) \wedge R,\, S \wedge T\} \vdash Q \wedge S$
   (b) $\{(P \wedge Q) \wedge R\} \vdash (P \wedge R) \vee Z$
   (c) $\{P, P \to (P \to Q)\} \vdash Q$
   (d) $\vdash (P \wedge Q) \to P$
   (e) $\{P\} \vdash Q \to P \wedge Q$
   (f) $\{P\} \vdash (P \to Q) \to Q$
   (g) $\vdash (P \wedge Q) \to P \vee Q$
   (h) $\{Q \to (P \to R),\, \neg R,\, Q\} \vdash \neg P$
   (i) $\{P\} \vdash Q \to (P \wedge Q)$
   (j) $\{(P \to R) \wedge (Q \to R),\, P \wedge Q\} \vdash Q \wedge R$
   (k) $\{P \to Q \to R, P \to Q\} \vdash P \to R$
   (l) $\{P \to Q, R \to S\} \vdash (P \vee R) \to (Q \vee S)$
   (m) $\{Q \to R\} \vdash (P \to Q) \to (P \to R)$
   (n) $\{(P \wedge Q) \vee (P \wedge R)\} \vdash P \wedge (Q \vee R)$
   (o) $\{P \to Q \wedge R\} \vdash (P \to Q) \wedge (P \to R)$
   (p) $\{(P \to Q) \wedge (P \to R)\} \vdash P \to (Q \wedge R)$
   (q) $\{P \to Q\} \vdash (((P \wedge Q) \to P) \wedge (P \to P \wedge Q))$
   (r) $\{P \to (Q \vee R), Q \to S, R \to S\} \vdash P \to S$
   (s) $\vdash \neg P \to P \to P \to Q$
   (t) $\{P \wedge Q \to R, R \to S, Q \wedge \neg S\} \vdash \neg P$
   (u) $\{(P \to Q) \to R, S \to \neg P, T, \neg S \wedge T \to Q\} \vdash R$
   (v) $\{(S \to P) \vee (T \to Q)\} \vdash (S \to q) \vee (T \to P)$
   (w) $\{\neg(P \to Q)\} \vdash Q \to P$

2. Prove os seguintes sequentes

   (a) $\{\neg(A \vee B)\} \vdash \neg A \wedge \neg B$
   (b) $\{\neg A \wedge \neg B\} \vdash \neg(A \vee B)$
   (c) $\{\neg(A \wedge B)\} \vdash \neg A \vee \neg B$
   (d) $\{\neg A \vee \neg B\} \vdash \neg(A \wedge B)$

3. Demonstre os seguintes sequentes. Nestes sequentes você terá que utilizar a regra $\{_{RAA}\}$ para deduzí-los.

   (a) $\{A \to B\} \vdash \neg A \vee B$
   (b) $\vdash (\neg B \to \neg A) \to (A \to B)$
   (c) $\vdash (A \to B) \to (\neg A \to B) \to B$
   (d) $\vdash ((A \to B) \to A) \to A$

2.6 Álgebra Booleana para Lógica Proposicional

Até o presente momento, apresentamos duas abordagens para o estudo da lógica proposicional: uma baseada na semântica, utilizando tabelas verdade e, uma abordagem sintática utilizando regras de inferência da dedução natural. Além destas duas abordagens, existe uma terceira, a álgebra booleana, que é uma abordagem axiomática para o estudo da lógica.

A álgebra booleana consiste de um conjunto de leis que estabelecem quando duas fórmulas podem ser consideradas logicamente equivalentes. A próxima definição apresenta as condições para que duas fórmulas sejam consideradas equivalentes.

Definição 18 (Equivalência Lógica)**.** Dizemos que duas fórmulas α e β são equivalentes, $\alpha \equiv \beta$, se estas possuem o mesmo valor lógico para uma mesma atribuição de valores às suas variáveis. Podemos verificar se α e β são equivalentes se a fórmula $\alpha \leftrightarrow \beta$ é uma tautologia. ■

Exemplo 27. As fórmulas $\neg\neg A$ e A são equivalentes, o que pode ser verificado pela seguinte tabela verdade: ■

A	$\neg A$	$\neg\neg A$	$\neg\neg A \leftrightarrow A$
F	T	F	T
T	F	T	T

Evidentemente, o uso de tabelas verdade para determinar a equivalência lógica de duas fórmulas possui o inconveniente de que o número de linhas de uma tabela verdade é exponencial no número de variáveis de uma fórmula. O objetivo desta seção é apresentar a álgebra booleana que permite determinar se duas fórmulas são equivalentes sem a utilização de tabelas verdade.

A álgebra booleana é uma forma de raciocínio algébrico sobre fórmulas, o que, de maneira simples, permite: 1) mostrar que duas fórmulas são iguais por meio de uma sequência de igualdades [3] e; 2) se $x = y$ e você possui uma expressão que possui ocorrências de x, você pode substituir ocorrências de x por y nesta expressão. Essa última propriedade é conhecida como *indiscernibilidade de valores iguais*[4]. Como exemplo, considere que $x = y + 2$ e que $z = 2 \times x + 5$, usando a propriedade de indiscernibilidade de iguais, temos que

$$\begin{array}{lcl} z & = & \\ 2 \times x + 5 & = & \{\text{pela def. de } z\} \\ 2 \times (y + 2) + 5 & = & \{\text{por } x = y + 2\} \\ 2 \times y + 4 + 5 & = & \\ 2 \times y + 9 & & \end{array}$$

Note que neste exemplo, envolvendo aritmética, apresentamos uma justificativa para cada passo da dedução de que as fórmulas z e $2 \times y + 9$ são equivalentes. Considera-se uma boa prática rotular cada passo de uma equação com

[3] A noção de "sequência de igualdades" é formalizada em termos da seguinte propriedade, denominada *transitividade*: se $a = b$ e $b = c$ então $a = c$.

[4] Essa regra é comumente citada na comunidade de lógica e teoria de tipos como regra de Leibniz, que pode ser expressa da seguinte maneira: Seja P uma propriedade qualquer, se sabemos que $x = y$ e que a propriedade P é verdadeira para x, então esta também deve ser para o valor y.

a justificativa que permite concluir a próxima expressão da cadeia de igualdades. Adotaremos essa convenção durante a apresentação do conteúdo de álgebra booleana.

2.6.1 Leis da Álgebra Booleana

A álgebra boolena consiste de um conjunto de equações que descreve propriedades algébricas de proposições. Estas equações são normalmente chamadas de "leis", uma vez que estas são aceitas como verdadeiras a priori. Dizemos que uma proposição é uma lei se esta é sempre verdadeira, independente dos valores lógicos atribuídos às suas variáveis[5].

As leis da álgebra boolena são análogas às leis da álgebra convencional. Existem leis que especificam que certos valores agem como elementos neutros, outras para dizer que certas operações são associativas e que algumas operações distribuem sobre outras. Na álgebra convencional, temos que a adição é associativa, isto é que para quaisquer valores numéricos x, y e z temos que $x + (y + z) = (x + y) + z$. A multiplicação, por sua vez, distribui com respeito a adição, isto é, para x, y e z temos que $x \times (y + z) = (x \times y) + (x \times z)$. Leis similares existem para os conectivos da lógica proposicional. As próximas seções apresentarão estas regras.

2.6.2 Leis Envolvendo Constantes

As leis envolvendo constantes especificam como as constantes lógicas interagem com os conectivos $\wedge$ e $\vee$.

$\alpha \wedge \bot$	$\equiv$	$\bot$	$\{\wedge - \text{null}\}$
$\alpha \vee \top$	$\equiv$	$\top$	$\{\vee - \text{null}\}$
$\alpha \wedge \top$	$\equiv$	α	$\{\wedge - \text{identidade}\}$
$\alpha \vee \bot$	$\equiv$	α	$\{\vee - \text{identidade}\}$

O seguinte exemplo mostra como estas leis podem ser utilizadas para demonstrar a equivalência de duas fórmulas.

Exemplo 28. As fórmulas $(A \vee \bot) \wedge (B \vee \top)$ e A são equivalentes, o que pode ser demonstrado pela seguinte dedução algébrica:

$(A \vee \bot) \wedge (B \vee \top)$	$\equiv$	$\{\vee - \text{identidade}\}$
$A \wedge (B \vee \top)$	$\equiv$	$\{\vee - \text{null}\}$
$A \wedge \top$	$\equiv$	$\{\wedge - \text{identidade}\}$
A		

■

A partir do exemplo anterior, podemos perceber a estrutura de uma demonstração de equivalência entre duas fórmulas. Como o objetivo é demonstrar uma igualdade, a dedução consiste de uma sequência de igualdades. A sequência inicia com o lado esquerdo da igualdade que desejamos deduzir e termina com o lado direito. Além disso, perceba que cada passo da dedução é justificado pelo nome da regra utilizada.

[5]Note que, desta maneira, toda tautologia pode ser vista como uma lei.

2.6.3 Leis Elementares dos Conectivos $\wedge$ e $\vee$

As leis seguintes descrevem que os conectivos $\wedge$ e $\vee$ são idempotentes, associativos e comutativos. Se α é uma fórmula e $\circ$ um conectivo binário, dizemos que $\circ$ é idempotente se $\alpha \circ \alpha = \alpha$. Por sua vez, dizemos que $\circ$ é associativo se, para fórmulas α, β e γ, temos que $\alpha \circ (\beta \circ \gamma) = (\alpha \circ \beta) \circ \gamma$. Finalmente, dizemos que $\circ$ é comutativo, se a seguinte igualdade é verdadeira, para fórmulas α e β: $\alpha \circ \beta = \beta \circ \alpha$.

A tabela seguinte apresenta essas propriedades, em termos para os conectivos $\wedge$ e $\vee$.

$\alpha \wedge \alpha$	$\equiv$	α	$\{\wedge - \text{idempotente}\}$
$\alpha \vee \alpha$	$\equiv$	α	$\{\vee - \text{idempotente}\}$
$\alpha \wedge \beta$	$\equiv$	$\beta \wedge \alpha$	$\{\wedge - \text{comutativo}\}$
$\alpha \vee \beta$	$\equiv$	$\beta \vee \alpha$	$\{\vee - \text{comutativo}\}$
$\alpha \wedge (\beta \wedge \gamma)$	$\equiv$	$(\alpha \wedge \beta) \wedge \gamma$	$\{\wedge - \text{associativo}\}$
$\alpha \vee (\beta \vee \gamma)$	$\equiv$	$(\alpha \vee \beta) \vee \gamma$	$\{\vee - \text{associativo}\}$

O próximo exemplo mostram como utilizar essas regras para demonstrar uma equivalência.

Exemplo 29. As fórmulas $(\bot \wedge A) \vee B$ e B são equivalentes, o que pode ser confirmado pela seguinte demonstração:

$$
\begin{array}{lcl}
(\bot \wedge A) \vee B & \equiv & \{\wedge - \text{comutativo}\} \\
(A \wedge \bot) \vee B & \equiv & \{\wedge - \text{null}\} \\
\bot \vee B & \equiv & \{\vee - \text{comutativo}\} \\
B \vee \bot & \equiv & \{\vee - \text{identidade}\} \\
B &&
\end{array}
$$

■

Encerraremos esta seção apresentando um conjunto de leis que descreve o relacionamento dos conectivos $\wedge$ e $\vee$ com a negação lógica e leis que mostram que estes conectivos distribuem um sobre o outro. Em álgebra, sabemos que a multiplicação distribui sobre a adição, isto é, para quaisquer valores numéricos a, b e c temos que $a \times (b + c) = (a \times b) + (a \times c)$. A tabela seguinte apresenta estas leis:

$\alpha \wedge (\beta \vee \gamma)$	$\equiv$	$(\alpha \wedge \beta) \vee (\alpha \wedge \gamma)$	$\{\wedge - \text{distribui} - \vee\}$
$\alpha \vee (\beta \wedge \gamma)$	$\equiv$	$(\alpha \vee \beta) \wedge (\alpha \vee \gamma)$	$\{\vee - \text{distribui} - \wedge\}$
$\neg(\alpha \wedge \beta)$	$\equiv$	$\neg\alpha \vee \neg\beta$	$\{\text{DeMorgan} - \wedge\}$
$\neg(\alpha \vee \beta)$	$\equiv$	$\neg\alpha \wedge \neg\beta$	$\{\text{DeMorgan} - \vee\}$

As duas últimas regras apresentadas são conhecidas como leis de DeMorgan e estas possuem explicações intuitivas. Por exemplo, $\neg(\alpha \wedge \beta)$ especifica que "não é verdade que α e β são simultaneamente verdadeiros" logo, temos que ou α é falso ou β é falso. Pode-se explicar a regra DeMorgan $- \vee$ de maneira similar.

2.6.4 Leis Envolvendo a Negação

As leis algébricas relacionadas com o conectivo de negação são bem diretas e refletem o significado deste conectivo:

$\neg\top$	$\equiv$	$\bot$	{negação-$\top$}
$\neg\bot$	$\equiv$	$\top$	{negação-$\bot$}
$\alpha \wedge \neg\alpha$	$\equiv$	$\bot$	{complemento-$\wedge$}
$\alpha \vee \neg\alpha$	$\equiv$	$\top$	{complemento-$\vee$}
$\neg(\neg\alpha)$	$\equiv$	α	{dupla-negação}

Utilizando as leis apresentadas até o momento, podemos demonstrar a equivalência $A \wedge \neg(B \vee A) \equiv \bot$.

Exemplo 30. A fórmula $A \wedge \neg(B \vee A)$ é equivalente a $\bot$, conforme demonstrado abaixo:

$$\begin{array}{lcl} A \wedge \neg(B \vee A) & \equiv & \{\text{DeMorgan} - \vee\} \\ A \wedge \neg B \wedge \neg A & \equiv & \{\wedge - \text{comutativo}\} \\ A \wedge \neg A \wedge \neg B & \equiv & \{\text{complemento} - \wedge\} \\ \bot \wedge \neg B & \equiv & \{\wedge - \text{comutativo}\} \\ \neg B \wedge \bot & \equiv & \{\wedge - \text{null}\} \\ \bot & & \end{array}$$

■

2.6.5 Leis Envolvendo a Implicação e Bicondicional

As leis algébricas para a implicação e bicondicional mostram como expressar esses conectivos em termos de outros.

$\alpha \rightarrow \beta$	$\equiv$	$\neg\alpha \vee \beta$	{implicação}
$\alpha \leftrightarrow \beta$	$\equiv$	$(\alpha \rightarrow \beta) \wedge (\beta \rightarrow \alpha)$	{bicondicional}

Usando as equivalências até aqui apresentadas, podemos demonstrar alguns resultados conhecidos da lógica, como por exemplo, a contrapositiva de uma implicação, que é apresentada no próximo exemplo.

Exemplo 31. A fórmula $A \rightarrow B$ é equivalente a $\neg B \rightarrow \neg A$, conforme demonstrado a seguir:

$$\begin{array}{lcl} A \rightarrow B & \equiv & \{\text{implicação}\} \\ \neg A \vee B & \equiv & \{\text{dupla-negação}\} \\ \neg A \vee \neg(\neg B) & \equiv & \{\vee - \text{comutativo}\} \\ \neg(\neg B) \vee \neg A & \equiv & \{\text{implicação}\} \\ \neg B \rightarrow \neg A & & \end{array}$$

Note que no último passo, utilizamos a regra da implicação, que especifica que $\alpha \rightarrow \beta \equiv \neg\alpha \vee \beta$, sobre a fórmula $\neg(\neg B) \vee \neg A$. Neste caso, temos que $\alpha = \neg B$ e $\beta = \neg A$, o que nos permite deduzir que $\neg(\neg B) \vee \neg A = \neg B \rightarrow \neg A$. ■

Uma das aplicações da álgebra boolena é permitir expressar algumas funções lógicas (conectivos) em termos de outros. Por exemplo, utilizando as leis de

DeMorgan, podemos expressar o conectivo $\wedge$ em termos de $\vee$ e $\neg$, conforme apresentado abaixo:

$$\begin{array}{ll} A \wedge B & \equiv \\ \neg\neg A \wedge \neg\neg B & \equiv \\ \neg(\neg A \vee \neg B) & \end{array}$$

Uma vez que podemos expressar conectivos em termos de outros, cabe perguntar se existe um conjunto "mínimo" de conectivos a partir dos quais é possível definir todos os outros. A próxima definição formaliza este conceito.

Definição 19 (Conjunto Completo de Conectivos)**.** Seja $\mathcal{C} \subseteq \{\bot, \neg, \vee, \wedge, \rightarrow, \leftrightarrow\}$ um conjunto de conectivos. Dizemos que $\mathcal{C}$ é completo para $\{\bot, \neg, \vee, \wedge, \rightarrow, \leftrightarrow\}$ se é possível expressar todos os conectivos não presentes em $\mathcal{C}$ em termos dos conectivos presentes no conjunto $\mathcal{C}$ e variáveis. ■

Exemplo 32. O conjunto $\{\neg, \vee\}$ é completo, pois é possível expressar todos os outros conectivos da lógica utilizando apenas $\neg$, $\vee$ e variáveis, conforme apresentado abaixo:

1. A constante $\top$ pode ser representada como $\alpha \vee \neg\alpha$.

2. A constrante $\bot$ pode ser representada como $\neg(\alpha \vee \neg\alpha)$. Sabe-se que $\alpha \vee \neg\alpha \equiv \top$, pela regra $\{\vee - \text{null}\}$, e que $\bot \equiv \neg\top$. Logo, $\bot \equiv \neg(\alpha \vee \neg\alpha)$, para qualquer fórmula α.

3. Conectivo de conjunção pode ser represento por $\neg$ e $\vee$ da seguinte maneira, em que α e β são fórmulas quaisquer:

$$\begin{array}{ll} \alpha \wedge \beta & \equiv \\ \neg\neg\alpha \wedge \neg\neg\beta & \equiv \\ \neg(\neg\alpha \vee \neg\beta) & \end{array}$$

4. A implicação lógica possui representação direta: $\alpha \rightarrow \beta \equiv \neg\alpha \vee \beta$.

5. Finalmente, representamos o conectivo bicondicional da seguinte forma:

$$\begin{array}{ll} \alpha \leftrightarrow \beta & \equiv \\ (\alpha \rightarrow \beta) \wedge (\beta \rightarrow \alpha) & \equiv \\ (\neg\alpha \vee \beta) \wedge (\neg\beta \vee \alpha) & \end{array}$$

   Agora, utilizaremos o fato de que deduzimos em um item anterior que $A \wedge B \equiv \neg(\neg A \vee \neg B)$ e, consideraremos que $A = \neg\alpha \vee \beta$ e $B = \neg\beta \vee \alpha$. Com isso, obtemos:

$$\neg(\neg(\neg\alpha \vee \beta) \vee \neg(\neg\beta \vee \alpha))$$

   que é a representação do conectivo bicondicional em termos de $\neg$ e $\vee$.

■

2.6.6 Exercícios

1. Determine se as seguintes fórmulas são ou não equivalentes.
   (a) $P \leftrightarrow Q$ e $(P \rightarrow Q) \wedge (\neg P \rightarrow \neg Q)$
   (b) $(P \wedge \neg Q) \vee (\neg P \wedge Q)$ e $(P \vee Q) \wedge \neg(P \wedge Q)$
2. Prove as seguintes equivalências usando raciocínio álgebrico:
   (a) $(A \vee B) \wedge B \equiv B$
   (b) $(\neg A \wedge B) \vee (A \wedge \neg B) \equiv (A \vee B) \wedge \neg(A \wedge B)$
   (c) $((A \rightarrow B) \rightarrow A) \rightarrow A \equiv T$
3. Mostre que o conjunto $\{\neg, \wedge\}$ é completo para os conectivos $\{\bot, \neg, \vee, \wedge, \rightarrow, \leftrightarrow\}$.
4. Mostre que o conjunto $\{\neg, \rightarrow\}$ é completo para os conectivos $\{\bot, \neg, \vee, \wedge, \rightarrow, \leftrightarrow\}$.
5. Descreva como podemos determinar que uma fórmula é uma tautologia utilizando leis da álgebra booleana.
6. O conectivo de negação conjunta, $\alpha \downarrow \beta$, é definido como verdadeiro sempre que α ou β são falsos.
   (a) Apresente a tabela verdade para $\alpha \downarrow \beta$.
   (b) Mostre que o conjunto $\{\bot, \downarrow\}$ é completo para os conectivos $\{\bot, \neg, \vee, \wedge, \rightarrow, \leftrightarrow\}$.

2.7 Formas Normais

A lógica proposicional possui diversas aplicações práticas na ciência da computação. Porém, algumas destas aplicações exigem que as fórmulas possuam uma certa estrutura. Nesta seção apresentaremos duas formas normais que são amplamente utilizadas: a forma normal disjuntiva, aplicada em minimização de fórmulas lógicas e a formal normal conjuntiva, utilizada como entrada para algoritmos para teste de satisfazibilidade.

2.7.1 Forma Normal Conjuntiva

A definição seguinte apresenta condições para que uma dada fórmula bem formada da lógica proposicional seja considerada uma fórmula na forma normal conjuntiva.

Definição 20 (Forma Normal Conjuntiva)**.** Definimos o conjunto de fórmulas da lógica proposicional na forma normal conjuntiva (FNC), da seguinte maneira:

1. As constantes lógicas $\bot$ e $\top$ são fórmulas na forma normal conjuntiva.
2. Seja $\mathcal{V}$ o conjunto de todas as variáveis da lógica proposicional. Seja $\alpha \in \mathcal{V}$ e $\neg\alpha \in \mathcal{V}$ são fórmulas na forma normal conjuntiva. Dá-se o nome de literal a fórmulas que são variáveis ou negação de variáveis.

3. Seja $\{l_1, ..., l_n\}$ um conjunto de $n \geq 0$ literais. Então,

$$\bigvee_{i=1}^{n} l_i$$

é uma fórmula na forma normal conjuntiva. Dá-se o nome de cláusula a fórmulas que consistem apenas de uma disjunção de literais.

4. Seja $\{C_1, ..., C_n\}$ um conjunto de $n \geq 0$ cláusulas. Então,

$$\bigwedge_{i=1}^{n} C_i$$

é uma fórmula na forma normal conjuntiva.

■

Os exemplos a seguir ilustram o conceito de fórmulas na forma normal conjuntiva.

Exemplo 33. São exemplos de fórmulas na forma normal conjuntiva:

- $\bot, \top, \alpha, \neg\alpha$, em que α é uma variável.
- Sejam x_1, x_2 e x_3 variáveis da lógica proposicional. Então $\neg x_1 \vee x_2 \vee x_3$ e $x_1 \vee \neg x_2 \vee \neg x_3$ são cláusulas e, portanto, são, cada uma, fórmulas na forma normal conjuntiva.
- Sejam $\neg x_1 \vee x_2 \vee x_3$, $x_1 \vee \neg x_2 \vee \neg x_3$ e $x_4 \vee \neg x_5$ cláusulas. Então, a fórmula $(\neg x_1 \vee x_2 \vee x_3) \wedge (x_1 \vee \neg x_2 \vee \neg x_3) \wedge (x_4 \vee \neg x_5)$ está na forma normal conjuntiva.

Os próximos exemplos mostram fórmulas que não estão na forma normal conjuntiva.

- $A \rightarrow B \wedge (C \leftrightarrow B)$, não está na FNC, pois possui o conectivo de implicação e bicondicional.
- $\neg(A \wedge B)$, não está na FNC, pois a negação não está associada somente a variáveis.
- $A \vee (B \wedge C)$, não está na FNC, pois, de acordo com a definição 20, o conectivo $\vee$ ocorre apenas em cláusulas e não no nível mais externo da fórmula.

■

A definição 20 mostra que apenas um subconjunto das fórmulas bem formadas da lógica proposicional pode ser considerada na FNC. Porém, este fato não constitui uma limitação já que toda fórmula bem formada da lógica proposicional pode ser convertida para FNC, seguindo-se o algoritmo seguinte. O próximo exemplo mostra como converter uma fórmula da lógica para a forma normal conjuntiva.

Algorithm 1 Convertendo para a Forma Normal Conjuntiva

Require: Uma fórmula bem formada α

1: **for** Todas as subfórmulas β, γ, φ de α **do**
2: Eliminar bicondicionais usando $\beta \leftrightarrow \gamma = (\beta \rightarrow \gamma) \wedge (\gamma \rightarrow \beta)$
3: Eliminar implicações usando $\beta \rightarrow \gamma = \neg\beta \vee \gamma$
4: Empurrar as negações usando as leis de DeMorgan

$$\begin{aligned} \neg(\beta \wedge \gamma) &= \neg\beta \vee \neg\gamma \\ \neg(\beta \vee \gamma) &= \neg\beta \wedge \neg\gamma \end{aligned}$$

até que estas fiquem associadas a variáveis

5: Elimine duplas negações usando $\neg\neg\beta = \beta$
6: Aplique, enquanto possível, a distributividade do $\vee$ sobre $\wedge$:

$$\beta \vee (\gamma \wedge \varphi) = (\beta \vee \gamma) \wedge (\beta \vee \varphi))$$

7: **end for**
8: A fórmula resultante estará na forma normal conjuntiva.

Exemplo 34. Mostraremos como converter a fórmula $A \rightarrow (B \wedge C)$ para a forma normal conjuntiva. Para isso, executaremos, passo a passo, o algoritmo 1.

1. O passo 1 do algoritmo é desnecessário, já que a fórmula $A \rightarrow (B \wedge C)$ não possui bicondicionais.

2. No passo 2, eliminamos a implicação:

$$A \rightarrow (B \wedge C) = \neg A \vee (B \wedge C)$$

3. No passo 3, não há nada a fazer já que a negação está associada somente a variáveis.

4. No passo 4, não há nada a fazer já que não há dupla negações.

5. No passo 5, temos que distribuir o $\vee$ sobre o $\wedge$, obtendo:

$$\neg A \vee (B \wedge C) = (\neg A \vee B) \wedge (\neg A \vee C)$$

■

Exemplo 35. Mostraremos como converter a fórmula $\neg(A \leftrightarrow B)$ para a forma normal conjuntiva, passo a passo.

1. No passo 1, eliminamos o bicondicional obtendo:

$$\neg(A \leftrightarrow B) = \neg[(A \rightarrow B) \wedge (B \rightarrow A)]$$

2. No passo 2, eliminamos a implicação obtendo:

$$\neg[(A \rightarrow B) \wedge (B \rightarrow A)] = \neg[(\neg A \vee B) \wedge (\neg B \vee A)]$$

3. No passo 3, movemos as negações utilizando as leis de DeMorgan:

$$\begin{array}{ll} \neg[(\neg A \vee B) \wedge (\neg B \vee A)] & = \\ \neg(\neg A \vee B) \vee \neg(\neg B \vee A) & = \\ (\neg\neg A \wedge \neg B) \vee (\neg\neg B \wedge \neg A) & \end{array}$$

4. No passo 4, eliminamos as duplas negações:

$$(\neg\neg A \wedge \neg B) \vee (\neg\neg B \wedge \neg A) = (A \wedge \neg B) \vee (B \wedge \neg A)$$

5. Finalmente, no passo 5 distribuímos o $\vee$ sobre $\wedge$:

$$\begin{array}{ll} (A \wedge \neg B) \vee (B \wedge \neg A) & = \\ ((A \wedge \neg B) \vee B) \wedge ((A \wedge \neg B) \vee \neg A) & = \\ ((A \vee B) \wedge (\neg B \vee B)) \wedge ((A \vee \neg A) \wedge (\neg B \vee \neg A)) & \end{array}$$

Obtendo assim a fórmula equivalente a $\neg(A \leftrightarrow B)$ na forma normal conjuntiva. ■

2.7.2 Forma Normal Disjuntiva

A próxima definição especifica as condições para que uma fórmula esteja na forma normal disjuntiva (FND).

Definição 21 (Forma Normal Disjuntiva)**.** Definimos o conjunto de fórmulas da lógica proposicional na forma normal disjuntiva (FND), da seguinte maneira:

1. As constantes lógicas $\bot$ e $\top$ são fórmulas na forma normal disjuntiva.
2. Seja $\mathcal{V}$ o conjunto de todas as variáveis da lógica proposicional. Seja $\alpha \in \mathcal{V}$ e $\neg\alpha \in \mathcal{V}$ são fórmulas na forma normal disjuntiva. Dá-se o nome de literal a fórmulas que são variáveis ou negação de variáveis.
3. Seja $\{l_1, ..., l_n\}$ um conjunto de $n \geq 0$ literais. Então,

$$\bigwedge_{i=1}^{n} l_i$$

é uma fórmula na forma normal disjuntiva. Dá-se o nome de cláusula dual a fórmulas que consistem apenas de uma conjunção de literais.

4. Seja $\{C_1, ..., C_n\}$ um conjunto de $n \geq 0$ cláusulas duais. Então,

$$\bigvee_{i=1}^{n} C_i$$

é uma fórmula na forma normal disjuntiva.

■

O seguinte algoritmo pode ser utilizado para converter qualquer fórmula da lógica proposicional para a forma normal disjuntiva.

A seguir, os próximos exemplos mostram a conversão de duas fórmulas para a forma normal disjuntiva utilizando o algoritmo 2.

Algorithm 2 Convertendo para a Forma Normal Disjuntiva

Require: Uma fórmula bem formada α

for Todas as subfórmulas β, γ, φ de α **do**

Eliminar bicondicionais usando $\beta \leftrightarrow \gamma = (\beta \rightarrow \gamma) \wedge (\gamma \rightarrow \beta)$

Eliminar implicações usando $\beta \rightarrow \gamma = \neg\beta \vee \gamma$

Empurrar as negações usando as leis de DeMorgan

$$\begin{aligned} \neg(\beta \wedge \gamma) &= \neg\beta \vee \neg\gamma \\ \neg(\beta \vee \gamma) &= \neg\beta \wedge \neg\gamma \end{aligned}$$

até que estas fiquem associadas a variáveis

Elimine duplas negações usando $\neg\neg\beta = \beta$

Aplique, enquanto possível, a distributividade do $\wedge$ sobre $\vee$:

$$\beta \wedge (\gamma \vee \varphi) = (\beta \wedge \gamma) \vee (\beta \wedge \varphi))$$

end for

A fórmula resultante estará na forma normal disjuntiva.

Exemplo 36. Mostraremos como converter a fórmula $A \rightarrow (B \wedge C)$ para a forma normal disjuntiva. Para isso, executaremos, passo a passo, o algoritmo 2.

1. O passo 1 do algoritmo é desnecessário, já que a fórmula $A \rightarrow (B \wedge C)$ não possui bicondicionais.
2. No passo 2, eliminamos a implicação:
$$A \rightarrow (B \wedge C) = \neg A \vee (B \wedge C)$$
3. No passo 3, não há nada a fazer já que a negação está associada somente a variáveis.
4. No passo 4, não há nada a fazer já que não há dupla negações.
5. No passo 5, não há nada a fazer já que não há conjunções a serem distribuídas sobre disjunções.

■

Exemplo 37. Mostraremos como converter a fórmula $\neg(A \leftrightarrow B)$ para a forma normal disjuntiva, passo a passo.

1. No passo 1, eliminamos o bicondicional obtendo:
$$\neg(A \leftrightarrow B) = \neg[(A \rightarrow B) \wedge (B \rightarrow A)]$$
2. No passo 2, eliminamos a implicação obtendo:
$$\neg[(A \rightarrow B) \wedge (B \rightarrow A) = \neg[(\neg A \vee B) \wedge (\neg B \vee A)]$$
3. No passo 3, movemos as negações utilizando as leis de DeMorgan:
$$\begin{aligned} &\neg[(\neg A \vee B) \wedge (\neg B \vee A)] \quad = \\ &\neg(\neg A \vee B) \vee \neg(\neg B \vee A) \quad = \\ &(\neg\neg A \wedge \neg B) \vee (\neg\neg B \wedge \neg A) \end{aligned}$$

4. No passo 4, eliminamos as duplas negações:

$$(\neg\neg A \wedge \neg B) \vee (\neg\neg B \wedge \neg A) = (A \wedge \neg B) \vee (B \wedge \neg A)$$

5. No passo 5, não há o que fazer pois não há conjunções para serem distribuídas sobre disjunções. Logo, a fórmula

$$(A \wedge \neg B) \vee (B \wedge \neg A)$$

está na forma normal disjuntiva.

■

2.7.3 Exercícios

1. Para cada uma das fórmulas a seguir, apresente fórmulas equivalentes na forma normal conjuntiva e disjuntiva.

   (a) $(A \wedge B) \vee C \rightarrow A \wedge (B \vee C)$

   (b) $A \wedge \neg(\neg A \vee \neg B)$

   (c) $A \wedge B \rightarrow \neg A$

   (d) $(A \rightarrow B) \rightarrow [(A \vee C) \rightarrow (B \vee C)]$

   (e) $A \rightarrow (B \rightarrow A)$

   (f) $(A \wedge B) \leftrightarrow (\neg B \vee \neg A)$

2.8 Considerações Meta-matemáticas

A meta-matemática consiste em utilizar técnicas matemáticas para o estudo da própria matemática. Nesta seção apresentaremos alguns conceitos importantes relativos à lógica proposicional, sem demonstrá-los.

2.8.1 Corretude e Completude

Neste capítulo, apresentamos a sintaxe, semântica e um sistema de provas para a lógica proposicional: a dedução natural, que nos permite demonstrar consequências lógicas.

Apesar da dedução natural possuir uma semântica intuitiva, não apresentamos como este se relaciona com a semântica da lógica proposicional. A relação de um certo sistema de provas para um formalismo e a semântica deste é dada por propriedades conhecidas como correção e completude. Essas propriedades expressam o relacionamento de um sistema de provas com o conceito de validade semântica do formalismo em questão. Para o caso da lógica proposicional, podemos considerar que o conceito de validade é exatamente o conceito de tautologia.

Desta forma, estamos interessados em saber:

1. Se sempre que uma fórmula for dedutível no sistema de prova em questão, então esta é válida — Essa propriedade é conhecida como correção.

2. Se toda fórmula válida possui uma dedução no sistema de prova — Essa propriedade é conhecida como completude.

A dedução natural é um sistema de prova correto e completo para a lógica proposicional. Essas propriedades são enunciadas a seguir.

Teorema 1 (Correção da dedução natural)**.** *Seja α uma fórmula bem formada qualquer da lógica proposicional. Se $\vdash \alpha$, então $\models \alpha$*

Por sua vez, a completude especifica que toda tautologia é demonstrável utilizando o sistema de dedução natural.

Teorema 2 (Completude da dedução natural)**.** *Seja α uma fórmula bem formada qualquer da lógica proposicional. Se $\models \alpha$, então $\vdash \alpha$.*

Note que representamos o fato de uma fórmula α ser uma tautologia utilizando o conceito de consequência lógica, $\models \alpha$.

Infelizmente, não possuímos as ferramentas necessárias para demonstrar esses resultados. Para provar estes teoremas precisamos utilizar indução matemática, que será abordada posteriormente.

2.8.2 Decidibilidade

Dizemos que um conjunto é *decidível*, se existe um algoritmo que determina se um certo valor é ou não um elemento deste conjunto. Para apresentar este conceito para a lógica proposicional, devemos, primeiramente, caracterizar o conjunto de fórmulas desta lógica em termos do conceito de *teoria*, que é apresentado a seguir.

Definição 22 (Teoria)**.** Dada uma linguagem $\mathcal{L}$ e uma noção de validade semântica sobre fórmulas de $\mathcal{L}$, denominada $\models$, a *teoria* $\langle \mathcal{L}, \models \rangle$ é o conjunto de fórmulas válidas de $\mathcal{L}$, isto é.

$$\langle \mathcal{L}, \models \rangle = \{\alpha \mid \models \alpha\}$$

■

Para o caso da lógica proposicional, temos que $\mathcal{L}$ é o conjunto de fórmulas bem formadas desta lógica e o conceito de validade corresponde à consequência lógica (tautologia).

Definição 23 (Decidibilidade)**.** Seja T um conjunto qualquer. Dizemos que um subconjunto $S \subseteq T$ é decidível se existe um algoritmo $f : T \to \mathcal{B}$, que partir de um elemento $t \in T$, determina se este pertence ou não ao conjunto S[6] e termina para todas as possíveis entradas. ■

Teorema 3 (Decidibilidade da lógica proposicional)**.** *A teoria $\langle \mathcal{F}, \models \rangle$, em que $\mathcal{F}$ é o conjunto de fórmulas bem formadas da lógica proposicional e $\models$ a relação de consequência lógica, é decidível.*

Demonstração. Para mostrar que a teoria $\langle \mathcal{F}, \models \rangle$ é decidível devemos apenas apresentar um algoritmo que, a partir de uma fórmula $\alpha \in \mathcal{F}$ determina se $\models \alpha$ é verdadeiro ou não (isto é, se α é ou não uma tautologia). O algoritmo que soluciona esse problema consiste em construir a tabela verdade de α e verificar se todas as linhas desta são iguais a T. □

[6] Lembre-se que $\mathcal{B}$ é o conjunto de valores booleanos (introduzido no capítulo 1).

2.9 Notas Bibliográficas

Existem diversos bons livros que abordam a lógica proposicional. Citaremos apenas alguns:

3

Lógica de Predicados

> "Observe a estrada e diga-me quem você vê", disse o Rei.
> "Eu vejo ninguém", disse Alice.
> "Mas que excelente visão você possui!", exclamou o Rei. "Ver Ninguém a tal distância! Eu nunca o vi!"
>
> Lewis Carroll, Alice no País dos Espelhos.

3.1 Motivação

No capítulo anterior, estudamos a lógica proposicional de um ponto de vista sintático e semântico e, além disso, utilizamos a dedução natural e álgebra Booleana para verificar consequências e equivalências lógicas.

Apesar de possuir uma série de aplicações, a lógica proposicional possui limitações. A seguir apresentamos um exemplo que ilustra este problema.

Exemplo 38. Considere o seguinte argumento dedutivo:

> Todo homem é mortal.
> Sócrates é um homem.
> Logo, Sócrates é mortal.

De acordo com nossa noção informal de dedução, este parece ser um argumento válido. Sendo assim, este pode ser representado como um sequente demonstrável utilizando dedução natural. Porém, quando tentamos representar estas sentenças como fórmulas da lógica, podemos perceber que nenhuma delas possui conectivos lógicos. Logo, todas podem ser consideradas proposições simples, conforme mostramos na tabela a seguir:

Utilizando a modelagem apresentada na tabela acima, o sequente

$$\{A, B\} \vdash C$$

Sentença	Fórmula
Todo homem é mortal	A
Sócrates é um homem	B
Sócrates é mortal	C

representa o argumento dedutivo em questão. Mas, como o leitor já deve ter percebido, este não é provável utilizando o sistema de dedução natural apresentado neste texto. ■

Na seção 2.8 apresentamos que o sistema de dedução natural é completo para a lógica proposicional, desta forma, toda consequência lógica deve possuir um sequente provável correspondente. De acordo com uma noção intuitiva de dedução lógica, o argumento anterior é correto e portanto, deveríamos conseguir representá-lo como um sequente demonstrável, o que, conforme apresentado, não é possível.

O problema na modelagem formal deste argumento é que a lógica proposicional não possui expressividade para representar sentenças que possuam alguma das seguintes formas:

- Todo x possui a propriedade p.
- Algum x possui a propriedade p.

Tais sentenças possuem, implicitamente, um conjunto sobre o qual a frase em questão deve ser interpretada como verdadeira ou falsa. No caso do exemplo anterior, temos que a frase:

Todo homem é mortal.

implicitamente se refere ao conjunto de todos os seres humanos. Esta mesma frase poderia ser re-escrita de maneira a tornar o conjunto de seres humanos (que está "implícito") explícito como:

Todo elemento do conjunto de seres humanos possui a propriedade "mortal".

Para representar sentenças como "Todo homem é mortal" precisamos de estender a lógica proposicional de forma que sejamos capazes de expressar propriedades sobre elementos de um certo conjunto. O objetivo deste capítulo é estudarmos esta lógica, conhecida como lógica de predicados ou lógica de primeira ordem.

3.2 Introdução à lógica de predicados

Para representar argumentos dedutivos como o apresentado na seção anterior, devemos estender a lógica proposicional de maneira a sermos capazes de nos referir a elementos de um certo conjunto, denominado universo de discurso, e suas propriedades. Para isso, a linguagem da lógica proposicional será extendida com termos, que denotam elementos do universo de discurso; predicados, que representam propriedades destes elementos e quantificadores, que permite a especificação de que "todos" ou "algum" elemento do conjunto possui uma certa propriedade especificada.

As próximas subseções apresentam uma descrição informal dos conceitos de universo de discurso, predicados e quantificadores.

3.2.1 Universo de discurso

De maneira simples, qualquer conjunto não vazio pode ser considerado como um universo de discurso para interpretação de uma fórmula. O conjunto {Sócrates} é um universo de discurso válido para o argumento dedutivo apresentado no início deste capítulo, assim como o conjunto $\{a\}$ ou o conjunto de todos os seres humanos, já que todos são conjuntos não vazios de elementos.

Denominamos por *constante*, um elemento qualquer do universo de discurso. A lógica de predicados também permite a definição de símbolos funcionais (funções), que podem ser utilizados para representar elementos do universo de discurso sem a necessidade de nomeá-lo. O próximo exemplo ilustra a utilização de símbolos funcionais e constantes.

Exemplo 39. Considere como universo de discurso o conjunto H de todos os seres humanos, as constantes *Hermengarda* e *Eudésio* e a função *mãe*, que a partir de uma constante h que representa um ser humano retorna a constante que denota a mãe de h. Se considerarmos que *Hermengarda* é mãe de *Eudésio*, temos que ao aplicarmos a função *mãe* a constante *Eudésio* o resultado será *Hermengarda*.

Note que podemos nos referir ao mesmo elemento usando uma constante (como, por exemplo, *Hermengarda*) ou utilizando símbolos funcionais (como, por exemplo, *mãe(Eudésio)*. ■

3.2.2 Predicados

Predicados descrevem propriedades que elementos do universo de discurso podem ou não possuir portanto, possuem valor verdadeiro ou falso. A seguir apresentamos alguns exemplos de predicados.

Exemplo 40. Vamos considerar o argumento dedutivo apresentado no início deste capítulo, repetido abaixo:

Sentença	Fórmula
Todo homem é mortal	A
Sócrates é um homem	B
Sócrates é mortal	C

Note que nestas sentenças existe uma propriedade: *mortal*. Logo, ao formalizarmos este argumento, *mortal* será um predicado que descreverá a propriedade de "ser mortal" dos elementos do universo de discurso sobre o qual esta fórmula está sendo interpretada.

Como outros exemplos de predicados, considere $x > 10$ que é um exemplo de um predicado e seu valor lógico depende do valor da variável x. Note que, além de possuir a variável x, o predicado $x > 10$ também envolve uma constante: 10. Evidentemente, predicados podem envolver diversas variáveis como $x > y$ ou apenas constantes, como em $10 < 4$. ■

Usualmente representamos predicados de maneira concisa como em $F(x)$, em que F é o símbolo que representa uma certa propriedade. Considerando a propriedade *mortal*, esta poderia ser representada pelo predicado $M(x)$, que pode ser lido como "x é mortal". Predicados podem ter uma quantidade $n \geq 0$

de parâmetros. Quando um predicado possui nenhum parâmetro, dizemos que este é uma variável proposicional.

3.2.3 Quantificadores

Existem dois quantificadores na lógica de predicados: o quantificador universal, representado pelo símbolo $\forall$, e o quantificador existencial, representado pelo símbolo $\exists$.

Na lógica de predicados utilizamos variáveis para representar objetos arbitrários do universo de discurso em questão. Por exemplo, se desejamos especificar uma propriedade da álgebra, uma variável (por exemplo, x) representa um número qualquer. Se a propriedade se refere a geometria, variáveis podem representar objetos geométricos, como pontos, triângulos, etc.

Se $P(x)$ é uma fórmula qualquer da lógica de predicados, representamos a sentença "todo x possui a propriedade P", por $\forall x.P(x)$. De maneira similar, representamos a sentença "algum x possui a propriedade P" por $\exists x.P(x)$.

Dizemos que a fórmula $\forall x.P(x)$ é considerada verdadeira se para todo elemento do universo de discurso em questão a propriedade P é verdadeira. Por sua vez, a fórmula $\exists x.P(x)$ é considerada verdadeira se pelo menos um elemento do universo de discurso torna a propriedade P verdadeira. A seguir apresentamos alguns exemplos.

Exemplo 41. Considere o seguinte universo de discurso $U = \{\text{Zeus,Sócrates}\}$, que a constante Zeus representa um deus da mitologia grega e Sócrates o conhecido filósofo. Além disso, considere o predicado $M(x)$ que é verdadeiro se "x é um mortal". Desta forma, temos que a fórmula $\forall x.M(x)$ é falsa em U, já que nem todo elemento deste conjunto torna o predicado M verdadeiro[1].

Porém, se considerarmos o universo de discurso $F = \{\text{Sócrates, Platão}\}$, temos que a fórmula $\forall x.M(x)$ é verdadeira, visto que todos os elementos de F satisfazem a propriedade M. Em ambos os conjuntos U e F a fórmula $\exists x.M(x)$ é verdadeira, visto que há pelo menos um elemento nestes conjuntos que representa um mortal. ■

Exemplo 42. Neste exemplo vamos considerar a tarefa de interpretar a validade de algumas fórmulas envolvendo o predicado $>$ sobre números. Estas fórmulas são: $\forall x.x > 0$ e $\forall x.\exists y.x > y$.

Inicialmente, vamos considerar como universo de discurso o conjunto dos números naturais. A primeira fórmula é *falsa* pois, temos que o número $0 \in \mathbb{N}$ não é maior que 0.

A segunda fórmula também é falsa pois esta especifica que para qualquer número natural x, existe y tal que $x > y$, o que não é verdadeiro para $x = 0$. Porém, ao considerarmos o conjunto dos números inteiros, $\mathbb{Z}$, temos que a primeira fórmula é falsa (porquê?) e a segunda verdadeira, visto que no conjunto dos números inteiros, para qualquer x existe um número menor que x.

Considerando qualquer conjunto numérico a seguinte propriedade é falsa: $\exists y.\forall x.y > x$, já que esta especifica que existe algum valor y que é maior que qualquer outro valor x. ■

[1] O elemento Zeus torna este predicado falso, já que, este representa o deus grego, que é imortal.

3.2.4 Formalizando sentenças

Para a formalização de sentenças utilizando a lógica de predicados devemos especificar o universo de discurso, a interpretação de predicados e dos símbolos funcionais que podem ser utilizados. Os próximos exemplos ilustram a utilização destes conceitos na formalização de sentenças na língua portuguesa.

Exemplo 43. Nos próximos exemplos, vamos considerar sentenças envolvendo o predicado $C(x, y)$, que denota "x conhece y", o predicado $G(x, y)$ que representa "x gosta de y", a função *mãe* que possui significado óbvio. O universo de discurso considerado será, novamente, o conjunto de todos os seres humanos.

- A sentença "Todo mundo gosta de alguém" pode ser representada como: $\forall x.\exists y.G(x, y)$.
- A sentença "Astobaldo não gosta de sua mãe" pode ser representada como $\neg G(\text{Astobaldo}, \mathit{mãe}(\text{Astobaldo}))$.
- A sentença "Ninguém gosta de todo mundo" pode ser formalizada como $\neg\exists x.\forall y.G(x, y)$. Note que esta sentença é equivalente a "Não existe alguém que goste de todo mundo".
- A sentença "Todos gostam da mãe de Carlos" pode ser representada como $\forall x.G(x, \mathit{mãe}(\text{Carlos}))$.
- A sentença "Todos que conhecem Clementino, não gostam da mãe dele" pode ser representada como $\forall x.C(x, \text{Clementino}) \rightarrow \neg G(x, \mathit{mãe}(\text{Clementino}))$.

■

3.3 Exercícios

1. Considere como universo de discurso o conjunto de todos os seres humanos, e que *Holmes* e *Moriarty* são constantes. Além disso, considere o predicado $C(x, y)$ que denota "x pode capturar y". Com base no apresentado, represente as seguintes sentenças como fórmulas da lógica de predicados.
   (a) Holmes pode capturar qualquer um que pode capturar Moriarty.
   (b) Holmes pode capturar alguém que Moriarty pode capturar.
   (c) Se alguém pode capturar Moriarty, então Holmes também pode.
   (d) Ninguém pode capturar Holmes, a menos que possa capturar Moriarty.
   (e) Qualquer um que pode capturar Holmes pode capturar todos que Holmes pode capturar.
2. Expresse as seguintes frases utilizando lógica de predicados. Para isso, crie predicados, funções e constantes do domínio de interpretação que julgar adequados.
   (a) Quem faz exercícios tem melhor qualidade de vida.
   (b) Alunos não gostam de fazer provas.

(c) Nem tudo que reluz é ouro.

(d) Quem conhece Godofredo o adora.

(e) Não conheço quem não odeie as brincadeiras de Eudésio.

(f) Ninguém visita Hermengarda, a menos que ela esteja afônica.

3. Considerando como universo de discurso o conjunto de alunos e professores de uma universidade e os seguintes predicados:

$A(x, y)$	x admira y
$S(x, y)$	x estava presente em y
$P(x)$	x é um professor
$E(x)$	x é um estudante
$L(x)$	x é uma aula

e a constante *Maria*, represente as seguintes sentenças como fórmulas da lógica de predicados.

(a) Maria admira todo professor.

(b) Algum professor admira Maria.

(c) Maria admira a si própria.

(d) Nenhum estudante estava presente em todas as aulas.

(e) Nenhuma aula teve a presença de todos os estudantes.

(f) Nenhuma aula teve a presença de qualquer estudante.

3.4 Sintaxe da lógica de predicados

A seção anterior teve como objetivo mostrar como codificar sentenças como fórmulas da lógica de predicados e introduziu, de maneira informal, a sintaxe e como fórmulas são interpretadas em um determinado universo de discurso. Nesta seção vamos definir de maneira precisa a sintaxe da lógica de predicados, para na próxima seção definirmos a semântica de fórmulas bem formadas nesta lógica.

Ao observarmos com atenção os exemplos de fórmulas, podemos perceber que estas são compostas de componentes de dois tipos: valores que representam elementos do universo de discurso e componentes lógicos. Damos o nome de *termos* aos componentes da sintaxe da lógica de predicados que representam elementos do universo de discurso.

3.4.1 Termos

O conjunto $\mathcal{T}$ de termos da lógica de predicados é formado por variáveis, constantes e funções aplicadas a ambos. A seguir apresentamos a definição formal do conjunto $\mathcal{T}$.

Definição 24 (Conjunto de Termos da Lógica de Predicados)**.** O conjunto $\mathcal{T}$ de termos da lógica de predicados é definido recursivamente como:

- Seja $\mathcal{V}$ o conjunto de todas as variáveis da lógica de predicados. Então $\mathcal{V} \subseteq \mathcal{T}$, isto é, toda variável é um termo.

- Seja $\mathcal{C}$ o conjunto de todas as constantes da lógica de predicados. Então, $\mathcal{C} \subseteq \mathcal{T}$, isto é, toda constante é um termo.
- Seja $\mathcal{F}$ o conjunto de todos os símbolos funcionais da lógica de predicados. Considere que $f \in \mathcal{F}$ é uma função de aridade[2] n, $n \geq 1$, e que $t_1, ..., t_n \in \mathcal{T}$. Então, $f(t_1, ..., t_n) \in \mathcal{T}$, isto é, toda função de aridade n aplicada a n termos é também um termo.

Todos os elementos de $\mathcal{T}$ podem ser construídos pelas regras anteriores. ■

A seguir apresentamos alguns exemplos de termos e como estes são construídos utilizando a definição 24.

Exemplo 44. Suponha que a, b e c sejam constantes de algum universo de discurso, f e g duas funções de aridade 1 e 2, respectivamente. As expressões seguintes são termos da lógica de predicados:

1. $g(a, b)$
2. $f(g(f(a), c))$

A fórmula 1) pode ser construída da seguinte maneira: primeiramente, a e b, por serem constantes, são termos. Finalmente, $g(a, b)$ é um termo pois a função g, de aridade 2, está aplicada a dois termos. Por sua vez, a fórmula 2) é bem formada, pois tanto a quanto c são termos (já que ambos são constantes). Sendo assim, $f(a)$ é um termo, já que a função f, de aridade 1, está aplicada a um termo. De maneira similar, temos que $g(f(a), c)$ é um termo pois, a função g (de aridade 2) está aplicada a $f(a)$ e c. Finalmente, $f(g(f(a), c))$ é um termo pois, a função f, de aridade 1, está aplicada a $g(f(a), c)$.

As seguintes expressões não podem ser consideradas termos já que não respeitam a aridade das funções f e g: $f(a, c)$, $g(f(a))$. ■

3.4.2 Fórmulas

A partir da definição de termos, podemos definir o conjunto de fórmulas bem formadas da lógica de predicados, $\mathbb{F}$.

Definição 25 (Fórmulas bem formadas)**.** O conjunto de fórmulas bem formadas da lógica de predicados, $\mathbb{F}$, pode ser definido recursivamente da seguinte maneira:

1. Seja p um predicado de aridade $n \geq 0$ e $t_1, ..., t_n \in \mathcal{T}$ termos. Então, $p(t_1, ..., t_n) \in \mathbb{F}$, isto é, $p(t_1, ..., t_n)$ é uma fórmula (tais fórmulas são usualmente denominadas de fórmulas atômicas).
2. $\bot, \top \in \mathbb{F}$.
3. Sejam $\alpha, \beta \in \mathbb{F}$ fórmulas quaisquer. Então:
   (a) $\neg\alpha \in \mathbb{F}$.
   (b) $\alpha \circ \beta \in \mathbb{F}$, em que $\circ \in \{\vee, \wedge, \rightarrow, \leftrightarrow\}$.
   (c) Se $x \in \mathcal{V}$ (isto é, x é uma variável), então: $\forall x.\alpha \in \mathbb{F}$ e $\exists x.\alpha \in \mathbb{F}$.
   (d) $(\alpha) \in \mathbb{F}$.

[2] Denomina-se por aridade o número de parâmetros de uma função.

Toda fórmula bem formada da lógica de predicados pode ser construída utilizando as regras anteriores.

■

A seguir apresentamos alguns exemplos de fórmulas bem formadas.

Exemplo 45. Primeiramente, considere os seguintes exemplos de fórmulas atômicas:

1. f — um predicado de aridade 0 (isto é, uma variável proposicional).

2. pai(Adão,Abel) — um predicado de aridade 2 (pai) e duas constantes: Adão e Abel. Esta fórmula poderia representar a sentença "Adão é pai de Abel".

3. casados(João,irmã(Maria)) — um predicado de aridade 2 (casados), aplicado a constante João e ao termo irmã(Maria), em que irmã é uma função. Esta fórmula poderia representar a sentença "João é casado com a irmã de Maria".

A seguir apresentamos alguns exemplos de fórmulas não atômicas.

1. pai(Adão,Abel) $\wedge$ pai(Adão,Caim). Esta fórmula representa a sentença "Adão é pai de Abel e Caim".

2. $\exists x$.tia(x,Joaquim). Esta fórmula representa a sentença "Joaquim tem uma tia".

3. $\forall x$.gosta(x,mãe(x)) Esta fórmula representa a sentença "Todos gostam de sua respectiva mãe".

■

Assim como na lógica proposicional, utilizaremos precedências entre conectivos e quantificadores na lógica de predicados para evitar o uso excessivo de parênteses. Para os conectivos, utilizaremos as mesmas regras de precedência da lógica proposicional e consideraremos que quantificadores possuem a mesma precedência que o conectivo $\neg$.

3.4.3 Variáveis Livres e Ligadas

Antes de apresentarmos os conceitos de variável livre e ligada, devemos definir de maneira precisa o escopo de um quantificador em uma fórmula.

Definição 26 (Escopo de quantificadores)**.** Seja $x \in \mathcal{V}$ uma variável e $\alpha \in \mathbb{F}$ uma fórmula. Dizemos que o escopo da variável x em $\forall x.\alpha$ é a fórmula α. De maneira similar, o escopo de x em $\exists x.\alpha$ é também a fórmula α. ■

Dizemos que uma variável é livre em uma certa fórmula se esta não está no escopo de nenhum quantificador. Uma variável que não é livre é dita ser ligada. O próximo exemplo ilustra estes conceitos.

Exemplo 46. Considere a seguinte equação envolvendo símbolos da aritmética sobre números naturais:

$$x = 5y$$

Esta equação não pode ser considerada verdadeira ou falsa, uma vez que o seu valor lógico depende dos valores atribuídos às variáveis x e y. Note que, tanto x quanto y, são variáveis que ocorrem livres na fórmula anterior. Considere a seguinte variação da fórmula anterior:

$$\exists y.x = 5y$$

Note que o valor lógico da fórmula acima depende apenas do valor de x e não de y. Esta fórmula pode ser descrita na língua portuguesa sem mencionarmos a variável y como "x é um múltiplo de 5". O fato do valor lógico desta fórmula não depender da variável y é uma consequência desta ser uma variável ligada. ■

De maneira simples, podemos determinar se uma variável x é livre ou ligada em uma fórmula utilizando a função fv, que calcula o conjunto de variáveis livres de uma dada fórmula. Esta é definida a seguir.

Definição 27 (Variáveis livres)**.** Seja $t \in \mathcal{T}$ um termo qualquer. O conjunto de variáveis livres em t, $fv_{\mathcal{T}}(t)$, é definido recursivamente como:

$$fv_{\mathcal{T}}(t) = \begin{cases} \{x\} & \text{se } t = x, \text{ para algum } x \in \mathcal{V}, \text{ isto é, se } t \text{ é uma variável.} \\ \emptyset & \text{se } t = c, \text{ para algum } c \in \mathcal{C}, \text{ isto é, se } t \text{ é uma constante.} \\ \bigcup_{i=1}^{n} fv_{\mathcal{T}}(t_i) & \text{se } t = f(t_1, ..., t_n), \text{ em que } t_1, ..., t_n \in \mathcal{T}, \text{ e } f \text{ possui aridade } n. \end{cases}$$

Dada uma fórmula $\alpha \in \mathbb{F}$, o conjunto de variáveis livres de α, $fv(\alpha)$, é definido recursivamente como:

$$fv(\alpha) = \begin{cases} \bigcup_{i=1}^{n} fv_{\mathcal{T}}(t_i) & \text{se } t = p(t_1, ..., t_n) \text{ em que } p \text{ possui aridade } n \geq 0. \\ \emptyset & \text{se } \alpha = \top \text{ ou } \alpha = \bot \\ fv(\beta) & \text{se } \alpha = \neg\beta \\ fv(\beta) \cup fv(\gamma) & \text{se } \alpha = \beta \circ \gamma, \circ \in \{\vee, \wedge, \rightarrow, \leftrightarrow\}. \\ fv(\beta) - \{x\} & \text{se } \alpha = \forall x.\beta \text{ ou } \alpha = \exists x.\beta. \end{cases}$$

Uma fórmula $\alpha \in \mathbb{F}$ em que $fv(\alpha) = \emptyset$ é dita ser *fechada*. Caso contrário, *aberta*. ■

O leitor deve ter notado que não apresentamos uma função para o cálculo de variáveis ligadas de uma fórmula. Esta é deixada como exercício (veja o exercício 1 da seção 3.5).

3.4.4 Substituição

Conforme discutido no exemplo 46, fórmulas que possuem variáveis livres só podem possuir significado se estas forem substituídas por "valores concretos", isto é, termos cujo significado não depende de nenhum valor externo à definição da fórmula em questão.

De maneira mais formal, para atribuir significado a fórmulas com variáveis livres, estas precisam ser substituídas por termos que não possuem este tipo de variável. A operação de substituir uma variável por um termo qualquer é denominada *substituição*.

Definição 28 (Substituição)**.** Sejam $x \in \mathcal{V}$ e $t, s \in \mathcal{T}$ uma variável e termos, respectivamente. Denotamos por $[x \mapsto s]\, t$ o termo obtido pela substituição de toda ocorrência livre de x em t por s. Mais formalmente (considere que $y \in \mathcal{V}$, $c \in \mathcal{C}$ e que $x \equiv y$ é verdadeiro se x for igual a variável y):

$$
\begin{array}{lcll}
[x \mapsto s]\, y & = & \begin{cases} s & \text{se } x \equiv y \\ y & \text{caso contrário.} \end{cases} & (1) \\
{[x \mapsto s]\, c} & = & c & (2) \\
{[x \mapsto s]\, f(t_1, ..., t_n)} & = & f([x \mapsto s]\, t_1, ..., [x \mapsto s]\, t_n) & (3)
\end{array}
$$

Utilizando a definição de substituição para termos, podemos definir a substituição para fórmulas quaisquer (em que $\alpha, \beta \in \mathbb{F}, \circ \in \{\vee, \wedge, \rightarrow, \leftrightarrow\}$):

$$
\begin{array}{lcll}
[x \mapsto s] p(t_1, ..., t_n) & = & p([x \mapsto s]\, t_1, ..., [x \mapsto s]\, t_n) & (4) \\
{[x \mapsto s] \top} & = & \top & (5) \\
{[x \mapsto s] \bot} & = & \bot & (6) \\
{[x \mapsto s](\neg \alpha)} & = & \neg([x \mapsto s]\, \alpha) & (7) \\
{[x \mapsto s](\alpha \circ \beta)} & = & ([x \mapsto s]\alpha) \circ ([x \mapsto s]\beta) & (8) \\
{[x \mapsto s](\forall y.\alpha)} & = & \begin{cases} \forall y.[x \mapsto s]\alpha & \text{se } x \not\equiv y \\ \forall y.\alpha & \text{se } x \equiv y \end{cases} & (9) \\
{[x \mapsto s](\exists y.\alpha)} & = & \begin{cases} \exists y.[x \mapsto s]\alpha & \text{se } x \not\equiv y \\ \exists y.\alpha & \text{se } x \equiv y \end{cases} & (10)
\end{array}
$$

Note que as equações que definem a substituição para fórmulas com quantificadores proíbem que a substituição seja realizada sobre variáveis ligadas. ■

O seguinte ilustra a operação de substituição sobre termos e fórmulas.

Exemplo 47. Seja $x \in \mathcal{V}$ uma variável e $f(a)$ um termo (a é uma constante). Temos que o resultado de aplicar a substituição $[x \mapsto f(a)]$ ao termo $g(a, h(x, a))$ é $g(a, h(f(a), a))$. O resultado de aplicar esta mesma substituição à fórmula $\forall y.g(y, x)$ é $\forall y.g(y, f(a))$. De maneira similar, a aplicação da substituição $[x \mapsto g(a, a)]$ à fórmula $\exists x.f(x)$ produz o termo $\exists x.f(x)$, já que este não possui variáveis livres. ■

3.5 Exercícios

1. Apresente a definição recursiva para uma função que calcula o conjunto de variáveis ligadas de uma fórmula da lógica de predicados.

2. Para cada um dos termos da lógica de predicados a seguir, use a definição de fórmulas bem formadas (definição 25) para justificar o porquê estes podem ser considerados fórmulas bem formadas. Considere que os símbolos f, g são funções de aridade 1 e 2, respectivamente, que a, b são constantes e p, q são predicados de aridade 1 e 2 respectivamente.

   (a) $f(a)$

   (b) $\forall x.q(f(a), x)$

   (c) $\exists y.p(y) \wedge \forall x.q(f(a), g(x, b))$.

3. Obtenha o conjunto de variáveis livres para cada uma das fórmulas seguintes utilizando a definição 27. Considere que a e b são constantes.

(a) $\forall x.(p(x,z) \rightarrow q(y)) \wedge s(a,x)$
(b) $\exists y.p(y,z) \wedge \forall x.q(f(a), g(x,b))$.
(c) $\exists y.p(y) \wedge \forall x.q(f(a), g(x,b))$.

3.6 Semântica da lógica de predicados

Na lógica proposicional, para ser possível determinar o valor lógico de uma fórmula basta uma interpretação para as variáveis nela contidas. Isso é feito atribuindo, às variável da fórmula, todas as possíveis combinações de verdadeiro (T) ou falso (F). Porém, como apresentado informalmente na seção 3.2, para interpretarmos fórmulas da lógica de predicados, devemos possuir um universo de discurso (que dá significado às constantes) e conjuntos de relações e funções que atribuem significado aos símbolos predicativos e funcionais, respectivamente.Isto é, para definirmos o significado de fórmulas da lógica de predicados, necessitamos de uma *estrutura*, conceito este apresentado na definição seguinte.

Definição 29 (Estrutura)**.** Uma estrutura é uma tripla $I = (U, R, F)$, em que:

- U é um conjunto não vazio tal que para cada constante $a \in \mathcal{T}$, temos que $a^I \in U$, em que a^I é a denotação de a em U.
- R é um conjunto de relações, em que, para cada símbolo predicativo p de aridade $n \geq 1$, existe uma relação n-ária $p^I \subseteq U^n$.
- F é um conjunto de funções, em que, para cada símbolo funcional f, de aridade $n \geq 1$, existe uma função $f^I : U^n \rightarrow U \in F$.

■

A definição de uma função para atribuição significado a fórmulas da lógica de predicados é apresentada a seguir. Primeiramente, apresentamos a definição da semântica de um termo. Como termos denotam elementos do universo de discurso, a função semântica para termos deverá produzir como resultado um elemento do conjunto U, considerando uma estrutura $I = (U, R, F)$.

Definição 30 (Semântica de termos)**.** Seja $I = (U, R, F)$ uma estrutura. Definimos a função $\varepsilon : \mathcal{T} \rightarrow U$, que define a semântica de um termo t como um elemento $u \in U$, recursivamente como:

$$\begin{array}{lcll} \varepsilon(a) & = & a^I, & \text{em que } a^I \in U \quad (1) \\ \varepsilon(f(t_1, ..., t_n)) & = & f^I(\varepsilon(t_1), ..., \varepsilon(t_n)) & \text{em que } f^I \in F \text{ e } t_1, ..., t_n \in \mathcal{T} \quad (2) \end{array}$$

■

Note que apesar de variáveis serem consideradas termos, não apresentamos a semântica destas, uma vez que termos possuem somente variáveis livres e, como já citado anteriormente, apresentaremos a semântica apenas de fórmulas fechadas. Isto não constitui uma limitação, uma vez que, substituições podem ser utilizadas para eliminar variáveis livres de fórmulas.

Antes de apresentarmos a função semântica de fórmulas da lógica de predicados, vamos considerar um exemplo para ilustrar a semântica de termos desta lógica.

Exemplo 48. Suponha um universo de discurso em que objetos sejam países e cidades, dentre as quais citamos *Rio de Janeiro, Berlim, Nova York, Tóquio,* entre outras. Desejamos formalizar as seguintes funções e constantes envolvendo países e cidades:

- *capital*: função de aridade 1 que associa a cada país sua respectiva capital. Representaremos, na lógica de predicados, a função *capital* pelo símbolo funcional *cap*. Logo, $cap^I = capital$.
- Representarmos as constantes *Rio de Janeiro, Berlim, Nova York, Tóquio* por RJ, BL, NY e TK, respectivamente. Além disso, consideraremos que a constante Brasil é representada pelo termo BR e Alemanha por GE.

Considere, agora, a tarefa de interpretrar o significado do termo $cap(GE)$. Utilizando a definição da semântica de termos, temos:

$\varepsilon(cap(GE))$	=	
$cap^I(\varepsilon(GE))$	=	{pela eq. (2) de ε.}
$cap^I(GE^I)$	=	{pela eq. (1) de ε.}
$cap^I(Alemanha)$	=	{pela semântica da constante GE}
$capital(Alemanha)$	=	{pela semântica do símbolo funcional *cap*.}
$Berlim$		{pela semântica da função da função *capital*}.

Logo, o termo $cap(GE)$ denota o mesmo elemento (a cidade de Berlim) que a constante BL. ■

A seguir, é apresentada a semântica para fórmulas da lógica de predicados.

Definição 31 (Semântica de Fórmulas)**.** Seja $I = (U, R, F)$ uma estrutura. Definimos a função $[\![\_]\!] : \mathbb{F} \to \{T, F\}$, que associa a cada fórmula fechada da lógica de predicados o seu respetivo valor lógico, recursivamente como (em que $t_1, ..., t_n$ representam termos, $c \in \mathcal{C}$ uma constante qualquer, α, β fórmulas quaisquer e $\circ \in \{\vee, \wedge, \to, \leftrightarrow\}$):

$$
\begin{array}{lcll}
[\![\bot]\!] & = & F & (1)\\
[\![\top]\!] & = & T & (2)\\
[\![p(t_1, ..., t_n)]\!] & = & (\varepsilon(t_1), ..., \varepsilon(t_n)) \in p^I & (3)\\
[\![\neg\alpha]\!] & = & \neg[\![\alpha]\!] & (4)\\
[\![\alpha \circ \beta]\!] & = & [\![\alpha]\!] \circ [\![\beta]\!] & (5)\\
[\![\forall x.\alpha]\!] & = & \bigwedge\limits_{u \in \mathcal{C}} [\![[x \mapsto u]\alpha]\!] & (6)\\
[\![\exists x.\alpha]\!] & = & \bigvee\limits_{u \in \mathcal{C}} [\![[x \mapsto u]\alpha]\!] & (7)
\end{array}
$$

A notação $[x \mapsto u]\alpha$ denota a fórmula α em que toda ocorrência da variável livre x é substituída pela constante u. ■

O próximo exemplo ilustra a utilização das funções semânticas para lógica de predicados.

Exemplo 49. Considere a seguinte estrutura $I = (U, R, F)$, em que:

- $U = \{2, 3, 4\}$, em que cada um dos números $x \in U$, será representado pela constante x.

- O conjunto R é formado pelos seguintes conjuntos:
  - $par = \{2, 4\}$, que será representado pelo símbolo predicativo p, de aridade 1.
  - *ímpar* $= \{3\}$, que será representado pelo símbolo predicativo i, de aridade 1.
  - $M = \{(2, 3), (2, 4), (3, 4)\}$, que será representado pelo símbolo predicativo m, de aridade 2.
- O conjunto F é vazio, isto é, não existem funções nesta estrutura.

Tendo apresentado o significado dos símbolos não funcionais em termos da estrutura I, considere a tarefa de calcular o valor lógico da seguinte fórmula $\forall x.p(x) \vee i(x)$. O cálculo de cada uma destas, utilizando a definição 31, é mostrada passo-a-passo a seguir.

$$
\begin{array}{lcl}
\forall x.p(x) \vee i(x) & = & \\
([\![p(2) \vee i(2)]\!]) \wedge ([\![p(3) \vee i(3)]\!]) \wedge ([\![p(4) \vee i(4)]\!]) & = & \{\text{pela eq.(6)}\} \\
([\![p(2)]\!] \vee [\![i(2)]\!]) \wedge ([\![p(3)]\!] \vee [\![i(3)]\!]) \wedge ([\![p(4)]\!] \vee [\![i(4)]\!]) & = & \{\text{pela eq.(5)}\} \\
\quad (\varepsilon(2) \in \textit{par} \vee \varepsilon(2) \in \textit{ímpar}) \quad \wedge & & \\
\quad (\varepsilon(3) \in \textit{par} \vee \varepsilon(3) \in \textit{ímpar}) \quad \wedge & = & \{\text{pela eq. (3)}\} \\
\quad (\varepsilon(4) \in \textit{par} \vee \varepsilon(4) \in \textit{ímpar}) & & \\
T \wedge T \wedge T & = & \{\text{pela def. de } \varepsilon \text{ e } \textit{par, ímpar.}\} \\
T & &
\end{array}
$$

Logo, de acordo com a definição 31, a fórmula $\forall x.p(x) \vee i(x)$ é verdadeira para a estrutura I. ■

Com base na semântica de fórmulas, podemos classificá-las de maneira similar ao que fazemos com a lógica proposicional. A próxima definição formaliza estes conceitos.

Definição 32 (Classificação de fórmulas)**.** Seja α uma fórmula bem formada da lógica de predicados. Dizemos que α é satisfazível se existe uma estrutura I tal que $[\![\alpha]\!] = T$. De maneira similar, dizemos que α é falseável se existe uma estrutura tal que $[\![\alpha]\!] = F$. Uma fórmula α é dita ser uma tautologia[3] se esta é verdadeira para toda estrutura I. Uma fórmula α é uma contradição se não existe uma estrutura que a satisfaça. Finalmente, α é uma contingência se esta for satisfazível e falseável. ■

3.7 Exercícios

1. Para cada uma das fórmulas a seguir, indique se ela é verdadeira ou falsa, quando o universo de discurso é cada um dos seguintes conjuntos: $\mathbb{N}$: conjunto dos números naturais, $\mathbb{Z}$: conjunto dos números inteiros e $\mathbb{R}$ conjunto dos números reais. Considere que os símbolos matemáticos possuem o significado usual.

[3] Também chamada por alguns autores de fórmulas válidas

Fórmula	$\mathbb{N}$	$\mathbb{Z}$	$\mathbb{R}$
$\exists x.x^2 = 2$			
$\forall x.\exists y.x^2 = y$			
$\forall x.x \neq 0 \rightarrow \exists y.xy = 1$			
$\exists x.\exists y.(x + 2y^2 = 2) \wedge (2x + 4y = 5)$			

3.8 Dedução natural para lógica de predicados

As regras para dedução natural para lógica proposicional podem ser estendidas para lidar com os quantificadores da lógica de predicados. Apenas quatro regras adicionais são necessárias para lidar com a lógica de predicados, a saber: regras para introdução e eliminação dos quantificadores universal e existencial.

Uma maneira de compreender as regras para ambos os quantificadores é vê-los como generalizações dos conectivos de conjunção (para o quantificador universal) e disjunção (para o quantificador existencial). As próximas subseções utilizarão essa analogia para apresentar as regras de introdução e eliminação destes quantificadores.

3.8.1 Regras para o quantificador universal

Conforme apresentado na definição 31, o quantificador universal pode ser entendido como uma conjunção de fórmulas, em que a variável ligada a este quantificador é substituída por cada um dos elementos do universo de discurso em questão. Mais formalmente:

$$\forall x.P(x) \equiv \bigwedge_{u \in C} [x \mapsto u]P(x)$$

Nas subseções seguintes, utilizaremos esta analogia para apresentarmos, informalmente, as regras de introdução e eliminação deste quantificador.

Introdução do quantificador universal $\{\forall_I\}$

Se pensarmos que o quantificador universal é uma generalização da conjunção, podemos conjecturar que a regra de introdução deste quantificador deve ser similar a

$$\frac{\bigwedge_{u \in C} [x \mapsto u]P(x)}{\forall x.P(x)} \ \forall_{I1}$$

Isto é, para concluirmos que $\forall x.P(x)$ é provável basta demonstrar $[x \mapsto u]P(x)$, em que u representa cada uma das constantes do universo de discurso sobre o qual esta fórmula deve ser interpretada. Esta analogia é válida (e útil) se o universo de discurso é finito e possui poucos elementos. Caso contrário, essa abordagem para provar $\forall x.P(x)$ é impraticável.

Propriedades que envolvem "todos" os possíveis valores de um universo de discurso são comuns na matemática e, portanto, deve haver uma maneira mais simples de demonstrar afirmativas da forma $\forall x.P(x)$. A idéia utilizada para provar fórmulas que utilizam o quantificador universal pode ser expressa de maneira intuitiva da seguinte forma: Se uma certa propriedade P é verdadeira

para um objeto *arbitrário* do universo de discurso, então esta deve ser verdadeira para todo elemento deste conjunto. Porém, esta explicação deixa a seguinte pergunta: Quando podemos considerar que um certo objeto é ou não "arbitrário"? Há uma resposta simples para isso, baseada em um critério sintático sobre sequentes. Dizemos que um valor x é arbitrário se este não pertence ao conjunto de variáveis livres do sequente a ser demonstrado. Logo, podemos concluir que $\Gamma \vdash \forall x.P(x)$ se conseguirmos demonstrar $P(x)$, em que x é um valor arbitrário, isto é $x \notin fv(\Gamma)$ e $fv(\Gamma)$ é definido como:

$$fv(\Gamma) = \bigcup_{\alpha \in \Gamma} fv(\alpha)$$

Abaixo apresentamos a regra de introdução do quantificador universal.

$$\frac{P(x) \quad x \notin fv(\Gamma)}{\forall x.P(x)} \ \{\forall_I\}$$

Assim como fizemos para lógica proposicional, vamos omitir completamente o conjunto de hipóteses Γ e também a demonstração de que $x \notin fv(\Gamma)$, pois, normalmente a última demonstração é imediata a partir das hipóteses de um dado sequente. Visando ilustrar essa convenção, o seguinte exemplo ilustra a utilização da regra $\{\forall_I\}$.

Exemplo 50. Considere a tarefa de demonstrar o sequente: $\vdash \forall x.E(x) \rightarrow E(x) \vee \neg E(x)$. Como este utiliza o quantificador universal, iniciaremos a demonstração utilizando a regra $\{\forall_I\}$. Ao aplicarmos esta regra, devemos demonstrar que $E(x) \rightarrow E(x) \vee \neg E(x)$, para um valor x arbitrário. Uma vez que o conjunto de hipóteses deste sequente é vazio, a variável livre x em $E(x) \rightarrow E(x) \vee \neg E(x)$ pode ser considerada arbitrária, já que esta não ocorre livre nas hipóteses.

A prova deste sequente é apresentada abaixo:

$$\dfrac{\dfrac{\dfrac{\dfrac{}{E(x)^1}\ \{_{ID}\}}{E(x) \vee \neg E(x)}\ \{\vee_{IE}\}}{E(x) \rightarrow E(x) \vee \neg E(x)}\ \{\rightarrow_I\}^1}{\forall x.E(x) \rightarrow E(x) \vee \neg E(x)}\ \{\forall_I\}$$

■

Eliminação do quantificador universal $\{\forall_E\}$

A regra de eliminação do quantificador universal permite-nos concluir, a partir de $\forall x.P(x)$, que $[x \mapsto a]P(x)$, em que a é uma constante ou uma variável livre na conclusão da regra $\{\forall_E\}$.

$$\frac{\forall x.P(x)}{[x \mapsto a]P(x)}\ \{\forall_E\}$$

A regra $\{\forall_E\}$ é uma generalização das regras para eliminação da conjunção, uma vez que, ao utilizarmos esta regra estamos concluindo um dos possivelmente infinitos componentes da conjunção $\bigwedge_{u \in C}[x \mapsto u]\, P(x)$.

Exemplo 51. Considere a tarefa de demonstrar o seguinte sequente $\{F(a), \forall x.F(x) \rightarrow G(x)\} \vdash G(a)$. Para demonstrar esse sequente, utilizaremos a eliminação da implicação, para concluir $G(a)$ a partir de $F(a)$ e $F(a) \rightarrow G(a)$. Esta última pode ser deduzida utilizando a regra $\{\forall_E\}$ sobre a hipótese $\forall x.F(x) \rightarrow G(x)$, conforme apresentado abaixo:

$$\dfrac{\dfrac{}{F(a)}\ \{_{ID}\} \quad \dfrac{\dfrac{}{\forall x.F(x) \rightarrow G(x)}\ \{_{ID}\}}{F(a) \rightarrow G(a)}\ \{\forall_E\}}{G(a)}\ \{\rightarrow\ _E\}$$

■

Restrições sobre as regras do quantificador universal

O objetivo desta seção é apresentar exemplos que mostrem a necessidade das restrições sobre a aplicabilidade das regras de introdução e eliminação do quantificador universal.

Primeiramente, vamos considerar a restrição $x \notin fv(\Gamma)$ sobre a regra $\{\forall_I\}$. Esta é realmente necessária? Ao invés de tentarmos apresentar um argumento formal (o que foge ao escopo deste texto), apresentaremos um exemplo que, ao não utilizarmos essa restrição, produziremos um argumento incorreto.

Exemplo 52. Considere a seguinte fórmula da lógica de predicados: $0 = 0 \rightarrow \forall x.(x = 0)$ que evidentemente não é uma tautologia[4] e a seguinte "demonstração" (incorreta):

$$\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{}{x = 0^1}\ \{_{ID}\}}{\forall x.x = 0}\ \{\forall_I\}}{x = 0 \rightarrow \forall x.x = 0}\ \{\rightarrow\ _I\}^1}{\forall x.x = 0 \rightarrow \forall x.x = 0}\ \{\forall_I\}}{0 = 0 \rightarrow \forall x.x = 0}\ \{\forall_E\}$$

Note que a aplicação da regra $\{\forall_I\}$ sobre a hipótese $x = 0$ é ilegal, uma vez que a variável x ocorre livre nas hipóteses. ■

A restrição imposta sobre a regra $\{\forall_E\}$ é que o valor que substitui a variável ligada ao quantificador eliminado deve ocorrer livre na conclusão. O próximo exemplo ilustra que, ao ignorar essa restrição, podemos deduzir fórmulas que não são consequências lógicas das hipóteses do sequente em questão.

Exemplo 53. Considere a tarefa de demonstrar o seguinte sequente:

$$\vdash \forall x.\neg\forall y.x = y \rightarrow \neg\forall y.y = y$$

. A "demonstração" (incorreta) deste é apresentada abaixo:

$$\dfrac{\dfrac{\dfrac{\dfrac{}{\forall x.\neg\forall y.x = y^1}\ \{_{ID}\}}{\neg\forall y.y = y}\ \{\forall_E\}}{(\forall x.\neg\forall y.x = y) \rightarrow \neg\forall y.y = y}\ \{\rightarrow\ _I\}^1$$

Note que a aplicação da regra $\{\forall_E\}$ logo no início da dedução está incorreto, já que a variável eliminada (x) foi substituída por y, que ocorre ligada na conclusão, alterando a semântica da fórmula em questão. ■

[4] A menos que o universo de discurso em questão possua apenas a constante 0.

3.8.2 Regras para o quantificador existencial

Conforme apresentado na definição 31, o quantificador existencial pode ser entendido como uma disjunção de fórmulas, em que a variável ligada a este quantificador é substituída por cada uma das constantes do universo de discurso em questão. Mais formalmente:

$$\exists x.P(x) \equiv \bigvee_{u \in C} [x \mapsto u]P(x)$$

Nas subseções seguintes, utilizaremos esta analogia para apresentarmos, informalmente, as regras de introdução e eliminação deste quantificador.

Introdução do quantificador existencial $\{\exists_I\}$

De maneira intuitiva, podemos concluir que $\exists x.P(x)$ se for possível provar que a propriedade P é verdadeira para algum valor a. Mais formalmente:

$$\frac{[x \mapsto a]P(x)}{\exists x.P(x)} \{\exists_I\}$$

Note que, para demonstrar que $\exists x.P(x)$ basta mostrar que *existe* um valor a que torna a propriedade P verdadeira. Desta forma, podemos entender a regra $\{\exists_I\}$ como uma generalização das regras de introdução da disjunção, já que para provar $\bigvee_{u \in C}[x \mapsto u]P(x)$, basta encontrar um valor $a \in C$ que torne $P(a)$ verdadeiro.

Caso o valor a seja uma variável, esta deve ocorrer livre em $[x \mapsto a]P(x)$ e não pode ocorrer livre na conclusão de $\{\exists_I\}$. Isto é, devemos associar todas as ocorrências de a a variável x introduzida pelo quantificador existencial. A seguir apresentamos um exemplo que ilustra a utilização desta regra.

Exemplo 54. Considere a tarefa de demonstrar o sequente $\{\forall x.P(x)\} \vdash \exists x.P(x)$. Iniciamos a demonstração utilizando a regra $\{\exists_I\}$, logo, devemos mostrar que $[x \mapsto b]P(x)$, para algum valor b. A partir da hipótese $\forall x.P(x)$, podemos concluir $[x \mapsto b]P(x)$ utilizando $\{\forall_E\}$. Esta dedução é apresentada a seguir.

$$\dfrac{\dfrac{\dfrac{}{\forall x.P(x)}\{_{ID}\}}{P(b)}\{\forall_E\}}{\exists x.P(x)}\{\exists_I\}$$

■

Eliminação do quantificador existencial $\{\exists_E\}$

A regra para eliminação do quantificador existencial $\{\exists_E\}$ generaliza para um universo possivelmente infinito a regra de eliminação da disjunção. Intuitivamente, a regra $\{\vee_E\}$ especifica que se $A \vee B$ é provável e que C pode ser deduzido a partir de A e que C pode ser deduzido a partir de B, então podemos concluir C a partir destes fatos. Mais formalmente:

$$\frac{\Gamma \vdash A \vee B \quad \Gamma \cup \{A\} \vdash C \quad \Gamma \cup \{B\} \vdash C}{\Gamma \vdash C} \{\vee_E\}$$

Evidentemente, podemos estender essa regra para disjunções envolvendo 3 termos de maneira quase que imediata:

$$\frac{\Gamma \vdash A \vee B \vee C \quad \Gamma \cup \{A\} \vdash D \quad \Gamma \cup \{B\} \vdash D \quad \Gamma \cup \{C\} \vdash D}{\Gamma \vdash D} \{\vee_E\}_3$$

Note que ao generalizarmos a regra de eliminação para 3 elementos, adicionamos uma nova premissa: $\Gamma \cup \{C\} \vdash D$ para que seja possível deduzir D. Desta forma, para concluir uma fórmula α a partir de uma disjunção de n fórmulas, devemos provar α a partir da suposição de cada uma da subfórmulas que formam a disjunção em questão. Como $\exists x.P(x)$ pode ser considerada uma disjunção envolvendo um número possivelmente infinito de componentes, isso nos leva a seguinte questão: como provar uma conclusão α a partir de um número possivelmente infinito de componentes que devem ser supostos para concluir esta fórmula α?

A solução para este problema é adotar uma estratégia similar ao que foi feito para a regra $\{\forall_I\}$: utilizar um valor arbitrário. Para deduzir uma fórmula α a partir de $\exists x.P(x)$, utilizando a regra $\{\exists_E\}$, devemos supor $[x \mapsto y]P(x)$, em que y é um valor arbitrário ($y \notin fv(\Gamma)$). Esta regra é apresentada a seguir.

$$\frac{\exists x.P(x) \quad \Gamma \cup \{[x \mapsto y]P(x)\} \vdash \alpha \quad y \notin fv(\Gamma)}{\Gamma \vdash \alpha} \{\exists_E\}$$

É importante notar que y não pode ocorrer livre em α, pois levaria a contradições. A seguir apresentamos um exemplo desta regra.

Exemplo 55. Considere a tarefa de demonstrar o sequente

$$\{\exists x.P(x), \forall x.P(x) \to Q(x)\} \vdash \exists y.Q(y)$$

A dedução é iniciada utilizando a regra $\{\exists_E\}$. Ao utilizar esta regra, podemos introduzir a hipótese $P(k)$[5] que possibilita utilizar a introdução da implicação para finalizar a demonstração. Esta é apresentada abaixo:

$$\dfrac{\dfrac{}{\exists x.P(x)}\{_{ID}\} \qquad \dfrac{\dfrac{\dfrac{\dfrac{\forall x.P(x) \to Q(x)}{P(k) \to Q(k)}(\forall E) \quad \dfrac{}{P(k)^1}\{_{ID}\}}{Q(k)}(\to E)}{\exists y.Q(y)}(\exists I)}{\exists y.Q(y)}(\exists E)^1$$

■

Note que a única restrição aplicável à regra $\{\exists_E\}$ é a mesma que se aplica a regra $\{\forall_I\}$: a variável "arbitrária" não deve ocorrer livre no conjunto de hipóteses.

Restrições sobre as regras do quantificador existencial

Nesta seção discutiremos a restrição sobre a regra $\{\exists_E\}$

[5] Note que $P(k)$ é equivalente a $[x \mapsto k]P(x)$.

$$\frac{\exists x.P(x) \quad \Gamma \cup \{[x \mapsto y]P(x)\} \vdash \alpha \quad y \notin fv(\Gamma)}{\Gamma \vdash \alpha}\ \{\exists_E\}$$

que não permite que $y \in fv(\alpha)$. Permitir que este fato ocorre leva a resultados obviamente contraditórios. Como exemplo, considere a "demonstração" do sequente $\{\exists x.P(x)\} \vdash \forall x.P(x)$.

$$\dfrac{\dfrac{\overline{\exists x.P(x)}^{\,(Id)} \quad \dfrac{\overline{P(x)}^{\,(Id)}}{P(x)}^{\,1}\ \{\exists_E\}}{\forall x.P(x)}\ \{\forall_I\}}{x \notin fv(\{\exists x.P(x)\})}$$

Evidemente não deveriámos ser capazes de deduzir este sequente, visto que tal demonstração implicaria que os quantificadores existencial e universal são "idênticos". Logo, para mantermos a correção da dedução natural, devemos garantir que a variável introduzida pela regra $\{\exists_E\}$ não ocorra livre na conclusão desta regra, caso contrário, podemos incorrer em erros de dedução como o apresentado no exemplo anterior.

3.9 Exercícios

1. Prove os seguintes sequentes usando dedução natural:
   (a) $\{\forall x.(P(x) \to Q(x))\} \vdash (\forall x.\neg Q(x)) \to (\forall x.\neg P(x))$
   (b) $\{\forall x.(P(x) \to \neg Q(x))\} \vdash \neg(\exists x.(P(x) \wedge Q(x)))$
   (c) $\{\forall x.(A(x) \to (B(x) \vee C(x))), \forall x.\neg B(x)\} \vdash (\forall x.A(x)) \to (\forall x.C(x))$
   (d) $\{\exists x.(P(x) \wedge Q(x)), \forall x.(P(x) \to R(x))\} \vdash \exists x.(R(x) \wedge Q(x))$
   (e) $\{\forall x.P(a,x,x), \forall x.\forall y.\forall z.P(x,y,z) \to P(f(x),y,f(z))\} \vdash P(f(a),a,f(a))$
   (f) $\{\forall x.P(x) \to Q(x)\} \vdash \forall x.P(x) \to \forall x.Q(x)$
   (g) $\{\exists x.\neg P(x)\} \vdash \neg\forall x.P(x)$

3.10 Equivalências algébricas para lógica de predicados

Assim como na dedução natural, todas as leis algébricas já vistas para lógica proposicional continuam válidas para lógica de predicados. O que faremos é apenas incluir novas leis para a manipulação adequada dos quantificadores universal e existencial.

As leis algébricas para manipulação dos quantificadores são apresentadas abaixo:

$\neg\forall x.P(x)$	$\equiv$	$\exists x.\neg P(x)$	$\{\neg - \forall\}$
$\neg\exists x.P(x)$	$\equiv$	$\forall x.\neg P(x)$	$\{\neg - \exists\}$
$\forall x.P(x) \wedge Q(x)$	$\equiv$	$\forall x.P(x) \wedge \forall x.Q(x)$	$\{\wedge - \forall\}$
$\exists x.P(x) \vee Q(x)$	$\equiv$	$\exists x.P(x) \vee \exists x.Q(x)$	$\{\vee - \exists\}$

As primeiras duas leis expressam a relação dos quantificadores com a negação lógica. O leitor atento deve ter percebido que estas regras são uma generalização das leis de DeMorgan para lógica proposicional. O segundo grupo de

regras expressa como os quantificadores universal e existencial distribuem sobre a conjunção e disjunção, respectivamente.

Exemplo 56. As fórmulas $\forall x.F(x) \wedge \neg G(x)$ e $\forall x.F(x) \wedge \neg\exists x.G(x)$ são equivalentes, conforme a dedução a seguir:

$$\begin{array}{lll} \forall x.(F(x) \wedge \neg G(x)) & = & \\ \forall x.F(x) \wedge \forall x.\neg G(x) & = & \{\wedge - \forall\} \\ \forall x.F(x) \wedge \neg\exists x.G(x) & & \{\neg - \forall\} \end{array}$$

■

3.11 Exercícios

1. Prove as seguintes equivalências utilizando regras algébricas para lógica de predicados.

   (a) $\forall x.P(x) \rightarrow \neg Q(x) \equiv \neg\exists x.P(x) \wedge Q(x)$

   (b) $\neg\forall x.\exists y.R(x,y) \wedge \neg P(x,y) \equiv \exists x.\forall y.R(x,y) \rightarrow P(x,y)$

3.12 Considerações meta-matemáticas

Nesta seção consideraremos, sem demonstração, algumas propriedades meta-matemáticas da lógica de predicados, a saber: corretude, completude e decidibilidade. Assim como na lógica proposicional, a dedução natural para lógica de predicados é um sistema formal correto e completo. Porém, a teoria associada a noção de satisfazibilidade da lógica de predicados não é decidível.

3.12.1 Correção e Completude

Nesta seção enunciaremos teoremas que afirmam que o sistema de dedução natural para lógica de predicados é correto e completo com respeito a noção de consequência lógica.

Teorema 4 (Correção da dedução natural)**.** *Seja α uma fórmula bem formada qualquer da lógica de predicados. Se $\vdash \alpha$, então $\models \alpha$.*

Teorema 5 (Completude da dedução natural)**.** *Seja α uma fórmula bem formada qualquer da lógica de predicados. Se $\models \alpha$, então $\vdash \alpha$.*

A prova da correção da dedução natural para lógica de predicados possui uma estrutura similar à demonstração para lógica proposicional. Basta utilizar indução sobre a estrutura das derivações de provas. Porém, a demonstração da propriedade de completude exige técnicas que vão além do objetivo deste texto.

3.12.2 Decidibilidade

Conforme apresentamos no capítulo 2, a teoria associada a linguagem da lógica proposicional é decidível, isto é, existe um algoritmo que responde "sim" sempre que a fórmula em questão for válida (tautologia) e "não", caso contrário.

Na seção anterior, apresentamos que a lógica de predicados possui as propriedades de correção e completude, como a lógica proposicional. Desta forma, podemos perguntar se a lógica de predicados também possui uma teoria decidível associada. Normalmente, para a lógica de predicados, considera-se a teoria que envolve o conjunto de fórmulas bem formadas desta lógica e a noção de satisfazibilidade como conceito de validade. Infelizmente, o problema de determinar se uma fórmula *arbitrária* da lógica de predicados é satisfazível é indecidível, isto é, não existe um algoritmo capaz de apresentar uma resposta correta para toda fórmula bem formada desta lógica. A demonstração deste resultado pode ser encontrada em livros que abordam teoria de computabilidade e está fora do escopo deste texto.

3.13 Notas Bibliográficas

4

Demonstração de Teoremas

> A matemática não é uma ciência dedutiva — isto é um clichê. Quando você tenta provar um teorema, você não apenas lista as hipóteses, e começa a dedução. O que normalmente fazemos é fazer uso de experimentação e tentativa e erro.
>
> Paul Richard Halmos,
> Matemático.

4.1 Motivação

Nos capítulos 2 e 3 foram apresentadas as lógicas proposicional e de predicados. Para cada uma destas lógicas, estudamos sua sintaxe, semântica e como verificar consequências lógicas utilizando dedução natural. Neste capítulo, apresentaremos uma importante aplicação de tudo que foi visto até o presente momento: usar estas lógicas para demonstrar teoremas matemáticos.

Mas qual a importância do uso de demonstrações em computação? A única tecnologia conhecida para garantir a ausência de erros em programas de computador é provando que este não possui erros. Evidentemente, isso requer a modelagem de programas em algum formalismo matemático adequado para esta tarefa, o que está fora do escopo deste texto. Porém, técnicas elementares de demonstração de teoremas são a "base" para a formalização e verificação de sistemas computacionais. Logo, é importante que todo estudante de computação saiba construir e entender demonstrações formais.

4.2 Introdução

Damos o nome de *teorema* a uma sentença matemática que é verdadeira e pode ser verificada como tal. Teoremas são compostos por um conjunto, possivelmente vazio, de sentenças, denominadas hipóteses (ou premissas), que são assumidas como verdadeiras *a priori* e uma conclusão. Normalmente, teoremas são

expressos utilizando variáveis possivelmente livres. Damos o nome de *instância* de um teorema a uma particular atribuição de valores às variáveis de um teorema. A *prova* ou *demonstração* de um teorema consiste de uma verificação que mostra que o teorema em questão é verdadeiro, para todas as possíveis instâncias deste. Note que um teorema só pode ser considerado como válido se este o for para todas suas instâncias. Para mostrar que um "teorema"[1] é inválido basta apresentar uma instância que torna o enunciado deste falso. Damos o nome de *contra-exemplo* a uma instância que torna uma sentença falsa.

A seguir apresentamos um exemplo que ilustra os conceitos apresentados no parágrafo anterior.

Exemplo 57. Considere a seguinte sentença matemática:

Sejam x, y dois números reais tais que $x > 3$ e $y < 2$. Então, $x^2 - 2y > 5$.

Esta sentença é um teorema e sua prova será apresentada em um exemplo posterior. Como esta é um teorema, ela deverá ser composta por um conjunto de hipóteses e uma conclusão. Note que o enunciado deste teorema assume que $x, y \in \mathbb{R}$ e que $x > 3$ e $y < 2$. Logo, estas são as suas hipóteses. A conclusão deste teorema é que a desigualdade $x^2 - 2y > 5$ deve ser verdadeira. Como um exemplo de uma possível instância desse teorema são $x = 4$ e $y = 0$ que tornam a desigualdade $16 - 2.0 > 5$ verdadeira. Evidentemente, caso $x = 3$ e $y = 2$ (violando, assim, as hipóteses $x > 3$ e $y < 2$) temos que a conclusão é falsa pois, $9 - 4 = 5 \not> 5$.

Agora, como exemplo de uma sentença inválida, considere:

Sejam x, y dois números reais tais que $x > 3$. Então, $x^2 - 2y > 5$.

Esta sentença não pode ser considerada um teorema por possuir um contra-exemplo. Seja $x = 4$ e $y = 6$. Temos que $x = 4 > 3$, mas $16 - 12 \not> 5$, o que torna falsa a sentença em questão.

É importante ter em mente que para demonstrar um teorema devemos construir uma prova (dedução) de que este é correto para todas as suas instâncias. Se quisermos mostrar que uma sentença é falsa, basta apresentar um contra-exemplo. ■

Você deve ter percebido que teoremas possuem a mesma estrutura de sequentes da dedução natural. Na verdade, todos os sequentes que demonstramos em capítulos anteriores, são teoremas! Neste capítulo, utilizaremos a dedução natural para demonstrar a validade de sentenças quaisquer da matemática. Porém, ao invés de utilizarmos uma notação hierárquica (em forma de uma árvore), como fizemos com a dedução natural, utilizaremos uma notação *estruturada*, no sentido que organizaremos demonstrações em blocos, similares à blocos de comandos presentes na maioria das linguagens existentes (como C/C++, Java, Python, etc.), visando facilitar a construção e o entendimento de provas.

4.3 Técnicas de Demonstração de Teoremas

Nesta seção apresentaremos as técnicas para demonstração de teoremas, que essencialmente, são as regras já vistas em nosso estudo de dedução natural.

[1] Note que uma sentença só pode ser considerada um teorema se esta for verdadeira. Afirmar que um teorema é falso é apenas um abuso de linguagem utilizado para facilitar a exposição deste conceito.

Visando facilitar a tarefa de construir demonstrações corretas e similares às encontradas em textos de matemática, dividiremos a tarefa de provar um teorema em duas partes relacionadas: 1) construção de um rascunho e 2) elaboração de um texto, a partir deste rascunho[2].

O rascunho é utilizado para realizar as deduções que formam a demonstração de um teorema. Normalmente este é dividido em duas colunas: a coluna de hipóteses e a de conclusão. Na coluna de hipóteses encontram-se todas as hipóteses e suposições feitas durante a demonstração e a coluna de conclusão registra qual fórmula devemos deduzir a partir das hipóteses, para estabelecer que o teorema em questão é realmente verdadeiro.

Mas, como construir um rascunho para um dado teorema? Como produzir um texto a partir deste rascunho? Ambas estas perguntas são respondidas considerando o que [9] chama de *estratégia de prova*. Uma estratégia de prova consiste de modelos para construção de rascunho e de textos que são aplicáveis a um certo tipo de hipótese ou conclusão. Veremos que a escolha de qual estratégia de prova será utilizada depende de quais conectivos / quantificadores a fórmula em questão possui.

Denominamos de *estratégia para utilização de hipóteses*, técnicas que permitenos deduzir novas fórmulas a partir de hipóteses. Estas estratégias correspondem às regras de eliminação de quantificadores e conectivos da dedução natural. Por sua vez, denominamos de *estratégia para demonstrar uma conclusão* técnicas que nos permitem deduzir uma fórmula com um certo conectivo / quantificador. Usualmente, estas técnicas permitem transformar o problema de demonstrar uma fórmula α em problemas mais simples. Estratégias para demonstrar conclusões correspondem às regras de introdução de conectivos / quantificadores da dedução natural.

Os modelos de rascunho presentes em uma estratégia de demonstração dividem-se em duas partes, a primeira mostra um esquema de rascunho antes de usar a estratégia e a segunda mostra o rascunho resultante. Já o modelo de texto, usualmente apresenta "buracos" a serem preenchidos com o texto de alguma sub-demonstração a ser realizada. Partes a serem preenchidas com sub-demonstrações são indicadas usando colchetes ('[' e ']').

A demonstração de um teorema deve seguir os seguintes passos:

1. Identifique as hipóteses e conclusão de um teorema e expresse-os como fórmulas da lógica.

2. A partir da representação do teorema como um conjunto de fórmulas da lógica, construa o rascunho que demonstra o teorema em questão.

3. A partir do rascunho produzido, elabore o texto final da demonstração.

As próximas seções apresentam cada uma das técnicas para os conectivos e quantificadores e exemplos que ilustram o uso destas estratégias.

4.3.1 Estratégias para Implicação ($\rightarrow$)

A primeira estratégia de demonstração que veremos é provavelmente a que será mais utilizada. Esta permite demonstrar implicações lógicas e é equivalente à

[2] A técnica que adotaremos neste texto para construção de demonstrações é a apresentada no livro de Daniel Velleman [9].

regra de introdução da implicação da dedução natural e é conhecida como prova direta.

Estratégia de Prova 1 (Para provar uma conclusão da forma $\alpha \to \beta$)**.** Suponha que α é verdadeiro e então prove β.

Rascunho.

Rascunho antes de usar a estratégia.

Hipóteses	Conclusão
$\gamma_1, \gamma_2, ..., \gamma_n$	$\alpha \to \beta$

Rascunho depois de usar a estratégia.

Hipóteses	Conclusão
$\gamma_1, \gamma_2, ..., \gamma_n$	β
α	

Texto:
Suponha que α.
[*Prova de* β]
Portanto, se α *então* β.

■

O exemplo a seguir utiliza esta estratégia para construção de um teorema simples.

Exemplo 58. Considere a tarefa de demonstrar o seguinte teorema:

Suponha que $a, b \in \mathbb{R}$. Se $0 < a < b$ então $a^2 < b^2$.

Seguindo os passos descritos anteriormente, primeiro devemos representar as hipóteses e a conclusão deste teorema como fórmulas da lógica. O teorema em questão possui como hipóteses que $a, b \in \mathbb{R}$ e a sua conclusão é a fórmula:

$$0 < a < b \to a^2 < b^2$$

A partir da representação das hipóteses e da conclusão como fórmulas da lógica, devemos proceder com a elaboração do rascunho. Inicialmente, o rascunho possui a seguinte forma:

Hipóteses	Conclusão
$a, b \in \mathbb{R}$	$0 < a < b \to a^2 < b^2$

Uma vez que a conclusão é uma implicação, podemos utilizar a estratégia de provas 1. Abaixo é apresentado o rascunho após a utilização desta estratégia:

Hipóteses	Conclusão
$a, b \in \mathbb{R}$	$a^2 < b^2$
$0 < a < b$	

Evidentemente, se $0 < a < b$ então $a > 0$, $a < b$, $b > 0$. Como tanto a quanto b são maiores que zero, podemos multiplicar ambos os lados de $a < b$ por cada um destes valores. Multiplicando por a, obtemos $a^2 < ab$, e ao multiplicarmos por b, obtemos $ab < b^2$.

Hipóteses	Conclusão
$a, b \in \mathbb{R}$	$a^2 < b^2$
$0 < a < b$	
$a^2 < ab$	
$ab < b^2$	

Uma vez que $a^2 < ab$ e $ab < b^2$, temos que $a^2 < b^2$, como queríamos demonstrar.

Como conseguimos deduzir a conclusão a partir das hipóteses, utilizando o rascunho, devemos proceder para a elaboração do texto final da prova. De acordo com o modelo de texto descrito na estratégia de prova 1, o texto deve possuir a seguinte estrutura inicial:

Suponha que $a, b \in \mathbb{R}$. Suponha que $0 < a < b$.
  [Prova de $a^2 < b^2$]
Portanto, se $0 < a < b$ então $a^2 < b^2$.

Para finalizar o texto, basta preencher o "buraco" com o texto da dedução de $a^2 < b^2$. O resultado final é apresentado a seguir.

Suponha que $a, b \in \mathbb{R}$. Suponha que $0 < a < b$.
  Como $0 < a < b$, temos que $a, b > 0$ e $a < b$.
  Como $a > 0$ e $a < b$, temos que $a^2 < ab$.
  Como $b > 0$ e $a < b$, temos que $ab < b^2$.
  Como $a^2 < ab$ e $ab < b^2$, temos que $a^2 < b^2$.
Portanto, se $0 < a < b$ então $a^2 < b^2$.

■

Note que a estrutura da demonstração é indicada utilizando indentação, de maneira similar a blocos de comandos em linguagens de programação. Apesar de não ser uma padrão em textos sobre matemática, há evidências que a utilização de indentação em provas ajuda no entendimento[3] e, por isso, este será o padrão adotado neste texto.

Outra maneira de demonstrar uma implicação é a utilização da seguinte equivalência lógica: $\alpha \to \beta \equiv \neg\beta \to \neg\alpha$, que é facilmente demonstrável utilizando álgebra booleana. Demonstrações de implicações baseadas nesta estratégia são comumente denominadas de provas pela contrapositiva. A próxima estratégia de prova é baseada nesta equivalência.

Estratégia de Prova 2 (Para provar uma conclusão da forma $\alpha \to \beta$)**.** Suponha que β é falso e prove que α é falso.

Rascunho.

Rascunho antes de usar a estratégia.

Hipóteses	Conclusão
$\gamma_1, \gamma_2, ..., \gamma_n$	$\alpha \to \beta$

Rascunho depois de usar a estratégia.

Hipóteses	Conclusão
$\gamma_1, \gamma_2, ..., \gamma_n$	$\neg\alpha$
$\neg\beta$	

[3] Uma argumentação detalhada a favor do uso de provas estruturadas é apresentada em [5].

Texto:
Suponha que β é falso.
    [*Prova de* $\neg\alpha$]
Portanto, se α então β.

■

O próximo exemplo ilustra o uso da estratégia de prova 2 para demonstrar um teorema simples.

Exemplo 59. Considere a tarefa de demonstrar o seguinte teorema:

Suponha que a, b e c são números reais tais que $a > b$. Se $ac \leq bc$ então $c \leq 0$.

Para demonstrar este teorema, primeiramente, devemos representar suas hipóteses e conclusão como fórmulas da lógica. Evidentemente, as hipóteses deste teorema são que $a, b, c \in \mathbb{R}$, $a > b$ e a sua conclusão é representada pela seguinte fórmula:

$$ac \leq bc \rightarrow c \leq 0$$

A partir da representação das hipóteses e conclusão devemos proceder para a construção do rascunho.

Hipóteses	Conclusão
$a, b, c \in \mathbb{R}$	$ac \leq bc \rightarrow c \leq 0$
$a > b$	

Como a conclusão deste teorema é formada por uma implicação, utilizaremos a estratégia de prova 2 para demonstrá-la. O resultado de se usar esta técnica de prova é apresentado no rascunho a seguir.

Hipóteses	Conclusão
$a, b, c \in \mathbb{R}$	$\neg(ac \leq bc)$
$a > b$	
$\neg(c \leq 0)$	

É óbvio que $\neg(ac \leq bc) \equiv ac > bc$ e que $\neg(c \leq 0) \equiv c > 0$.

Hipóteses	Conclusão
$a, b, c \in \mathbb{R}$	$ac > bc$
$a > b$	
$c > 0$	

Mas como $c > 0$, podemos multiplicar ambos os lados de $a > b$ obtendo $ac > bc$, o que conclui a demonstração deste teorema.

Após a conclusão do rascunho, devemos proceder com a elaboração do texto desta demonstração. De acordo com a estratégia 2, temos a seguinte estrutura inicial:

Sejam $a, b, c \in \mathbb{R}$ tais que $a > b$.
Suponha que $c > 0$.
    [Prova de $ac > bc$].
Portanto, se $ac \leq bc$ então $c \leq 0$.

Que é imediatamente encerrada com a dedução de que $ac > bc$, a partir das hipóteses $a > b$ e $c > 0$, conforme apresentado a seguir.

Sejam $a, b, c \in \mathbb{R}$ tais que $a > b$.
Suponha que $c > 0$.
Como $a > b$ e $c > 0$ temos que $ac > bc$.
Portanto, se $ac \leq bc$ então $c \leq 0$.

■

Note que este teorema pode ser provado usando a estratégia de prova 1. Isso mostra que, muitas vezes, há mais de uma possível estratégia aplicável a demonstração de um certo teorema. Porém, certamente, uma das possibilidades resultará em uma prova mais simples, como o caso do exemplo anterior. Apesar de demonstrável usando uma prova direta, o uso de contrapositiva permitiu uma prova quase que imediata.

As próximas seções deste capítulo apresentarão técnicas e exemplos de utilização destas para demonstração de teoremas envolvendo outros conectivos / quantificadores.

4.3.2 Estratégias para Negação ($\neg$) e Implicação ($\rightarrow$)

Para provar que uma dada conclusão é falsa (isto é, provar $\neg\alpha$), devemos proceder de maneira similar ao que era feito na dedução natural: tratar a negação como uma implicação (lembre-se $\neg\alpha \equiv \alpha \rightarrow \bot$) e demonstrar uma contradição. Esta idéia é formalizada pela próxima estratégia de prova.

Estratégia de Prova 3 (Para provar uma conclusão da forma $\neg\alpha$)**.** Suponha que α é verdadeiro e obtenha uma contradição.

Rascunho.

Rascunho antes de usar a estratégia.

Hipóteses	Conclusão
$\gamma_1, \gamma_2, ..., \gamma_n$	$\neg\alpha$

Rascunho depois de usar a estratégia.

Hipóteses	Conclusão
$\gamma_1, \gamma_2, ..., \gamma_n$	$\bot$
α	

Texto:
Suponha que α é verdadeiro.
[*Prova de* $\bot$]
Portanto, α é falso.

■

Exemplo 60. Considere o seguinte teorema:

Se $x^2 + y = 13$ e $y \neq 4$ então $x \neq 3$.

Note que este teorema não possui hipóteses e sua conclusão é uma fórmula que possui o conectivo de implicação, conforme apresentado a seguir:

$$x^2 + y = 13 \wedge y \neq 4 \rightarrow x \neq 3$$

Hipóteses	Conclusão
	$x^2 + y = 13 \wedge y \neq 4 \rightarrow x \neq 3$

Utilizando a estratégia de prova direta, temos:

Hipóteses	Conclusão
$x^2 + y = 13$	$x \neq 3$
$y \neq 4$	

Como a conclusão é uma negação ($x \neq 3 \equiv \neg(x = 3)$), podemos utilizar a estratégia de prova 3 obtendo:

Hipóteses	Conclusão
$x^2 + y = 13$	$\perp$
$y \neq 4$	
$x = 3$	

Porém, ao substituir $x = 3$ em $x^2 + y = 13$ obtemos $y = 4$, o que contradiz a suposição de que $y\neg 4$, o que conclui a demonstração do teorema.

Agora, a partir do rascunho, basta construir o texto utilizando os modelos para as estratégias de prova utilizadas. Primeiramente, usando o modelo de texto para prova direta, obtemos:

Suponha que $x^2 + y = 13$ e que $y \neq 4$.
  [Prova de $x \neq 3$]
Portanto, se $x^2 + y = 13$ e $y \neq 4$ então $x \neq 3$.

Na sequência, utilizamos o modelo para negação:

Suponha que $x^2 + y = 13$ e que $y \neq 4$.
  Suponha que $x = 3$.
    [Prova de $\perp$]
  Logo, $x \neq 3$.
Portanto, se $x^2 + y = 13$ e $y \neq 4$ então $x \neq 3$.

Finalmente, encerramos a demonstração apresentando a contradição obtida a partir das hipóteses.

Suponha que $x^2 + y = 13$ e que $y \neq 4$.
  Suponha que $x = 3$.
    Como $x^2 + y = 13$ e $y \neq 4$, temos que $y = 4$.
    Como $y \neq 4$ e $y = 4$, temos uma contradição.
  Logo, $x \neq 3$.
Portanto, se $x^2 + y = 13$ e $y \neq 4$ então $x \neq 3$.

■

Sentenças negativas são, usualmente, mais difíceis de provar que positivas. Isto motiva uma técnica para demonstração que é bem útil e vale-se de equivalências algébricas da lógica.

Estratégia de Prova 4 (Para provar uma conclusão da forma $\neg\alpha$)**.** Tente reexpressá-la como uma fórmula sem negação utilizando equivalências da álgebra booleana e então utilize outras estratégias de prova. ■

Como um exemplo desta estratégia, considere a seguinte variação do exemplo 60.

Exemplo 61. Sejam $x, y \in \mathbb{N}$. Se $x^2 + y = 13$ então não é verdade que $x \neq 3$ e $y = 4$.

É fácil perceber que este teorema é composto apenas por uma conclusão e que esta é representada pela seguinte fórmula:

$$x^2 + y = 13 \rightarrow \neg(x \neq 3 \wedge y = 4)$$

O rascunho para este teorema possui a seguinte configuração inicial:

Hipóteses	Conclusão
$x, y \in \mathbb{N}$	$x^2 + y = 13 \rightarrow \neg(x \neq 3 \wedge y = 4)$

Como a conclusão é uma implicação, podemos iniciar a demonstração usando a técnica de prova direta.

Hipóteses	Conclusão
$x, y \in \mathbb{N}$	$\neg(x \neq 3 \wedge y = 4)$
$x^2 + y = 13$	

Note que a conclusão possui uma negação. Logo, podemos então tentar a estratégia de prova 4 e usar álgebra booleana para mudar a forma da conclusão. Note que $\neg(x \neq 3 \wedge y = 4)$ é equivalente a $y = 4 \rightarrow x = 3$, conforme a dedução seguinte:

$$\begin{array}{ll} \neg(x \neq 3 \wedge y = 4) & \equiv \\ x = 3 \vee y \neq 4 & \equiv \\ y \neq 4 \vee x = 3 & \equiv \\ y = 4 \rightarrow x = 3 & \end{array}$$

Usando esta equivalência, temos que o rascunho pode ser alterado para:

Hipóteses	Conclusão
$x, y \in \mathbb{N}$	$y = 4 \rightarrow x = 3$
$x^2 + y = 13$	

Logo, podemos utilizar novamente uma estratégia de prova direta, obtendo:

Hipóteses	Conclusão
$x, y \in \mathbb{N}$	$x = 3$
$x^2 + y = 13$	
$y = 4$	

O que é evidentemente verdadeiro. A seguir, apresentamos a construção passo-a-passo do texto desta demonstração.

Suponha que $x^2 + y = 13$ e que $y = 4$.
Como $y = 4$ e $x^2 + y = 13$, temos que $x = 3$.
Portanto, se $x^2 + y = 13$ então não é verdade que $x \neq 3$ e $y = 4$.

Note que, a manipulação algébrica que transformou a negação em uma implicação não é sequer citada no texto da demonstração. Manipulações algébricas sobre fórmulas da lógica devem apenas fazer parte do rascunho, nunca da demonstração final de um teorema. ■

Até o presente momento foram apresentadas apenas estratégias para demonstrar teoremas que possuem um certo conectivo. Um ponto chave na demonstração de teoremas é a utilização adequada de hipóteses. As próximas estratégias a serem apresentadas mostram como utilizar hipóteses e por isso, são chamadas de estratégias para uso de hipóteses.

Estratégia de Uso de Hipóteses 1 (Para usar uma hipótese da forma $\neg\alpha$)**.** Caso possível, reexpresse $\neg\alpha$ utilizando regras da álgebra booleana de maneira que a negação seja removida desta fórmula. ■

Estratégia de Uso de Hipóteses 2 (Para usar uma hipótese da forma $\alpha \to \beta$)**.** Caso seja possível deduzir α ou $\neg\beta$, então podemos utilizar $\alpha \to \beta$ para deduzir β ou $\neg\alpha$. Note que esta estratégia é equivalente a utilizar a regra $\{\to\ _E\}$ ou o seguinte sequente da dedução natural $\{\alpha \to \beta, \neg\beta\} \vdash \alpha$. ■

Exemplo 62. Suponha que $P \to Q \to R$. Então, $\neg R \to (P \to \neg Q)$.

Como este teorema envolve fórmulas da lógica diretamente, podemos proceder para a construção do rascunho.

Hipóteses	Conclusão
$P \to Q \to R$	$\neg R \to (P \to \neg Q)$

Como desejamos concluir uma implicação, vamos iniciar esta demonstração usando a técnica de prova direta.

Hipóteses	Conclusão
$P \to Q \to R$	$P \to \neg Q$
$\neg R$	

Novamente, usando prova direta temos:

Hipóteses	Conclusão
$P \to Q \to R$	$\neg Q$
$\neg R$	
P	

Como possuímos P e $P \to Q \to R$, podemos utilizar a estratégia de uso de hipóteses 2 para concluir $Q \to R$.

Hipóteses	Conclusão
$P \to Q \to R$	$\neg Q$
$\neg R$	
P	
$Q \to R$	

Como $Q \to R$ e $\neg R$ são verdadeiras, temos que $\neg Q$ também o é, terminando assim, a dedução. A seguir apresentamos o texto desta demonstração.

Suponha que $P \to Q \to R$.
  Suponha que $\neg R$.
    Suponha que P.
      Como $P \to Q \to R$ e P, temos que $Q \to R$.
      Como $Q \to R$ e $\neg R$, temos que $\neg Q$.
    Logo, $P \to \neg Q$.
  Assim, $\neg R \to (P \to \neg Q)$.
Portanto, $\neg R \to (P \to \neg Q)$.

■

4.3.3 Exercícios

1. Prove os seguintes teoremas.
   (a) Suponha $a, b \in \mathbb{R}$. Se $a < b < 0$ então $a^2 > b^2$.
   (b) Suponha $a, b \in \mathbb{R}$. Se $0 < a < b$ então $\frac{1}{b} < \frac{1}{a}$.
   (c) Suponha $a, b, c, d \in \mathbb{R}$, $0 < a < b$ e $d > 0$. Se $ac \geq bd$ então $c > d$.
   (d) Suponha que $a, b \in \mathbb{R}$. Se $a^2 b = 2a + b$, então se $b \neq 0$ então $a \neq 0$.

4.3.4 Estratégias para Quantificadores ($\forall$), ($\exists$)

Estratégias de prova para conclusões envolvendo quantificadores são análogas às regras de introdução destes apresentadas nos capítulos 2 e 3.

Estratégia de Prova 5 (Para provar uma conclusão da forma $\forall x.P(x)$)**.** Suponha que x é um valor arbitrário[4] e prove $P(x)$.

Rascunho.

Rascunho antes de usar a estratégia.

Hipóteses	Conclusão
$\gamma_1, \gamma_2, ..., \gamma_n$	$\forall x.P(x)$

Rascunho depois de usar a estratégia.

Hipóteses	Conclusão
$\gamma_1, \gamma_2, ..., \gamma_n$	$P(x)$
x arbitrário	

Texto:
Suponha que x é arbitrário.
    [*Prova de $P(x)$*]
Portanto, $\forall x.P(x)$.

■

Estratégia de Prova 6 (Para provar uma conclusão da forma $\exists x.P(x)$)**.** Tente encontrar o valor de x que torna $P(x)$ verdadeiro e então prove esta fórmula.

Rascunho.

Rascunho antes de usar a estratégia.

Hipóteses	Conclusão
$\gamma_1, \gamma_2, ..., \gamma_n$	$\exists x.P(x)$

Rascunho depois de usar a estratégia.

Hipóteses	Conclusão
$\gamma_1, \gamma_2, ..., \gamma_n$	$P(x)$
$x =$[valor escolhido por você.]	

Texto:
Seja $x =$[valor escolhido por você].
    [*Prova de $P(x)$*]
Portanto, $\exists x.P(x)$.

[4]Lembre-se: um valor x é arbitrário se este não pertence ao conjunto de variáveis livres das hipóteses do teorema em questão.

■

A seguir, apresentamos um exemplo que ilustra a utilização destas duas estratégias de prova.

Exemplo 63. Considere o seguinte teorema:

Para todo $x \in \mathbb{R}$, se $x > 0$ então existe um $y \in \mathbb{R}$ tal que $y(y+1) = x$.

Este teorema é formado apenas pela conclusão, expressa simbolicamente a seguir:

$$\forall x.x \in \mathbb{R} \rightarrow x > 0 \rightarrow \exists y.y \in \mathbb{R} \wedge y(y+1) = x$$

A partir da representação deste teorema, podemos iniciar a construção de seu rascunho:

Hipóteses	Conclusão
	$\forall x.x \in \mathbb{R} \rightarrow x > 0 \rightarrow \exists y.y \in \mathbb{R} \wedge y(y+1) = x$

Como a conclusão é uma fórmula envolvendo o quantificador universal, podemos utilizar a estratégia de prova 5, obtendo:

Hipóteses	Conclusão
x arbitrário	$x \in \mathbb{R} \rightarrow x > 0 \rightarrow \exists y.y \in \mathbb{R} \wedge y(y+1) = x$

Utilizando a estratégia de prova direta (duas vezes) obtemos:

Hipóteses	Conclusão
x arbitrário	$\exists y.y \in \mathbb{R} \wedge y(y+1) = x$
$x \in \mathbb{R}$	
$x > 0$	

Agora, temos que mostrar que existe um valor $y \in \mathbb{R}$ tal que $y(y+1) = x$. Mas qual seria este valor? Olhando com um pouco de atenção a equação $y(y+1) = x$, podemos perceber que esta é uma equação de 2^o grau sobre a variável y. Resolvendo-a obtemos:

$$\begin{array}{lcl} \Delta & = & 1 - 4.1.(-x) \\ y' & = & \frac{-1+\sqrt{1+4x}}{2} \\ y'' & = & \frac{-1-\sqrt{1+4x}}{2} \end{array}$$

Desta forma, temos que tanto $\frac{-1+\sqrt{1+4x}}{2}$ quanto $\frac{-1-\sqrt{1+4x}}{2}$ são possíveis valores para y que tornam a equação $y(y+1) = x$ verdadeira, conforme demonstrado a seguir:

$$\begin{array}{lcl} y(y+1) & = & \{\text{por } y = \frac{-1+\sqrt{1+4x}}{2}\} \\ \frac{-1+\sqrt{1+4x}}{2}\left(\frac{-1+\sqrt{1+4x}}{2}+1\right) & = & \\ \frac{-1+\sqrt{1+4x}}{2}\left(\frac{-1+\sqrt{1+4x}+2}{2}\right) & = & \\ \frac{(-1+\sqrt{1+4x})}{2}\frac{(1+\sqrt{1+4x})}{2} & = & \\ \frac{1+4x-1}{4} & = & \\ \frac{4x}{4} & = & \\ x & & \end{array}$$

Logo, para $y = \frac{-1+\sqrt{1+4x}}{2}$, temos que $y(y+1) = x$, o que conclui a demonstração deste teorema. Abaixo, apresentamos passo-a-passo a construção do texto a partir do rascunho. Primeiramente, o texto para o uso da técnica de provas para o quantificador universal:

Suponha x arbitrário.
  [Prova de $x \in \mathbb{R} \rightarrow x > 0 \rightarrow \exists y.y \in \mathbb{R} \wedge y(y+1) = x$]
Como x é arbitrário temos que se $x > 0$ então existe y tal que $y(y+1) = x$.

Na sequência, o texto é alterado para refletir o uso da técnica de prova direta.

Suponha x arbitrário.
  Suponha que $x \in \mathbb{R}$ e $x > 0$.
    [Prova de $\exists y.y \in \mathbb{R} \wedge y(y+1) = x$]
  Logo, se $x \in \mathbb{R}$ e $x > 0$ então existe y tal que $y(y+1) = x$
Como x é arbitrário temos que se $x > 0$ então existe y tal que $y(y+1) = x$.

Agora, resta demonstrar o quantificador existencial da conclusão:

Suponha x arbitrário.
  Suponha que $x \in \mathbb{R}$ e $x > 0$.
    Seja $y = \frac{-1+\sqrt{1+4x}}{2}$
      [Prova de que $y(y+1) = x$]
    Logo, existe y tal que $y(y+1) = x$.
  Logo, se $x \in \mathbb{R}$ e $x > 0$ então existe y tal que $y(y+1) = x$
Como x é arbitrário temos que se $x > 0$ então existe y tal que $y(y+1) = x$.

Para encerrarmos a demonstração, basta utilizar o desenvolvimento algébrico apresentado anteriormente.

Suponha x arbitrário.
  Suponha que $x \in \mathbb{R}$ e $x > 0$.
    Seja $y = \frac{-1+\sqrt{1+4x}}{2}$

$$
\begin{array}{lcl}
y(y+1) & = & \{\text{por } y = \frac{-1+\sqrt{1+4x}}{2}\} \\
\frac{-1+\sqrt{1+4x}}{2}\left(\frac{-1+\sqrt{1+4x}}{2}+1\right) & = & \\
\frac{-1+\sqrt{1+4x}}{2}\left(\frac{-1+\sqrt{1+4x}+2}{2}\right) & = & \\
\frac{(-1+\sqrt{1+4x})}{2}\frac{(1+\sqrt{1+4x})}{2} & = & \\
\frac{1+4x-1}{4} & = & \\
\frac{4x}{4} & = & \\
x & &
\end{array}
$$

    Logo, existe y tal que $y(y+1) = x$.
  Logo, se $x \in \mathbb{R}$ e $x > 0$ então existe y tal que $y(y+1) = x$
Como x é arbitrário temos que se $x > 0$ então existe y tal que $y(y+1) = x$.

■

Note que no texto final da demonstração de um teorema envolvendo o quantificador existencial não há explicação sobre como o valor utilizado para provar $\exists y.y(y+1) = x$ foi encontrado. Isto é uma prática padrão em matemática, já que a única coisa que estamos interessados é em mostrar que um certo valor existe e não em como obtê-lo.

As próximas estratégias de utilização de hipóteses mostram como hipóteses envolvendo os quantificadores existencial e universal podem ser utilizadas. O leitor verá que estas são exatamente as regras para eliminação para estes quantificadores.

Estratégia de Uso de Hipóteses 3 (Para utilizar uma hipótese da forma $\forall x.P(x)$)**.** Basta adicionar como hipótese $[x \mapsto a]P(x)$, em que a é um valor qualquer do universo de discurso. Note que esta estratégia é exatamente a regra de eliminação do quantificador universal.

Rascunho.

Rascunho antes de usar a estratégia.

Hipóteses	Conclusão
$\gamma_1, \gamma_2, ..., \gamma_n$	β
$\forall x.P(x)$	

Rascunho depois de usar a estratégia.

Hipóteses	Conclusão
$\gamma_1, \gamma_2, ..., \gamma_n$	β
$\forall x.P(x)$	
$[x \mapsto a]P(x)$	

■

Estratégia de Uso de Hipóteses 4 (Para utilizar uma hipótese da forma $\exists x.P(x)$)**.** Basta adicionar como hipótese $[x \mapsto x_0]P(x)$, em que x_0 é um valor arbitrário. Note que esta estratégia é exatamente a regra de eliminação do quantificador existencial.

Rascunho.

Rascunho antes de usar a estratégia.

Hipóteses	Conclusão
$\gamma_1, \gamma_2, ..., \gamma_n$	β
$\exists x.P(x)$	

Rascunho depois de usar a estratégia.

Hipóteses	Conclusão
$\gamma_1, \gamma_2, ..., \gamma_n$	β
$\exists x.P(x)$	
$[x \mapsto x_0]P(x)$	
x_0 é arbitrário	

■

Antes de apresentarmos um exemplo destas estratégias, daremos uma definição formal do conceito de divisibilidade de dois números inteiros.

Definição 33 (Divisibilidade)**.** Sejam $a, b \in \mathbb{Z}$. Dizemos que a divide b, $a \mid b$, se $\exists k.k \in \mathbb{Z} \wedge ka = b$. ■

Exemplo 64. Considere o seguinte teorema:

Para todo $a, b, c \in \mathbb{Z}$, se $a \mid b$ e $b \mid c$ então $a \mid c$.

que pode ser representado pela seguinte fórmula da lógica:

$$\forall abc.a, b, c \in \mathbb{Z} \rightarrow a \mid b \wedge b \mid c \rightarrow a \mid c$$

A configuração inicial do rascunho deste teorema é:

Hipóteses	Conclusão
	$\forall a\, b\, c. a, b, c \in \mathbb{Z} \rightarrow a \mid b \wedge b \mid c \rightarrow a \mid c$

Como a conclusão envolve um quantificador universal, utilizaremos a estratégia de prova para este quantificador.

Hipóteses	Conclusão
a, b, c são arbitrários	$a, b, c \in \mathbb{Z} \rightarrow a \mid b \wedge b \mid c \rightarrow a \mid c$

Utilizando a estratégia de prova direta (duas vezes), temos:

Hipóteses	Conclusão
a, b, c são arbitrários	$a \mid c$
$a, b, c \in \mathbb{Z}$	
$a \mid b$	
$b \mid c$	

Para continuar esta demonstração, devemos utilizar a definição 33, obtendo a seguinte configuração do rascunho:

Hipóteses	Conclusão
a, b, c são arbitrários	$\exists k. k \in \mathbb{Z} \wedge ka = c$
$a, b, c \in \mathbb{Z}$	
$\exists k_1. k_1 \in \mathbb{Z} \wedge k_1 a = b$	
$\exists k_2. k_2 \in \mathbb{Z} \wedge k_2 b = c$	

Utilizando a estratégia para utilização de hipóteses envolvendo o quantificadores existenciais, temos:

Hipóteses	Conclusão
a, b, c são arbitrários	$\exists k. k \in \mathbb{Z} \wedge ka = c$
$a, b, c \in \mathbb{Z}$	
$\exists k_1. k_1 \in \mathbb{Z} \wedge k_1 a = b$	
$\exists k_2. k_2 \in \mathbb{Z} \wedge k_2 b = c$	
$k_1 a = b$	
$k_2 b = c$	

A partir das hipóteses $k_1 a = b$ e $k_2 b = c$, temos que $c = k_1 k_2 a$. Logo, temos que o valor de k que torna a igualdade $ka = c$ é $k = k_1 k_2$.

Novamente, apresentaremos passo-a-passo a construção do texto para o rascunho apresentado.

Suponha a, b e c arbitrários.
[Prova de $a, b, c \in \mathbb{Z} \rightarrow a \mid b \wedge b \mid c \rightarrow a \mid c$]
Como a, b e c são arbitrários, temos que para todo a, b e c se $a \mid b$ e $b \mid c$ então $a \mid c$.

Agora, utilizando a estratégia de prova direta, temos a seguinte versão parcial do texto:

Suponha a, b e c arbitrários.
Suponha que $a, b, c \in \mathbb{Z}$, $a \mid b$ e $b \mid c$.
[Prova de $a \mid c$]
Logo, se $a, b, c \in \mathbb{Z}$, $a \mid b$ e $b \mid c$ então $a \mid c$.
Como a, b e c são arbitrários, temos que para todo a, b e c se $a \mid b$ e $b \mid c$ então $a \mid c$.

Utilizando a definição de divisibilidade, temos que a demonstração de $a \mid c$ envolve o uso da estratégia do quantificador existencial:

Suponha a, b e c arbitrários.
  Suponha que $a, b, c \in \mathbb{Z}$, $a \mid b$ e $b \mid c$.
    Seja $k = k_1 k_2$.
      Como $a \mid b$, temos que existe k_1 tal que $k_1 a = b$.
      Como $b \mid c$, temos que existe k_2 tal que $k_2 b = c$.
      Assim, temos que $k_1 k_2 a = c$.
    Logo, $a \mid c$.
  Logo, se $a, b, c \in \mathbb{Z}$, $a \mid b$ e $b \mid c$ então $a \mid c$.
Como a, b e c são arbitrários, temos que para todo a, b e c se $a \mid b$ e $b \mid c$ então $a \mid c$.

■

4.3.5 Exercícios

1. Prove os seguintes teoremas:

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

   (b) Suponha que $x \in \mathbb{R}$. Se $\frac{y+1}{y-2} = x$ então $x \neq 1$.

   (c) Suponha que $x \in \mathbb{R}$. Se $x > 2$ então existe y tal que $y + \frac{1}{y} = x$.

   (d) Suponha que $a, b, c \in \mathbb{Z}$. Se $a \mid b$ e $a \mid c$ então $a \mid (b + c)$.

   (e) Suponha que $a, b, c \in \mathbb{Z}$. Se $ac \mid bc$ e $c \neq 0$ então $a \mid b$.

4.3.6 Estratégias para Conjunção ($\wedge$) e Bicondicional ($\leftrightarrow$)

As estratégias de prova para a conjunção e o bicondicional refletem diretamente o significado destes conectivos.

Estratégia de Prova 7 (Para provar uma conclusão da forma $\alpha \wedge \beta$)**.** Prove α e β separadamente.

Rascunho.

Rascunho antes de usar a estratégia.

Hipóteses	Conclusão
$\gamma_1, \gamma_2, ..., \gamma_n$	$\alpha \wedge \beta$

Rascunho depois de usar a estratégia.

Hipóteses	Conclusão
$\gamma_1, \gamma_2, ..., \gamma_n$	α
	β

■

Uma conclusão da forma $\alpha \wedge \beta$ deve ser considerada como "duas" conclusões[5]: α e β. De maneira similar, tratamos hipóteses envolvendo conjunções

[5]Note que isso é um abuso de linguagem, já que a conclusão de um teorema é única.

Estratégia de Uso de Hipóteses 5 (Para usar uma hipótese da forma $\alpha \wedge \beta$)**.** Considere-a como duas hipóteses separadas: α e β. ■

Agora que vimos como manipular conjunções em provas, você já deve ser capaz de deduzir como serão as estratégias para manipulação de bicondicionais, uma vez que $\alpha \leftrightarrow \beta \equiv (\alpha \rightarrow \beta) \wedge (\beta \rightarrow \alpha)$.

Estratégia de Prova 8 (Para provar uma conclusão da forma $\alpha \leftrightarrow \beta$)**.** Prove $\alpha \rightarrow \beta$ e $\beta \rightarrow \alpha$ separadamente.

Texto:
($\rightarrow$): [Prova de $\alpha \rightarrow \beta$].
($\leftarrow$): [Prova de $\beta \rightarrow \alpha$].

■

Note que ao contrário da estratégia de provas para a conjunção, apresentamos um modelo de texto para o bicondicional. Isto se deve ao fato de que provas envolvendo este conectivo usualmente "sinalizam" qual lado da implicação está sendo demonstrado utilizando setas apropriadas.

A manipulação de hipóteses envolvendo bicondicionais é imediata.

Estratégia de Uso de Hipóteses 6 (Para usar uma hipótese da forma $\alpha \leftrightarrow \beta$)**.** Trate-a como duas hipóteses distintas: $\alpha \rightarrow \beta$ e $\beta \rightarrow \alpha$. ■

Exemplo 65. Considere o seguinte teorema:

Suponha $n \in \mathbb{Z}$. Então, n é par se e somente se n^2 é par.

Este teorema possui como hipóteses o fato de que $n \in \mathbb{Z}$ e conclusão n é par se e somente se n^2 é par, que é representada pela seguinte fórmula:

$$n \text{ é par} \leftrightarrow n^2 \text{ é par.}$$

o que nos conduz a seguinte configuração inicial do rascunho:

Hipóteses	Conclusão
$n \in \mathbb{Z}$	n é par $\leftrightarrow$ n^2 é par

Utilizando a estratégia de provas para o conectivo bicondicional, obtemos o seguinte rascunho:

Hipóteses	Conclusão
$n \in \mathbb{Z}$	n é par $\rightarrow$ n^2 é par
	n^2 é par $\rightarrow$ n é par

Para facilitar a construção da demonstração, vamos dividir o rascunho em duas provas distintas, uma para cada uma das implicações. Primeiramente, temos:

Hipóteses	Conclusão
$n \in \mathbb{Z}$	n é par $\rightarrow$ n^2 é par

Utilizando a estratégia de prova direta, obtemos:

Hipóteses	Conclusão
$n \in \mathbb{Z}$	n^2 é par
n é par	

Evidentemente, podemos representar o fato de que um número x é par por $\exists y.x = 2y$. Usando esta representação:

Hipóteses	Conclusão
$n \in \mathbb{Z}$	$\exists k.n^2 = 2k$
$\exists m.n = 2m$	

Usando a estratégia de hipóteses para o quantificador existencial, obtemos:

Hipóteses	Conclusão
$n \in \mathbb{Z}$	$\exists k.n^2 = 2k$
$\exists m.n = 2m$	
$n = 2m$	

Agora, resta encontrar um valor de k que torne a igualdade $n^2 = 2k$ verdadeira. Note que possuímos como hipótese que $n = 2m$. Logo, temos que $n^2 = 4m^2$ e desta forma, temos que $k = 2m^2$, uma vez que $n^2 = 2k$ e $n^2 = 4m^2$.

Para a segunda implicação, ao invés de utilizarmos a técnica de prova direta, usaremos demonstração pela contrapositiva. Inicialmente, temos a seguinte configuração do rascunho:

Hipóteses	Conclusão
$n \in \mathbb{Z}$	n^2 é par $\rightarrow$ n é par

Ao usarmos a técnica de prova pela contrapositiva, temos:

Hipóteses	Conclusão
$n \in \mathbb{Z}$	$\neg(n^2$ é par $)$
$\neg(n$ é par$)$	

Evidentemente, como $n \in \mathbb{Z}$, se n não é par, temos que este deve ser ímpar. Logo:

Hipóteses	Conclusão
$n \in \mathbb{Z}$	n^2 é ímpar
n é ímpar	

Representamos o fato de que x é um número ímpar por $\exists y.x = 2y + 1$:

Hipóteses	Conclusão
$n \in \mathbb{Z}$	$\exists k.n^2 = 2k + 1$
$\exists m.n = 2m + 1$	

Usando a hipótese existencial, obtemos que $n = 2m + 1$.

Hipóteses	Conclusão
$n \in \mathbb{Z}$	$\exists k.n^2 = 2k + 1$
$\exists m.n = 2m + 1$	
$n = 2m + 1$	

Se $n = 2m + 1$, temos que $n^2 = 4m^2 + 4m + 1$. Para terminar a demonstração, devemos encontrar um valor k tal que $n^2 = 2k + 1$. Usando o fato de que $n^2 = 4m^2 + 4m + 1$ chega-se a conclusão de que $k = 2m^2 + 2m$[6].

Agora resta construirmos o texto a partir deste rascunho. Como já feito em outros exemplos, faremos a construção deste passo a passo. Primeiramente, utilizamos a estratégia de prova para o conectivo bicondicional.

[6]Note que não apresentamos a demonstração da igualdade anterior e desta propositalmente, visto que estas serão apresentadas no texto final desta prova.

Suponha $n \in \mathbb{Z}$.
($\rightarrow$): [Prova de n é par $\rightarrow n^2$ é par]
($\rightarrow$): [Prova de n^2 é par $\rightarrow n$ é par]
Portanto, se $n \in \mathbb{Z}$ então n é par sse n^2 é par.

Provando a primeira implicação por prova direta, obtemos a seguinte versão parcial do texto:

Suponha $n \in \mathbb{Z}$.
($\rightarrow$): Suponha n é par.
[Prova de n^2 é par].
Logo, se n é par, n^2 é par.
($\rightarrow$): [Prova de n^2 é par $\rightarrow n$ é par]
Portanto, se $n \in \mathbb{Z}$ então n é par sse n^2 é par.

Para completar a primeira parte da demonstração, basta provar que n^2 é par usando a estratégia de prova para quantificadores existenciais.

Suponha $n \in \mathbb{Z}$.
($\rightarrow$): Suponha n é par.
Como n é par, temos que $n = 2m$.
Seja $k = 2m^2$. Temos:

$$\begin{array}{ll} 2k & = \\ 2.2m^2 & = \\ 4m^2 & = \\ (2m)^2 & = \\ n^2 & \end{array}$$

Logo, n^2 é par.
Logo, se n é par, n^2 é par.
($\rightarrow$): [Prova de n^2 é par $\rightarrow n$ é par]
Portanto, se $n \in \mathbb{Z}$ então n é par sse n^2 é par.

Para a segunda implicação, utilizaremos a estratégia da contrapositiva, seguida da estratégia para o quantificador existencial.

Suponha $n \in \mathbb{Z}$.
($\rightarrow$): Suponha n é par.
Como n é par, temos que $n = 2m$.
Seja $k = 2m^2$. Temos:

$$\begin{array}{ll} 2k & = \\ 2.2m^2 & = \\ 4m^2 & = \\ (2m)^2 & = \\ n^2 & \end{array}$$

Logo, n^2 é par.
Logo, se n é par, n^2 é par.
($\rightarrow$): Suponha que n é ímpar.
Como n é ímpar, temos que $n = 2m + 1$.
Seja $k = 2m^2 + 2m$. Temos:

$$\begin{array}{ll} 2k+1 & = \\ 2(2m^2+2m)+1 & = \\ 4m^2+4m+1 & = \\ (2m+1)^2 & = \\ n^2 & \end{array}$$

Logo, n^2 é ímpar.
Logo, se n^2 é par, n é par.
Portanto, se $n \in \mathbb{Z}$ então n é par se e somente se n^2 é par.

■

4.3.7 Exercícios

1. Prove os seguintes teoremas:
   (a) Suponha $x, y \in \mathbb{Z}$ ímpares. Então xy é um número ímpar.
   (b) Suponha $n \in \mathbb{Z}$. Então n^3 é par se e somente se n é par.
   (c) Suponha $a, b \in \mathbb{Z}$ arbitrários. Então, existe $c \in \mathbb{Z}$ tal que $a \mid c$ e $b \mid c$.
   (d) Para todo $n \in \mathbb{Z}$, $15 \mid n$ se e somente se $3 \mid n$ e $5 \mid n$.

4.3.8 Estratégias para Disjunção ($\vee$)

Suponha que você possua uma hipótese da forma $\alpha \vee \beta$. Como utilizar esta hipótese para deduzir uma conclusão γ? Na dedução natural, a regra de eliminação da disjunção fornece uma forma de como utilizar $\alpha \vee \beta$ para deduzir γ: primeiramente, supomos que α é verdade e deduzimos γ e na sequência deduzimos γ a partir de β.

A utilização da eliminação da disjunção é comumente denominada de "prova por análise de casos" pois, considera-se cada uma das possibilidades de $\alpha \vee \beta$ ser verdadeiro para construção da demonstração. A seguir, apresentamos a estratégia de uso de hipóteses que resume esta idéia.

Estratégia de Uso de Hipóteses 7 (Para usar uma hipótese da forma $\alpha \vee \beta$)**.** Divida sua prova em casos. No primeiro caso, assuma que α é verdadeiro e deduza a conclusão γ. No segundo caso, assuma que β é verdadeiro e deduza a conclusão γ.

Rascunho.

Rascunho antes de usar a estratégia.

Hipóteses	Conclusão
$\alpha_1, \alpha_2, ..., \alpha_n$	γ
$\alpha \vee \beta$	

Rascunho depois de usar a estratégia.

Hipóteses	Conclusão
Caso 1:	
$\alpha_1, \alpha_2, ..., \alpha_n$	γ
α	
Caso 2:	
$\alpha_1, \alpha_2, ..., \alpha_n$	γ
β	

Texto:
Caso 1: α é verdadeiro..
[*Prova de γ*]
Caso 2: β é verdadeiro..
[*Prova de γ*]
Como os casos cobrem todas as possibilidades, temos que γ.

■

A seguir apresentamos um exemplo de uso desta estratégia de prova.

Exemplo 66. Considere a tarefa de demonstrar o seguinte teorema:

Suponha que $x \in \mathbb{R}$ é arbitrário. Então se $|x-3| > 3$ então $x^2 > 6x$.

Neste exemplo, possuímos como hipótese que $x \in \mathbb{R}$ e a conclusão é representada pela seguinte fórmula:

$$|x-3| > 3 \rightarrow x^2 > 6x$$

Logo, a versão inicial do rascunho possui a seguinte forma:

Hipóteses	Conclusão
$x \in \mathbb{R}$	$\|x-3\| > 3 \rightarrow x^2 > 6x$

Como a conclusão possui uma implicação, iniciamos a dedução utilizando a técnica de prova direta, o que nos leva a:

Hipóteses	Conclusão
$x \in \mathbb{R}$	$x^2 > 6x$
$\|x-3\| > 3$	

Note que para usarmos a hipótese $|x-3| > 3$ devemos saber se $x-3 \geq 0$ ou se $x-3 < 0$. Logo, devemos considerar uma divisão da prova em casos, usando a estratégia apresentada anteriormente.

Hipóteses	Conclusão
Caso 1:	
$x \in \mathbb{R}$	$x^2 > 6x$
$\|x-3\| > 3$	
$x-3 \geq 0$	
Caso 2:	
$x \in \mathbb{R}$	$x^2 > 6x$
$\|x-3\| > 3$	
$x-3 < 0$	

Se $x-3 \geq 0$, temos que $|x-3| = x-3$. Assim, temos que $|x-3| > 3 \equiv x-3 > 3 \equiv x > 6$. Logo, $x^2 > 6x$.

Por sua vez, se $x-3 < 0$, temos que $|x-3| = 3-x$. Assim, temos que $|x-3| > 3 \equiv 3-x > 3 \equiv -x > 3-3 \equiv -x > 0 \equiv x < 0$. Logo, $x^2 > 6x$.

Como terminamos a dedução da conclusão a partir das hipóteses, vamos proceder para a construção passo a passo do texto. Inicialmente, utilizamos a estratégia de prova direta.

Suponha $x \in \mathbb{R}$.
 Suponha que $|x-3| > 3$.
  [Prova $x^2 > 6x$.]
 Logo, se $|x-3| > 3$ então $x^2 > 6x$.
Logo, se $x \in \mathbb{R}$ então se $|x-3| > 3$ então $x^2 > 6x$.

Dividindo a prova em casos, temos que:

Suponha $x \in \mathbb{R}$.
 Suponha que $|x-3| > 3$.
  Caso 1: $x - 3 \geq 0$.
   [Prova de $x^2 > 6x$]
  Caso 2: $x - 3 < 0$.
   [Prova de $x^2 > 6x$]
  Como os casos cobrem todas as possibilidades, temos que $x^2 > 6x$.
 Logo, se $|x-3| > 3$ então $x^2 > 6x$. Logo, se $x \in \mathbb{R}$ então se $|x-3| > 3$ então $x^2 > 6x$.

Finalmente, concluímos o texto apresentando a dedução de $x^2 > 6x$ em cada caso.

Suponha $x \in \mathbb{R}$.
 Suponha que $|x-3| > 3$.
  Caso 1: $x - 3 \geq 0$.
   Como $x - 3 \geq 0$, temos que $|x-3| = x - 3$.
   Como $|x-3| = x - 3$ e $|x-3| > 3$, temos que $x > 6$.
   Como $x > 6$, temos que $x^2 > 6x$.
  Caso 2: $x - 3 < 0$.
   Como $x - 3 < 0$, temos que $|x-3| = 3 - x$.
   Como $|x-3| > 3$ e $|x-3| = 3 - x$, temos que $x < 0$.
   Como $x < 0$, temos que $x^2 > 6x$.
  Como os casos cobrem todas as possibilidades, temos que $x^2 > 6x$.
 Logo, se $|x-3| > 3$ então $x^2 > 6x$.
Logo, se $x \in \mathbb{R}$ então se $|x-3| > 3$ então $x^2 > 6x$.

■

Para demonstrar uma conclusão que é uma disjunção, devemos proceder de maneira similar às regras de introdução deste conectivo, conforme apresentado na estratégia seguinte.

Estratégia de Prova 9 (Para provar uma conclusão da forma $\alpha \vee \beta$)**.** Prove α ou prove β. ■

O próximo exemplo ilustra esta estratégia.

Exemplo 67. Considere a tarefa de provar o seguinte teorema:

Para todo $x \in \mathbb{Z}$, o resto da divisão de x^2 por 4 é 0 ou 1.

O teorema considerado pode ser representado pela seguinte fórmula, que corresponde a sua conclusão:

$$\forall x.x \in \mathbb{Z} \rightarrow x^2 \bmod 4 = 0 \vee x^2 \bmod 4 = 1$$

Assim, temos a seguinte versão inicial do rascunho:

x	x^2	$x^2 \div 4$	$x^2 \bmod 4$
1	1	0	1
2	4	1	0
3	9	2	1
4	16	4	0
5	25	6	1

Hipóteses	Conclusão
	$\forall x.x \in \mathbb{Z} \rightarrow x^2 \bmod 4 = 0 \vee x^2 \bmod 4 = 1$

Como a conclusão envolve uma fórmula que utiliza o quantificador universal e uma implicação, iniciamos a introdução utilizando as estratégias para estes símbolos da lógica.

Hipóteses	Conclusão
x arbitrário	$x^2 \bmod 4 = 0 \vee x^2 \bmod 4 = 1$
$x \in \mathbb{Z}$	

Neste ponto, surge a seguinte questão: Como continuar com esta prova? Visto que as hipóteses não acrescentam nenhuma idéia de como concluí-la, vamos montar uma tabela com alguns valores simples para tentar perceber se existe alguma estrutura "oculta". Aparentemente, temos que o resto é zero sempre que x é par e um caso x é ímpar. Logo, dividiremos a prova em casos. No primeiro caso, consideraremos que x é par e no segundo que x é ímpar.

Hipóteses	Conclusão
Caso 1:	
x é par	$x^2 \bmod 4 = 0 \vee x^2 \bmod 4 = 1$
x arbitrário	
$x \in \mathbb{Z}$	
Caso 2:	
x é ímpar	$x^2 \bmod 4 = 0 \vee x^2 \bmod 4 = 1$
x arbitrário	
$x \in \mathbb{Z}$	

Utilizando as definições de x é par e x é ímpar, temos que:

Hipóteses	Conclusão
Caso 1:	
x é par	$x^2 \bmod 4 = 0 \vee x^2 \bmod 4 = 1$
x arbitrário	
$x \in \mathbb{Z}$	
$\exists k_1.x = 2k_1$	
Caso 2:	
x é ímpar	$x^2 \bmod 4 = 0 \vee x^2 \bmod 4 = 1$
x arbitrário	
$x \in \mathbb{Z}$	
$\exists k_2.x = 2k_2 + 1$	

Se x é par, existe k_1 tal que $x = 2k_1$. Assim, temos que $x^2 = (2k_1)^2 = 4k_1^2$, que é divisível por 4 (resto igual a zero). Então, neste caso, optamos por demonstrar o lado esquerdo de $x^2 \bmod 4 = 0 \vee x^2 \bmod 4 = 1$, o que corresponde a regra de introdução da disjunção à esquerda.

Caso x seja ímpar, existe k_2 tal que $x = 2k_2 + 1$. Assim, temos que $x^2 = (2k_2 + 1)^2 = (2k_2)^2 + 2(2k_2) + 1 = 4k_2^2 + 4k_2 + 1 = 4(k^2 + k) + 1$, que dividido por 4 deixa um resto igual a um. Então, neste caso, optamos por demonstrar o lado direito de $x^2 \bmod 4 = 0 \vee x^2 \bmod 4 = 1$, o que corresponde a regra de introdução da disjunção à direita.

A construção do texto desta demonstração é apresentado a seguir.

Suponha x arbitrário.
[Prova de $x \in \mathbb{Z} \rightarrow x^2 \bmod 4 = 0 \vee x^2 \bmod 4 = 1$].
Como x é arbitrário, temos que para todo $x \in \mathbb{Z}$ o resto da divisão de x^2 por 4 é 0 ou 1.

Utilizando o texto para demonstrações de implicações, temos:

Suponha x arbitrário.
Suponha que $x \in \mathbb{Z}$.
[Prova de $x^2 \bmod 4 = 0 \vee x^2 \bmod 4 = 1$].
Logo, se $x \in \mathbb{Z}$ temos que $x^2 \bmod 4 = 0 \vee x^2 \bmod 4 = 1$.
Como x é arbitrário, temos que para todo $x \in \mathbb{Z}$ o resto da divisão de x^2 por 4 é 0 ou 1.

Neste ponto, consideramos os casos de que todo $x \in \mathbb{Z}$ é par ou ímpar.

Suponha x arbitrário.
Suponha que $x \in \mathbb{Z}$.
Caso 1: x é par.
[Prova de $x^2 \bmod 4 = 0 \vee x^2 \bmod 4 = 1$].
Caso 2: x é ímpar.
[Prova de $x^2 \bmod 4 = 0 \vee x^2 \bmod 4 = 1$].
Logo, se $x \in \mathbb{Z}$ temos que $x^2 \bmod 4 = 0 \vee x^2 \bmod 4 = 1$.
Como x é arbitrário, temos que para todo $x \in \mathbb{Z}$ o resto da divisão de x^2 por 4 é 0 ou 1.

Agora, para o caso de x ser par, provamos que $x^2 \bmod 4 = 0$ e para o caso de ser ímpar, provamos que $x^2 \bmod 4 = 1$.

Suponha x arbitrário.
Suponha que $x \in \mathbb{Z}$.
Caso 1: x é par.
[Prova de $x^2 \bmod 4 = 0$].
Caso 2: x é ímpar.
[Prova de $x^2 \bmod 4 = 1$].
Logo, se $x \in \mathbb{Z}$ temos que $x^2 \bmod 4 = 0 \vee x^2 \bmod 4 = 1$.
Como x é arbitrário, temos que para todo $x \in \mathbb{Z}$ o resto da divisão de x^2 por 4 é 0 ou 1.

Finalmente, concluímos a prova utilizando a hipótese existencial de que x é par ou ímpar em cada caso.

Suponha x arbitrário.
Suponha que $x \in \mathbb{Z}$.
Caso 1: x é par.
Como x é par, existe k_1 tal que $x = 2k_1$.

Como $x = 2k_1$, temos que $x^2 = (2k_1)^2 = 4k_1^2$.
Como $x^2 = 4k_1^2$, temos que $x^2 \bmod 4 = 0$.
Logo, $x^2 \bmod 4 = 0$ ou $x^2 \bmod 4 = 1$
Caso 2: x é ímpar.
Como x é ímpar, existe k_2 tal que $x = 2k_2 + 1$.
Como $x = 2k_2 + 1$, temos que $x^2 = (2k_2 + 1)^2 = 4k_2^2 + 4k_2 + 1$.
Como $x^2 = 4k_2^2 + 4k_2 + 1$, temos que $x^2 \bmod 4 = 1$.
Logo, $x^2 \bmod 4 = 0$ ou $x^2 \bmod 4 = 1$
Logo, se $x \in \mathbb{Z}$ temos que $x^2 \bmod 4 = 0 \vee x^2 \bmod 4 = 1$.
Como x é arbitrário, temos que para todo $x \in \mathbb{Z}$ o resto da divisão de x^2 por 4 é 0 ou 1.

■

Ainda resta uma última técnica pode ser utilizada para manipular hipóteses ou conclusões da forma $\alpha \vee \beta$. Esta técnica é baseada nas seguintes equivalências: $\alpha \vee \beta \equiv \neg\alpha \rightarrow \beta \equiv \neg\beta \rightarrow \alpha$[7].

Estratégia de Prova 10 (Para provar uma conclusão da forma $\alpha \vee \beta$)**.** Se α é verdadeiro, é evidente que $\alpha \vee \beta$ é verdadeiro. Logo, podemos supor que α é falso e demonstrar que β é verdadeiro para concluir $\alpha \vee \beta$.

Rascunho.

Rascunho antes de usar a estratégia.

Hipóteses	Conclusão
$\alpha_1, \alpha_2, ..., \alpha_n$	$\alpha \vee \beta$

Rascunho depois de usar a estratégia.

Hipóteses	Conclusão
$\alpha_1, \alpha_2, ..., \alpha_n$	β
$\neg\alpha$	

Texto:
Se α é verdadeiro, então $\alpha \vee \beta$ é verdadeiro. Então, suponha que $\neg\alpha$.
[Prova de β].
Logo, temos que $\alpha \vee \beta$.

■

Exemplo 68. Considere demonstrar o seguinte teorema simples:

Para todo $x \in \mathbb{R}$, se $x^2 \geq x$ então $x \leq 0$ ou $x \geq 1$.

Seguindo os passos já apresentados para demonstração de teoremas, temos que o teorema acima é representado pela seguinte fórmula:

$$\forall x.x \in \mathbb{R} \rightarrow x^2 \geq x \rightarrow x \leq 0 \vee x \geq 1$$

A configuração inicial do rascunho é:

Hipóteses	Conclusão
	$\forall x.x \in \mathbb{R} \rightarrow x^2 \geq x \rightarrow x \leq 0 \vee x \geq 1$

[7] Demonstre essas equivalências!

Devido a composição desta fórmula, iniciaremos a demonstração deste teorema utilizando as técnicas para o quantificador universal e implicação (prova direta). Com isso, obtemos:

Hipóteses	Conclusão
x arbitrário	$x \leq 0 \vee x \geq 1$
$x \in \mathbb{R}$	
$x^2 \geq x$	

Agora, utilizaremos a estratégia de considerar que $\alpha \vee \beta$ é equivalente a $\neg\alpha \rightarrow \beta$:

Hipóteses	Conclusão
x arbitrário	$x \geq 1$
$x \in \mathbb{R}$	
$x^2 \geq x$	
$x > 0$	

É óbvio que $\neg(x \leq 0) \equiv x > 0$. Como $x > 0$ e $x^2 \geq x$, dividindo ambos os lados da última desigualdade por x, obtemos $x \geq 1$, conforme requerido. O texto é construído passo a passo utilizando os modelos para cada uma das estratégias utilizadas. Inicialmente, o texto para o quantificador universal e provas diretas.

Suponha x arbitrário.
  Suponha $x \in \mathbb{R}$.
    Suponha $x^2 \geq x$.
      [Prova de $x \leq 0 \vee x \geq 1$].
    Logo, se $x^2 \geq x$ então $x \leq 0 \vee x \geq 1$.
  Logo, se $x \in \mathbb{R}$, então se $x^2 \geq x$ então $x \leq 0 \vee x \geq 1$.
Como x é arbitrário, temos que para todo $x \in \mathbb{R}$, se $x^2 \geq x$ então $x \leq 0$ ou $x \geq 1$.

Agora, utilizando o modelo de texto para disjunção, temos:

Suponha x arbitrário.
  Suponha $x \in \mathbb{R}$.
    Suponha $x^2 \geq x$.
      Se $x \leq 0$, temos que $x \leq 0$ ou $x \geq 1$. Então, suponha $x > 0$.
        [Prova de $x \geq 1$].
      Logo, $x \leq 0$ ou $x \geq 1$.
    Logo, se $x^2 \geq x$ então $x \leq 0 \vee x \geq 1$.
  Logo, se $x \in \mathbb{R}$, então se $x^2 \geq x$ então $x \leq 0 \vee x \geq 1$.
Como x é arbitrário, temos que para todo $x \in \mathbb{R}$, se $x^2 \geq x$ então $x \leq 0$ ou $x \geq 1$.

Finalmente, encerramos o texto desta demonstração utilizando a dedução de $x \geq 1$ a partir de $x^2 \geq x$ e $x > 0$.

Suponha x arbitrário.
  Suponha $x \in \mathbb{R}$.
    Suponha $x^2 \geq x$.
      Se $x \leq 0$, temos que $x \leq 0$ ou $x \geq 1$. Então, suponha $x > 0$.
        Como $x^2 \geq x$ e $x > 0$, temos que $x \geq 1$.
      Logo, $x \leq 0$ ou $x \geq 1$.

Logo, se $x^2 \geq x$ então $x \leq 0 \vee x \geq 1$.
Logo, se $x \in \mathbb{R}$, então se $x^2 \geq x$ então $x \leq 0 \vee x \geq 1$.
Como x é arbitrário, temos que para todo $x \in \mathbb{R}$, se $x^2 \geq x$ então $x \leq 0$ ou $x \geq 1$.

■

A próxima estratégia de uso de hipóteses mostra como podemos usar uma disjunção como uma implicação.

Estratégia de Uso de Hipóteses 8 (Para utilizar uma hipótese da forma $\alpha \vee \beta$)**.** Considere-a equivalente a $\neg\alpha \rightarrow \beta$ ou a $\neg\beta \rightarrow \alpha$. ■

4.3.9 Exercícios

1. Prove os seguintes teoremas:
   (a) Suponha $x, y \in \mathbb{R}$ e que $x \neq 0$. Então, $y + \frac{1}{x} = 1 + \frac{y}{x}$ se e somente se $x = 1$ ou $y = 1$.
   (b) Para todo $x \in \mathbb{Z}$, $x^2 + x$ é par.
   (c) Para todo $a, b \in \mathbb{R}$, $|a| \leq b$ se e somente se $-b \leq a \leq b$.
   (d) Para todo $x \in \mathbb{R}$, $|2x - 6| > x$ se e somente se $|x - 4| > 2$.

4.3.10 Existência e Unicidade

Em matemática é comum a especificação de propriedades similares a "existe um único elemento x que possui uma propriedade P". Considerando um certo universo de discurso U, dizemos que existe um único elemento de U que satisfaz uma propriedade P usando a seguinte fórmula:

$$\exists x.P(x) \wedge \neg\exists y.P(y) \wedge y \neq x.$$

Que essencialmente especifica que não existe um elemento diferente de x que satisfaça P. Normalmente, matemáticos expressam esta fórmula como um novo quantificador (representado por $\exists!$). Utilizando este quantificador, a fórmula anterior pode ser representada de maneira mais concisa como $\exists!x.P(x)$.

Porém, a fórmula

$$\exists x.P(x) \wedge \neg\exists y.P(y) \wedge y \neq x.$$

não é a única maneira de representarmos $\exists!x.P(x)$. Se utilizarmos um pouco de álgebra booleana podemos eliminar a negação da fórmula anterior, conforme apresentado a seguir:

$$\begin{array}{ll} \exists x.P(x) \wedge \neg\exists y.P(y) \wedge y \neq x & \equiv \\ \exists x.P(x) \wedge \forall y.\neg(P(y) \wedge y \neq x) & \equiv \\ \exists x.P(x) \wedge \forall y.\neg P(y) \vee \neg y \neq x & \equiv \\ \exists x.P(x) \wedge \forall y.P(y) \rightarrow x = y & \end{array}$$

Note que esta versão possui a vantagem de não envolver negação, o que usualmente facilita as demonstrações. Outra fórmula equivalente a $\exists!x.P(x)$ é:

$$\exists x.P(x) \wedge \forall y.\forall z.P(y) \wedge P(z) \rightarrow y = z$$

Note que a última fórmula apresentada é bastante similar a $\exists x.P(x) \wedge \forall y.P(y) \rightarrow x = y$. A diferença é a introdução da nova variável quantificada z. Acredito que o leitor deva estar se perguntando, "mas porquê introduzir uma nova variável?". O ponto é que na fórmula

$$\exists x.P(x) \wedge \forall y.\forall z.P(y) \wedge P(z) \rightarrow y = z$$

a variável x não aparece livre em $\forall y.\forall z.P(y) \wedge P(z) \rightarrow y = z$, o que nos permite dividir a tarefa de demonstrar $\exists x.P(x) \wedge \forall y.\forall z.P(y) \wedge P(z) \rightarrow y = z$ nas demonstrações:

- $\exists x.P(x)$
- $\forall y.\forall z.P(y) \wedge P(z) \rightarrow y = z$

o que não pode ser feito com a fórmula

$$\exists x.P(x) \wedge \forall y.P(y) \rightarrow x = y$$

já que o x aparece livre em $\forall y.P(y) \rightarrow x = y$.

A utilização destas equivalências é o que determinará as estratégias de prova para este quantificador.

4.3.11 Estratégias para Existências e Unicidade

Conforme discutido na seção anterior, existem diversas maneiras de se representar o quantificador $\exists!x.P(x)$ e estas determinam as estratégias de demonstração e uso de hipóteses para este tipo de fórmula. Estas estratégias são apresentadas a seguir.

Estratégia de Prova 11 (Para provar uma conclusão da forma $\exists!x.P(x).$)**.** Prove $\exists x.P(x)$ e $\forall y.\forall z.P(x) \wedge P(z) \rightarrow y = z$. A primeira parte da prova mostra que existe um valor x tal que $P(x)$ e a segunda mostra que este valor é único.

A construção do rascunho será omitida, visto que este utilizará estratégias de prova adequadas para cada uma das partes desta demonstração. Normalmente, adicionamos um rótulo no texto correspondente a cada uma das partes da prova. O rótulo "Existência" é utilizado para a demonstração de $\exists x.P(x)$ e "Unicidade" é utilizado para $\forall y.\forall z.P(x) \wedge P(z) \rightarrow y = z$. Isto é formalizado pelo seguinte modelo de texto.

Texto:

Existência: [Prova de $\exists x.P(x)$]
Unicidade: [Prova de $\forall y.\forall z.P(x) \wedge P(z) \rightarrow y = z$]

■

Outra possível estratégia de prova é baseada em outra equivalência para $\exists!x.P(x)$, conforme apresentado a seguir.

Estratégia de Prova 12 (Para provar uma conclusão da forma $\exists!x.P(x).$)**.** Prove $\exists x.P(x) \wedge \forall y.P(y) \rightarrow x = y$ utilizando outras estratégias de demonstração.

■

Exemplo 69. Considere a tarefa de demonstrar o seguinte teorema

Para todo $x \in \mathbb{R}$ se $x \neq 2$ então existe um único y tal que $\frac{2y}{y+1} = x$.

Este teorema é representado pela seguinte conclusão:

$$\forall x.x \in \mathbb{R} \rightarrow x \neq 2 \rightarrow \exists! y.\frac{2y}{y+1} = x$$

o que nos leva ao seguinte rascunho inicial

Hipóteses	Conclusão
	$\forall x.x \in \mathbb{R} \rightarrow x \neq 2 \rightarrow \exists! y.\frac{2y}{y+1} = x$

que utilizando estratégias de prova já conhecidas nos leva a seguinte situação do rascunho:

Hipóteses	Conclusão
x arbitrário	$\exists! y.\frac{2y}{y+1} = x$
$x \in \mathbb{R}$	
$x \neq 2$	

Para concluir a demonstração, utilizaremos a seguinte equivalência:

$$\exists! x.P(x) \equiv \exists x.P(x) \wedge \forall y.P(y) \rightarrow x = y$$

o que nos leva ao seguinte rascunho:

Hipóteses	Conclusão
x arbitrário	$\exists y.\frac{2y}{y+1} = x \wedge \forall z.\frac{2z}{z+1} = x \rightarrow z = y$
$x \in \mathbb{R}$	
$x \neq 2$	

Agora, temos que encontrar um valor de y que permita provar que $\exists y.\frac{2y}{y+1} = x$. Encontraremos o valor apropriado para y resolvendo a equação $\frac{2y}{y+1} = x$ para y, conforme apresentado a seguir:

$$\begin{array}{lcll} \frac{2y}{y+1} & = & x & \Rightarrow \\ x(y+1) & = & 2y & \Rightarrow \\ xy + x - 2y & = & 0 & \Rightarrow \\ y(x-2) & = & -x & \Rightarrow \\ y = \frac{x}{2-x} & & & \end{array}$$

Utilizando o valor de $y = \frac{x}{2-x}$ concluímos a demonstração sem maiores problemas. Abaixo apresentamos a versão final do texto desta prova.

Suponha x arbitrário.
  Suponha $x \in \mathbb{R}$.
    Suponha $x \neq 2$.
      Seja $y = \dfrac{x}{2-x}$. Temos:

$$
\begin{array}{ll}
\dfrac{2y}{y+1} & = \\
\dfrac{2\dfrac{x}{2-x}}{\dfrac{x}{2-x}+1} & = \\
\dfrac{\dfrac{2x}{2-x}}{\dfrac{x+2-x}{2-x}} & = \\
\dfrac{2x}{2-x} \times \dfrac{2-x}{2} & = \\
\dfrac{2x}{2} & = \\
x &
\end{array}
$$

Logo, existe y tal que $\dfrac{2y}{2-y} = x$.

Suponha z arbitrário.

Suponha que $\dfrac{2z}{z+1} = x$.

Como $\dfrac{2z}{z+1} = x$, temos:

$$
\begin{array}{lcll}
\dfrac{2z}{z+1} & = & x & \Rightarrow \\
x(z+1) & = & 2z & \Rightarrow \\
xz + x - 2z & = & 0 & \Rightarrow \\
z(x-2) & = & -x & \Rightarrow \\
z = \dfrac{x}{2-x} & & &
\end{array}
$$

Logo, $z = y$.

Logo, se $\dfrac{2z}{z+1} = x$ então $z = y$.

Como z é arbitrário, temos que para todo z, se $\dfrac{2z}{z+1} = x$ então $z = y$.

Como x é arbitrário, temos que se $x \neq 2$ então existe um único y tal que $\dfrac{2y}{y+1} = x$.

■

4.3.12 Exercícios

1. Seja $P(x)$ uma fórmula da lógica de predicados em que x é uma variável livre.

   (a) Encontre uma fórmula da lógica de predicados que represente: "existem exatamente dois valores de x que fazem $P(x)$ ser verdadeira".

   (b) Baseado na resposta do item anterior, descreva uma estratégia de prova para fórmulas "existem exatamente dois valores de x que tornam $P(x)$ verdadeiro".

   (c) Utilizando a estratégia de prova criada por você, mostre que a equação $x^3 = x^2$ possui exatamente duas raízes.

4.3.13 Estratégia de prova por absurdo

Agora, apresentaremos uma estratégia de prova que é aplicável a qualquer conclusão. Esta é equivalente a regra *reductio ad absurdum* da dedução natural.

Estratégia de Prova 13 (Para provar uma conclusão α qualquer.)**.** Suponha $\neg\alpha$ e tente deduzir uma contradição. A partir desta contradição pode-se concluir que α é verdadeiro.

Rascunho.

Rascunho antes de usar a estratégia.

Hipóteses	Conclusão
$\gamma_1, \gamma_2, ..., \gamma_n$	α

Rascunho depois de usar a estratégia.

Hipóteses	Conclusão
$\gamma_1, \gamma_2, ..., \gamma_n$	$\bot$
$\neg\alpha$	

Texto:
Suponha que $\neg\alpha$.
[Prova de $\bot$].
Logo, temos que α.

■

Exemplo 70. Neste exemplo, vamos demonstrar um dos mais conhecidos resultados da matemática: a infinitude do conjunto de números primos.

Para essa demonstração utilizaremos um resultado, sem demonstração, de que todo número $n > 1$ ou é primo ou é um produto de números primos[8].

O enunciado do teorema é apresentado a seguir:

Existem infinitos números primos.

Note que este teorema envolve uma negação, uma vez que, "infinito" pode ser entendido como "não finito". Porém, como podemos dizer que um conjunto é finito? Especificamos que um conjunto é finito dizendo que este possui n elementos, em que $n \in \mathbb{N}$. Logo, temos a seguinte versão inicial do rascunho:

Hipóteses	Conclusão
	$\neg\exists n.\{p_1, ..., p_n\}$ são todos primos.

Como a conclusão envolve uma negação, podemos iniciar a dedução utilizando a estratégia de demonstração por absurdo. Mas, como garantir que existem "finitos" números primos? A idéia é "listar" todos os n números primos e especificar que não existe outro número primo que não pertença a listagem de primos citada. Isto é especificado na seguinte versão do rascunho.

Hipóteses	Conclusão
$p_1, ..., p_n$ são primos	$\bot$
$\neg\exists q.q$ é primo $\wedge q \notin \{p_1, ..., p_n\}$	

Como obter uma contradição a partir destas hipóteses? Note que a hipótese:

[8] este resultado será demonstrado quando estudarmos indução matemática.

$\neg\exists q.q$ é primo $\wedge q \notin \{p_1, ..., p_n\}$

pode ser utilizada para construir uma contradição se mostrarmos um número q que é primo e que $q \notin \{p_1, ..., p_n\}$. Considere o número $x = p_1 \times p_2 \times ... \times p_n$ um número formado pelo produto de todos os n números primos. Evidentemente, x é divisível por cada um dos números deste conjunto. Seja $q = x + 1$. Evidentemente, temos que $q > 1$ e, portanto, temos que q deve ser um número primo ou um produto de primos. Se q é primo, temos que $\{p_1, ..., p_n\}$ não é o conjunto contendo todos os primos pois $q \notin \{p_1, ..., p_n\}$. Se q é um produto de primos, temos que q deve ser divisível por algum dos primos em $\{p_1, ..., p_n\}$. Mas, como $q = p_1 \times p_2 \times ... \times p_n + 1$, temos que q não é divisível por nenhum dos números em $\{p_1, ..., p_n\}$. Desta forma, temos que q deve ser primo e, como $q \notin \{p_1, ..., p_n\}$, temos que $\{p_1, ..., p_n\}$ não pode ser o conjunto de todos os números primos. Como ambos os casos cobrem todas as possibilidades, temos que q não é divisível por nenhum p_i, $1 \leq i \leq n$. Logo, q é divisível por 1 e por si próprio. Logo, q é primo, contrariando a hipótese de que todos os primos estão listados no conjunto $\{p_1, ..., p_n\}$. Assim, podemos concluir que existem infinitos números primos. A seguir, construímos o texto para esta demonstração.

Inicialmente, começamos o texto utilizando o modelo de texto para demonstrações por absurdo.

Suponha que existam finitos números primos.
[Prova de $\perp$]
Logo, temos que existem infinitos números primos.

E mostramos como deduzir $\perp$, a partir das informações disponíveis no contexto.

Suponha que existam finitos números primos.
Sejam $p_1, ..., p_n$ o conjunto de todos os números primos, $n \in \mathbb{N}$.
Seja $q = p_1 \times p_2 \times ... \times p_n + 1$ e $q \notin \{p_1, ..., p_n\}$.
Evidentemente, $q > 1$.
Considere os casos:
q é primo:
Logo, $\{p_1, ..., p_n\}$ não é o conjunto de todos os primos.
q é produto de primos:
Mas, q não é divisível por nenhum $\{p_1, ..., p_n\}$. Logo, q é primo.
Logo, q é divisível por 1 e por si próprio.
Mas, $q \notin \{p_1, ..., p_n\}$, o que é uma contradição, pois $\{p_1, ..., p_n\}$ é o conjunto de todos os primos.
Logo, temos que existem infinitos números primos.

■

4.4 Notas Bibliográficas

Neste capítulo descrevemos técnicas para demonstração de teoremas que, basicamente, são adaptações das regras da dedução natural já apresentadas em capítulos anteriores.

Este capítulo é uma adaptação do capítulo 3 do livro [9]. Ao contrário de [9], não utilizamos nenhum exemplo envolvendo teoria de conjuntos. Isto

é proposital, pois a apresentaremos a teoria de conjunto no próximo capítulo junto com diversos exercícios sobre demonstração de teoremas.

Parte II

Teoria de Conjuntos, Relações e Funções

5

Teoria de Conjuntos

> Ninguém deveria nos expulsar do paraíso que Cantor criou.
>
> David Hilbert, Matemático Alemão sobre a Teoria de Conjuntos criada por Georg Cantor.

5.1 Motivação

De maneira simplista, pode-se dizer que o alicerce fundamental da matemática é a teoria de conjuntos. Isto se torna mais e mais evidente a medida que você avança por cursos mais avançados de matemática, já que a teoria de conjuntos é uma linguagem projetada para descrever e explicar todos os tipos de estruturas matemáticas.

Em se tratando de computação, a teoria de conjuntos possui um papel importante no projeto de estruturas de dados e bancos de dados. Primeiramente, diversas estruturas eficientes são implementações de um tipo abstrato de dados que define operações sobre conjuntos. Por sua vez, toda a teoria de bancos de dados relacionais é baseada em operações básicas sobre conjuntos.

O objetivo deste capítulo é apresentar a teoria de conjuntos e como esta pode ser utilizada para descrever propriedades de objetos matemáticos.

5.2 Introdução aos Conjuntos

Não apresentaremos uma definição formal do que é um conjunto. Isto se deve ao fato de que a teoria de conjuntos foi concebida com o intuito de ser a fundamentação teórica de toda a matemática. Isto é, em princípio, todos os objetos matemáticos são definidos em termos de conjuntos.

Conjuntos nada mais são que uma coleção de objetos denominados *elementos*. Porém, existem algumas restrições para considerarmos uma coleção de objetos um conjunto. A primeira diz respeito a *ordem*. Em um conjunto a ordem dos elementos é irrelevante. A segunda é sobre a *multiplicidade*. Esta

especifica que em um conjunto qualquer há somente uma ocorrência de um certo valor, isto é, não é permitido que um elemento apareça mais de uma vez em um mesmo conjunto.

Uma vez que elementos podem ocorrer uma única vez em um conjunto, podemos dizer que a operação de determinar se um elemento está ou não em um conjunto possui um valor lógico (isto é, verdadeiro ou falso). Se A é um conjunto e x um elemento, representamos por $x \in A$ o fato de x ser um elemento do conjunto A. Representamos que x não é um elemento de A por $x \notin A$. Note que a seguinte equivalência é verdadeira: $x \notin A \equiv \neg(x \in A)$.

Denominamos por *cardinalidade* ou tamanho o número de elementos de um conjunto. Se A é um conjunto, representamos por $|A|$ o número de elementos de A.

Existe um único conjunto A tal que $|A| = 0$. Este é conhecido como conjunto vazio e é representado como $\{\}$ ou $\emptyset$.

Adotaremos, como convenção, que conjuntos serão sempre representados por letras maiúsculas e elementos por letras minúsculas.

5.3 Descrevendo Conjuntos

Existem diversas maneiras para se descrever conjuntos. Apresentaremos, de maneira suscinta três maneiras: enumeração, *set comprehension*[1] e por recursão.

5.3.1 Enumeração

Definimos um conjunto por enumeração simplesmente listando seus elementos. Este é um método conveniente para conjuntos finitos que possuam poucos elementos.

O exemplo a seguir mostra alguns conjuntos definidos por enumeração.

Exemplo 71. Abaixo apresentamos alguns conjuntos definidos por enumeração:

$$\begin{array}{lcl} V & = & \{a, e, i, o, u\} \\ P & = & \{\text{arara, pelicano, pardal}\} \\ X & = & \{2, 4, 6, 8\} \\ J & = & \{\} \\ L & = & \{1, \{1\}\} \end{array}$$

A cardinalidade de cada um deles é:

$$\begin{array}{lcl} |V| & = & 5 \\ |P| & = & 3 \\ |X| & = & 4 \\ |J| & = & 0 \\ |L| & = & 2 \end{array}$$

[1] Infelizmente, não conheço uma tradução para este termo. Por isso, mantive o nome original.

Note que as seguintes proposições sobre pertinência nestes conjuntos são verdadeiras:

$$a \in V$$
$$\text{arara} \in P$$
$$2 \in X$$
$$\{1\} \in L$$
$$1 \in L$$

Por sua vez, as seguintes proposições são falsas considerando os conjuntos anteriores:

$$p \in V$$
$$\text{vaca} \in P$$
$$7 \in X$$
$$2 \in L$$
$$\{1, \{1\}\} \in \{1, \{1\}\}$$

■

5.3.2 Set Comprehension

A notação de set comprehension[2] permite-nos especificar um conjunto em termos de uma propriedade que descreve quais são os elementos deste. De maneira simples, temos que um set comprehension é representado da seguinte maneira:

$$\{x \in X \mid p(x)\}$$

em que x é uma variável (ou uma expressão), X um conjunto e $p(x)$ é uma fórmula da lógica de predicados. Esta maneira de descrever conjuntos é útil para descrever conjuntos com muitos elementos ou infinitos.

É importante notar que ocorrências da variável x em $p(x)$ no set comprehension $\{x \in X \mid p(x)\}$ são consideradas ligadas a $x \in X$. Assim, podemos caracterizar a pertinência a um conjunto definido usando set comprehension utilizando a seguinte equivalência:

$$y \in \{x \in X \mid p(x)\} \equiv y \in X \wedge p(y)$$

O leitor atento deve ter percebido que a equivalência anterior nada mais é que a aplicação da substituição $[x \mapsto y]$ a fórmula especificada no set comprehension.

Exemplo 72. Considere a tarefa de representar os conjuntos de todos os números naturais pares e de todos os números naturais múltiplos de 3. Poderíamos representar estes conjuntos da seguinte maneira:

$$\begin{aligned} P &= \{0, 2, 4, 6, ...\} \\ T &= \{0, 3, 6, 9, ...\} \end{aligned}$$

Apesar da estrutura parecer óbvia, o uso de "..." deve ser evitado por este permitir ambiguidades na interpretação de um conjunto. Sem saber que o conjunto T representa os números naturais múltiplos de 3, como saber se 173 pertence ou não a este conjunto?

[2]Manteremos o nome sem tradução por não conhecer um termo em língua portuguesa para este tipo de notação matemática.

Para evitar este tipo de ambiguidade, podemos utilizar set comprehensions para definir conjuntos infinitos de maneira precisa. Os conjuntos anteriores podem ser representados da seguinte maneira:

$$\begin{aligned} P &= \{x \in \mathbb{N} \mid \exists y.y \in \mathbb{N} \wedge x = 2y\} \\ T &= \{x \in \mathbb{N} \mid \exists y.y \in \mathbb{N} \wedge x = 3y\} \end{aligned}$$

Podemos representar os fatos de que $a \in P$ e $b \in T$ como as seguintes fórmulas:

$$a \in \mathbb{N} \wedge \exists y.y \in \mathbb{N} \wedge a = 2y$$
$$b \in \mathbb{N} \wedge \exists y.y \in \mathbb{N} \wedge b = 3y$$

Note que como utilizamos uma fórmula da lógica de predicados para descrever elementos de um conjunto não há margem para interpretações ambíguas. ■

Exemplo 73. Considere o tarefa de definir o conjunto de todos os números naturais que são quadrados perfeitos. Mais formalmente:

$$\{n^2 \mid n \in \mathbb{N}\}$$

Note que se $y \in \{n^2 \mid n \in \mathbb{N}\}$ deveríamos ser capazes de deduzir que y é um quadrado perfeito. A questão é que o conjunto

$$\{n^2 \mid n \in \mathbb{N}\}$$

é equivalente a

$$\{x \mid \exists n.n \in \mathbb{N} \wedge x = n^2\}$$

que, permite-nos deduzir que y é um quadrado perfeito. ■

De maneira geral, se um conjunto é definido usando set comprehension em que os elementos são especificados em termos de uma expressão ao invés de uma simples variável, esta pode ser convertida em uma definição equivalente em que os elementos são especificados somente usando variáveis, como no exemplo anterior.

O mecanismo de set comprehension é bastante expressivo. Inclusive devemos ter alguns cuidados para evitar a definição de paradoxos, como o descrito na próxima seção.

O Paradoxo de Russell

Antes de apresentar o paradoxo de Russell formalmente, é útil analisá-lo em um contexto mais simples, porém, equivalente.

Exemplo 74. Considere o seguinte problema:

> "Considere uma cidade em que existe apenas um barbeiro e que este faz a barba de todos que não fazem a própria barba. O barbeiro faz sua própria barba?"

Após refletir uma pouco sobre esta sentença, percebemos que esta é um paradoxo, pois:

- Se o barbeiro não faz a própria barba, ele deveria fazê-la, já que ele faz a barba apenas de quem não faz a própria barba.

- Porém se ele faz a própria barba, pela definição, ele não deveria fazê-la.

Ou seja, a sentença sobre o barbeiro desta cidade é um paradoxo. ■

Russell percebeu que a definição usando set comprehension poderia gerar um paradoxo similar ao apresentado no exemplo anterior. A demonstração deste paradoxo é apresentada a seguir.

Seja $\mathcal{S}$ o conjunto de todos os conjuntos que não são elementos de si próprios, isto é:

$$\mathcal{S} = \{X \mid X \notin X\}$$

Evidentemente, temos que $\mathcal{S} \in \mathcal{S}$ ou $\mathcal{S} \notin \mathcal{S}$. Considere os seguintes casos:

- Caso $\mathcal{S} \in \mathcal{S}$: Se $\mathcal{S} \in \mathcal{S}$, pela definição de $\mathcal{S}$, temos que $\mathcal{S} \notin \mathcal{S}$, o que constitui uma contradição.

- Caso $\mathcal{S} \notin \mathcal{S}$: Logo, pela definição de $\mathcal{S}$, temos que $\mathcal{S} \in \mathcal{S}$, o que constitui uma contradição.

Como ambos os casos cobrem todas as possibilidades, temos que $\mathcal{S} \in \mathcal{S}$ não pode ser uma proposição lógica, já que esta não pode ser determinada como verdadeira ou falsa.

5.3.3 Conjuntos Definidos Recursivamente

Conjuntos definidos por recursão são muito utilizados em computação para a definição de estruturas de dados e algoritmos. Nesta seção, veremos como definir conjuntos recursivamente.

Assim como toda definição recursiva, conjuntos indutivos[3] devem possuir casos base e passos recursivos. Porém, apenas estes elementos não são suficientes para caracterizar uma definição de um conjunto. Adicionalmente, devemos possuir uma regra, denominada *regra de fechamento*[4] que especifica que todos os elementos do conjunto definido são formados a partir do(s) caso(s) base e de um número finito de usos do(s) passo(s) recursivo(s).

Resumindo, para definir um conjunto recursivamente devemos especificar três partes: casos base, passos recursivos e regra de fechamento:

- Casos base consistem de afirmativas simples, como $1 \in S$.

- Passos recursivos consistem de afirmativas envolvendo implicações e quantificadores universais, como

$$\forall x.x \in S \rightarrow x + 1 \in S$$

- Regras de fechamento especificam que todo elemento do conjunto pode ser obtido a partir de um número finito de utilização das regras anteriores.

A seguir apresentaremos algumas definições de conjuntos definidos recursivamente.

[3] Conjuntos definidos recursivamente são também conhecidos como conjuntos indutivos.

[4] Normalmente, estas regras são denominadas como *extremal rule*, em textos sobre teoria de conjuntos.

Exemplo 75. Como um primeiro exemplo de definição de um conjunto recursivo, vamos considerar o conjunto dos números naturais, $\mathbb{N}$. O leitor deve lembrar da definição do conjunto $\mathcal{N}$ do capítulo 1 em que apresentamos a sintaxe de termos que representam números naturais utilizando a operação de sucessor e uma constante para representar o número 0.

Utilizaremos a mesma idéia para a definição recursiva de $\mathbb{N}$:

- Caso base: $0 \in \mathbb{N}$.
- Passo recursivo: $\forall n.n \in \mathbb{N} \rightarrow n + 1 \in \mathbb{N}$
- Regra de fechamento: Todo $n \in \mathbb{N}$ pode se obtido por um número finito de aplicações das regras anteriores.

■

Exemplo 76. Considere $P = \{x \in \mathbb{N} \,|\, \exists y.y \in \mathbb{N} \wedge x = 2y\}$, que obviamente representa o conjunto de todos os números naturais pares. Podemos definí-lo recursivamente se percebermos que este é formado por um caso base ($0 \in P$) e que se $n \in P$ então $n + 2 \in P$, o que é definido formalmente a seguir:

- Caso base: $0 \in P$
- Passo recursivo: $\forall n.n \in P \rightarrow n + 2 \in P$
- Regra de fechamento: Todo $n \in P$ pode ser obtido por um número finito de aplicações de regras anteriores.

■

A seguir apresentamos uma exemplo de definição maior: a definição recursiva do conjunto dos números inteiros.

Exemplo 77. O conjunto dos números naturais possui uma definição direta porquê este é infinito em apenas "uma direção", o que não acontece com o conjunto $\mathbb{Z}$, que possui infinitos elementos positivos e negativos. Então como definir este conjunto recursivamente?

Uma primeira tentativa seria algo similar a dizer que $0 \in \mathbb{Z}$ e que se $n \in \mathbb{Z}$ então $n + 1$ e $n - 1$ também pertencem a $\mathbb{Z}$. Mais formalmente:

- Caso base: $0 \in \mathbb{Z}$.
- Passo recursivo: $\forall x.x \in \mathbb{Z} \rightarrow x + 1 \in \mathbb{Z} \wedge x - 1 \in \mathbb{Z}$

Apesar de correta, a definição anterior possui um inconveniente: existe mais de uma maneira de deduzirmos que um certo número pertence ou não a Z. Como exemplo, considere a seguinte derivação, de que $-2 \in \mathbb{Z}$:

$$\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{}{\forall n.n \in \mathbb{Z} \rightarrow n + 1 \in \mathbb{Z} \wedge n - 1 \in \mathbb{Z}}\,\{I_D\}}{-1 \in \mathbb{Z} \rightarrow 0 \in \mathbb{Z} \wedge -2 \in \mathbb{Z}}\,\{\forall_E\} \quad \dfrac{\dfrac{}{0 \in \mathbb{Z}}\,\{I_D\} \quad \dfrac{\dfrac{}{\forall n.n \in \mathbb{Z} \rightarrow n + 1 \in \mathbb{Z} \wedge n - 1 \in \mathbb{Z}}\,\{I_D\}}{0 \in \mathbb{Z} \rightarrow 1 \in \mathbb{Z} \wedge -1 \in \mathbb{Z}}\,\{\forall_E\}}{1 \in \mathbb{Z} \wedge -1 \in \mathbb{Z}}\,\{\rightarrow_E\}}{-1 \in \mathbb{Z}}\,\{\wedge_{ED}\}}{0 \in \mathbb{Z} \wedge -2 \in \mathbb{Z}}\,\{\rightarrow_E\}}{-2 \in \mathbb{Z}}\,\{\wedge_{ED}\}$$

Veja que da mesma maneira que concluímos que $-2 \in \mathbb{Z}$, poderíamos ter utilizado a regra de $\{\wedge_{EE}\}$ para concluir que $0 \in \mathbb{Z}$. Isto mostra que existe mais de uma maneira de deduzir que $0 \in \mathbb{Z}$: uma é utilizando o caso base do conjunto $\mathbb{Z}$ e outra é utilizando uma combinação de passo recursivo e do caso base.

Em termos matemáticos, a definição anterior não é problemática. Porém, definições que geram "elementos repetidos" são inconvenientes computacionalmente pois, esta repetição pode denotar desperdício de tempo de CPU ou de memória (para armazenar as repetições). Desta forma, devemos sempre especificar conjuntos recursivos de maneira que haja uma única maneira de representar qualquer elemento deste conjunto.

Uma definição equivalente que não possui o inconveniente de gerar elementos repetidos é baseada na seguinte observação: se $n \in \mathbb{Z}$ então tanto $n+1$ quanto $-(n+1)$ pertencem a $\mathbb{Z}$. Estes critérios serão utilizados na definição recursiva de $\mathbb{Z}$, apresentada a seguir:

- Caso base: $0 \in \mathbb{Z}$.
- Caso recursivo: $\forall n. n \in \mathbb{Z} \wedge n \geq 0 \rightarrow (n+1) \in \mathbb{Z} \wedge -(n+1) \in \mathbb{Z}$
- Regra de fechamento: Todo $n \in \mathbb{Z}$ pode ser gerado por um número finito de aplicações das regras anteriores.

É útil que o leitor tente verificar que a derivação de $-2 \in \mathbb{Z}$, utilizando esta última definição não possui o inconveniente de gerar elementos repetidos. ■

As formas apresentadas de descrever conjuntos não são equivalentes entre si. Evidentemente, não podemos utilizar enumeração para representar conjuntos infinitos. Porém, mesmo as técnicas de set comprehension e recursividade não são equivalentes. Como exemplo, considere o seguinte conjunto:

$$\{x \in \mathbb{R} \,|\, 0 \leq x \leq 1\}$$

Não é possível construir uma definição recursiva para este conjunto, uma vez que, dado um número real x não é possível determinar de maneira única qual seria o "sucessor"[5] de x na reta real.

Outra maneira de descrever conjuntos é definindo-os utilizando operações sobre conjuntos existentes. Este é o assunto da próxima seção.

5.3.4 Exercícios

1. Apresente uma definição recursiva do conjunto de números naturais ímpares.
2. Apresente uma definição recursiva do conjunto de números inteiros múltiplos de 5.
3. Uma sequência é um palíndromo se esta pode ser lida da mesma maneira da esquerda para direita quanto da direita para esquerda. Apresente uma definição recursiva do conjunto $\mathbb{P}$, que consiste de todos os palíndromos de bits (formados apenas pelos bits 0 e 1).

[5] Este uso da palavra sucessor é um abuso de linguagem.

5.4 Operações Sobre Conjuntos

Existem diversas operações que podem ser aplicadas a conjuntos. Seja para criar outros conjuntos ou mesmo para compará-los. As próximas subseções apresentam estas propriedades.

5.4.1 Subconjuntos e Igualdade de Conjuntos

Existem diversas relações entre conjuntos que são determinadas pelos elementos que estes compartilham. Uma destas operações é a de *continência*. A expressão $A \subseteq B$, que pode ser lida como "A está contido em B", é verdadeira se todo elemento de A é também elemento de B. Esta idéia é definida formalmente a seguir:

Definição 34 (Continência)**.** Sejam A e B dois conjuntos quaisquer. Dizemos que $A \subseteq B$ se e somente se

$$\forall x.x \in A \rightarrow x \in B$$

■

Por sua vez, dizemos que dois conjuntos são iguais se estes possuem exatamente os mesmos elementos. A seguinte definição formaliza este conceito.

Definição 35 (Igualdade)**.** Sejam A e B dois conjuntos quaisquer. Dizemos que $A = B$ se e somente se:

$$\forall x.x \in A \leftrightarrow x \in B$$

■

Utilizando álgebra para lógica de predicados e a definição de $A \subseteq B$, podemos obter uma definição alternativa da igualdade de conjuntos. Esta é demonstrada a seguir:

$$\begin{array}{ll} A = B & \equiv \\ \forall x.x \in A \leftrightarrow x \in B & \equiv \\ \forall x.(x \in A \rightarrow x \in B) \wedge (x \in B \rightarrow x \in A) & \equiv \\ (\forall x.x \in A \rightarrow x \in B) \wedge (\forall x.x \in B \rightarrow x \in A) & \equiv \\ A \subseteq B \wedge B \subseteq A & \end{array}$$

Logo, podemos concluir que $A = B$ é equivalente a $A \subseteq B \wedge B \subseteq A$.

A definição da igualdade de conjuntos implica que para dois conjuntos serem considerados diferentes, um deles deve possuir pelo menos um elemento que o outro não possui. Note que se $A \neq B$ e $A \subseteq B$, podemos dizer que existe y pertencente a B tal que $y \notin A$. Esta noção é definida formalmente a seguir.

Definição 36 (Subconjunto Próprio)**.** Sejam A e B dois conjuntos quaisquer. Dizemos que A é um subconjunto próprio de B, $A \subset B$, se e somente se $A \subseteq B$ e $A \neq B$. ■

5.4.2 União, Interseção, Complemento e Diferença de Conjuntos

Nesta seção descreveremos formalmente operações sobre conjuntos que já devem ser conhecidas pelo leitor. Para todas as operações, considere que os conjuntos A e B são subconjuntos de um conjunto universo $\mathcal{U}$.

- A união de dois conjuntos A e B, $A \cup B$, é o conjunto que contém todos os elementos que estão em A ou em B (ou ambos). Todo elemento de $A \cup B$ deve pertencer a A ou B ou ambos.
- A interseção de dois conjuntos A e B, $A \cap B$, é o conjunto que contém todos os elementos que estão em A e em B.
- O complemento de um conjunto A, $\overline{A}$, é o conjunto de todos elementos que pertencem ao conjunto universo $\mathcal{U}$ e não pertencem a A.
- A diferença de dois conjuntos A e B, $A - B$, é o conjunto de todos os elementos que estão em A, mas não estão em B.

A seguir, definimos estas operações de maneira precisa.

Definição 37 (União, interseção e diferença)**.** Sejam A e B dois conjuntos quaisquer. Então:

- $A \cup B = \{x \mid x \in A \vee x \in B\}$
- $A \cap B = \{x \mid x \in A \wedge x \in B\}$
- $\overline{A} = \{x \mid x \in \mathcal{U} \wedge x \notin A\}$
- $A - B = \{x \mid x \in A \wedge x \notin B\}$

Note que $\overline{A} = \mathcal{U} - A$. Dizemos que A e B são *disjuntos* se $A \cap B = \emptyset$ ■

Exemplo 78. Sejam $A = \{1, 2, 3\}$, $B = \{3, 4, 5\}$, $C = \{4, 5, 6\}$ e $\mathcal{U} = \{1, 2, 3, 4, 5, 6, 7\}$. Então:

$$\begin{array}{lcl} A \cup B & = & \{1, 2, 3, 4, 5\} \\ A \cap B & = & \{3\} \\ A - B & = & \{1, 2\} \\ A \cup C & = & \{1, 2, 3, 4, 5, 6\} \\ A \cap C & = & \emptyset \\ A - C & = & \{1, 2, 3\} \\ \overline{A} & = & \{4, 5, 6, 7\} \end{array}$$

■

5.4.3 Famílias de Conjuntos

Damos o nome de família conjuntos que possuem como elementos outros conjuntos contidos em um universo $\mathcal{U}$. Um exemplo de família é o chamado conjunto potência ou conjunto das partes de um conjunto, definido a seguir.

Definição 38 (Conjunto Potência)**.** Seja A um conjunto qualquer. Denomina-se conjunto potência ou conjunto das partes o conjunto de todos os subconjuntos de A. Representamos este conjunto por $\mathcal{P}(A)$. Mais formalmente, o conjunto potência é definido como:

$$\mathcal{P}(A) = \{X \mid X \subseteq A\}$$

■

Exemplo 79. Sejam $A = \{1, 2\}$ e $B = \emptyset$. Temos que $\mathcal{P}(A) = \{\emptyset, \{1\}, \{2\}, \{1, 2\}\}$ e $\mathcal{P}(B) = \{\emptyset\}$. ■

Note que se $|A| = n$, para algum $n \in \mathbb{N}$, então $|\mathcal{P}(A)| = 2^n$.

As operações de união e interseção de conjuntos se estendem naturalmente para famílias de conjuntos. A definição destas operações é apresentada a seguir.

Definição 39 (União e Interseção de Famílias)**.** Seja $\mathcal{F}$ uma família não vazia de conjuntos. A união e interseção da família $\mathcal{F}$ são definidas como:

$$\begin{array}{rcl} \bigcup \mathcal{F} & = & \{x \mid \exists A.A \in \mathcal{F} \wedge x \in A\} \\ \bigcap \mathcal{F} & = & \{x \mid \forall A.A \in \mathcal{F} \rightarrow x \in A\} \end{array}$$

■

Exemplo 80. Seja $\mathcal{F} = \{\{1, 2, 3\}, \{2, 3, 4\}, \{3, 4, 5\}\}$ uma família de conjuntos. Temos que:

$$\begin{array}{rcllcl} \bigcap \mathcal{F} & = & \{1, 2, 3\} \cap \{2, 3, 4\} \cap \{3, 4, 5\} & = & \{3\} \\ \bigcup \mathcal{F} & = & \{1, 2, 3\} \cup \{2, 3, 4\} \cup \{3, 4, 5\} & = & \{1, 2, 3, 4, 5\} \end{array}$$

■

Finalmente, uma notação alternativa para famílias de conjuntos são as chamadas *famílias indexadas*, que são definidas em termos de um conjunto de índices.

Definição 40 (Famílias Indexadas)**.** Seja I um conjunto não vazio, denominado conjunto de índices. Denominamos por família indexada o conjunto

$$\mathcal{F} = \{A_i \mid i \in I\}$$

em que cada A_i é definido em termos dos elementos do conjunto de índices. A união e interseção de famílias indexadas é formalizada como:

$$\begin{array}{rcl} \bigcup_{i \in I} A_i & = & \{x \mid \exists i.i \in I \wedge x \in A_i\} \\ \bigcap_{i \in I} A_i & = & \{x \mid \forall i.i \in I \rightarrow x \in A_i\} \end{array}$$

■

Exemplo 81. Considere o seguinte conjunto de índices $I = \{1, 2, 3\}$ e a família indexada $\mathcal{F} = \{A_i \mid i \in I\}$, em que $A_i = \{i, i + 1, i + 2\}$. Temos:

$$\begin{array}{rcllcl} \mathcal{F} & = & \{A_1, A_2, A_3\} & = & \{\{1, 2, 3\}, \{2, 3, 4\}, \{3, 4, 5\}\} \\ \bigcup_{i \in \{1,2,3\}} & = & \{1, 2, 3, 4, 5\} \\ \bigcap_{i \in \{1,2,3\}} & = & \{3\} \end{array}$$

■

5.4.4 Exercícios

1. Represente as seguintes fórmulas expressas utilizando a linguagem da teoria de conjuntos utilizando fórmulas da lógica de predicados. Você poderá utilizar apenas os seguintes símbolos em suas respostas: $\in, \notin, =, \neq, \wedge, \vee, \rightarrow, \leftrightarrow, \forall, \exists$. Observe que não é permitido utilizar $\neg$, logo, você deverá utilizar equivalências algébricas para eliminar as ocorrências de $\neg$.

   (a) $\mathcal{F} \subseteq \mathcal{P}(A)$
   (b) $A \subseteq \{2n \mid n \in \mathbb{N}\}$
   (c) $\{n^2 + n + 1 \mid n \in \mathbb{N}\} \subseteq \{2n + 1 \mid n \in \mathbb{N}\}$
   (d) $\mathcal{P}(\bigcup_{i \in I} A_i) \not\subseteq \bigcup_{i \in I} \mathcal{P}(A_i)$
   (e) $x \in \bigcup \mathcal{F} - \mathcal{G}$
   (f) $\{x \in B \mid x \notin C\} \in \mathcal{P}(A)$
   (g) $x \in \bigcap_{i \in I}(A_i \bigcup B_i)$
   (h) $x \in (\bigcap_{i \in I} A_i) \cup (\bigcap_{i \in I} B_i)$

2. Seja $I = \{2, 3, 4, 5\}$ e para cada $i \in I$ considere que $A_i = \{i, i+1, i-1, 2i\}$.

   (a) Liste os elementos de $\mathcal{F} = \{A_i \mid i \in I\}$.
   (b) Calcule $\bigcap_{i \in I} A_i$ e $\bigcup_{i \in I} A_i$.

3. Mostre, utilizando equivalências algébricas da lógica, que $x \in \mathcal{P}(A \cap B)$ é equivalente a $x \in \mathcal{P}(A) \cap \mathcal{P}(B)$, para qualquer x.

4. Apresente exemplos de conjuntos A e B tais que $\mathcal{P}(A \cup B) \neq \mathcal{P}(A) \cup \mathcal{P}(B)$.

5. Mostre que se $\mathcal{F} = \emptyset$ então a fórmula $x \in \bigcup \mathcal{F}$ é equivalente a F (contradição).

6. Mostre que se $\mathcal{F} = \emptyset$ então a fórmula $x \in \bigcap \mathcal{F}$ é equivalente a T (tautologia).

5.5 Leis Algébricas para Conjuntos

Como operações sobre conjuntos são definidas usando fórmulas da lógica (set comprehension), leis algébricas da lógica aplicam-se a expressões envolvendo conjuntos. A tabela seguinte apresenta as principais equivalências algébricas para conjuntos (em que $\circ \in \{\cap, \cup\}$).

$A \circ A$	$\equiv$	A	$A \cup \overline{A}$	$\equiv$	$\mathcal{U}$
$A \circ B$	$\equiv$	$B \circ A$	$\overline{A \cap B}$	$\equiv$	$\overline{A} \cup \overline{B}$
$(A \circ B) \circ C$	$\equiv$	$A \circ (B \circ C)$	$\overline{A \cup B}$	$\equiv$	$\overline{A} \cap \overline{B}$
$A \cup (B \cap C)$	$\equiv$	$(A \cup B) \cap (A \cup C)$	$A \cap \emptyset$	$\equiv$	$\emptyset$
$A \cap (B \cup C)$	$\equiv$	$(A \cap B) \cup (A \cap C)$	$A \cap \overline{A}$	$\equiv$	$\emptyset$
$A \cup \emptyset$	$\equiv$	A	$A \cap \mathcal{U}$	$\equiv A$	
$A - B$	$\equiv$	$A \cap \overline{B}$			

Devido a similaridade das leis algébricas para conjuntos com as da lógica, apresentaremos apenas alguns exemplos que ilustram sua utilização.

Exemplo 82. Considere a seguinte equivalência

$$[A \cup (B \cap C)] \cap \{[\overline{A} \cup (B \cap C)] \cap \overline{(B \cap C)}\} \equiv \emptyset$$

cuja demonstração apresentamos abaixo:

$$\begin{array}{ll} [A \cup (B \cap C)] \cap \{[\overline{A} \cup (B \cap C)] \cap \overline{(B \cap C)}\} & \equiv \\ [A \cup (B \cap C)] \cap \{[\overline{A} \cap \overline{(B \cap C)}] \cup [(B \cap C) \cap \overline{(B \cap C)}]\} & \equiv \\ [A \cup (B \cap C)] \cap \{[\overline{A} \cap \overline{(B \cap C)}] \cup \emptyset\} & \equiv \\ [A \cup (B \cap C)] \cap [\overline{A} \cap \overline{(B \cap C)}] & \equiv \\ [A \cup (B \cap C)] \cap \overline{[A \cup (B \cap C)]} & \equiv \\ \emptyset & \end{array}$$

■

Exemplo 83. Considere a seguinte equivalência

$$[C \cap (A \cup B)] \cup [(A \cup B) \cap \overline{C}] \equiv A \cup B$$

cuja demonstração apresentamos a seguir:

$$\begin{array}{ll} [C \cap (A \cup B)] \cup [(A \cup B) \cap \overline{C}] & \equiv \\ [(A \cup B) \cap C] \cup [(A \cup B) \cap \overline{C}] & \equiv \\ (A \cup B) \cap (C \cup \overline{C}) & \equiv \\ (A \cup B) \cap \mathcal{U} & \equiv \\ A \cup B & \end{array}$$

■

5.5.1 Exercícios

1. Demonstre as seguintes equivalências algébricas para conjuntos.
   (a) $(A \cup B) \cap (A \cup \overline{B}) \equiv A$
   (b) $A \cap (B \cup \overline{A}) \equiv B \cap A$
   (c) $(A \cup B) - C \equiv (A - C) \cup (B - C)$
   (d) $\overline{[(\overline{A} \cup \overline{B}) \cap \overline{A}]} \equiv A$

5.6 Teoremas Envolvendo Conjuntos

As técnicas de demonstração apresentadas no capítulo 4 podem ser utilizadas para provar diversos fatos da teoria de conjuntos. Para isso, representaremos os fatos expressando fórmulas da teoria de conjunto como fórmulas da lógica de predicados, utilizando as definições apresentadas neste capítulo.

O restante desta seção apresenta diversos teoremas envolvendo conjuntos e suas demonstrações. Inicialmente, consideraremos rascunhos com um maior nível de detalhes para um maior entendimento do leitor. Gradativamente, menos detalhes serão fornecidos até que apresentemos somente o texto final para um dado teorema. Nestes casos, recomendamos que o leitor "preencha" os detalhes omitidos ou mesmo reconstrua todo o rascunho da demonstração em questão.

Exemplo 84. Como um primeiro exemplo, considere a tarefa de demonstrar o seguinte teorema.

Sejam A, B e C conjuntos quaisquer. Então se $A \subseteq B$ e $B \subseteq C$ então $A \subseteq C$.

Primeiramente, temos que o teorema anterior possui como hipóteses que A, B e C são conjuntos e a conclusão:

$$A \subseteq B \wedge B \subseteq C \rightarrow A \subseteq C$$

A partir disto, podemos montar uma versão inicial do rascunho deste teorema:

Hipóteses	Conclusão
A, B, C são conjuntos	$A \subseteq B \wedge B \subseteq C \rightarrow A \subseteq C$

Evidentemente, esta prova deverá iniciar utilizando a estratégia de prova direta para implicação, que produz a seguinte configuração do rascunho:

Hipóteses	Conclusão
A, B, C são conjuntos	$A \subseteq C$
$A \subseteq B$	
$B \subseteq C$	

Para demonstrar $A \subseteq C$, devemos expressá-la utilizando sua definição usando lógica (o mesmo vale para as hipóteses):

Hipóteses	Conclusão
A, B, C são conjuntos	$\forall x.x \in A \rightarrow x \in C$
$\forall y.y \in A \rightarrow y \in B$	
$\forall z.z \in B \rightarrow z \in C$	

Agora, utilizamos a estratégia de prova para o quantificador universal e mais uma aplicação de prova direta, o que nos leva a:

Hipóteses	Conclusão
A, B, C são conjuntos	$x \in C$
$\forall y.y \in A \rightarrow y \in B$	
$\forall z.z \in B \rightarrow z \in C$	
x arbitrário	
$x \in A$	

Neste ponto, podemos utilizar a estratégia de uso de hipóteses envolvendo o quantificador universal (regra de eliminação deste quantificador), obtendo:

Hipóteses	Conclusão
A, B, C são conjuntos	$x \in C$
$\forall y.y \in A \rightarrow y \in B$	
$\forall z.z \in B \rightarrow z \in C$	
x arbitrário	
$x \in A$	
$x \in A \rightarrow x \in B$	

Usando uma vez a regra de eliminação da implicação, obtemos que $x \in B$, conforme apresentado a seguir:

Hipóteses	Conclusão
A, B, C são conjuntos	$x \in C$
$\forall y.y \in A \rightarrow y \in B$	
$\forall z.z \in B \rightarrow z \in C$	
x arbitrário	
$x \in A$	
$x \in A \rightarrow x \in B$	
$x \in B$	

Eliminando novamente o quantificador universal, obtemos:

Hipóteses	Conclusão
A, B, C são conjuntos	$x \in C$
$\forall y.y \in A \rightarrow y \in B$	
$\forall z.z \in B \rightarrow z \in C$	
x arbitrário	
$x \in A$	
$x \in A \rightarrow x \in B$	
$x \in B$	
$x \in B \rightarrow x \in C$	

A demonstração é concluída por uma eliminação da implicação que permite-nos deduzir que $x \in C$.

O texto para esta dedução é apresentado a seguir.

Suponha que A, B, C são conjuntos quaisquer.
Suponha que $A \subseteq B$ e $B \subseteq C$.
Suponha x arbitrário.
Suponha $x \in A$.
Como $x \in A$ e $A \subseteq B$, temos que $x \in B$.
Como $x \in B$ e $B \subseteq C$, temos que $x \in C$.
Logo, se $x \in A$ então $x \in C$.
Como x é arbitrário, temos que $A \subseteq C$.
Portanto, se $A \subseteq B$ e $B \subseteq C$ então $A \subseteq C$.
Logo, se A, B, C são conjuntos e se $A \subseteq B$ e $B \subseteq C$ então $A \subseteq C$.

■

Exemplo 85. Considere o seguinte teorema:

Suponha A, B e C conjuntos tais que $A - B \subseteq C$. Se $x \in A - C$ então $x \in B$.

Neste teorema, temos como hipóteses que A, B, C são conjuntos e $A - B \subseteq C$. A conclusão deste é expressa pela seguinte fórmula:

$$x \in A - C \rightarrow x \in B$$

Inicialmente, o rascunho possui a seguinte forma:

Hipóteses	Provar
A, B, C são conjuntos	$x \in A - C \rightarrow x \in B$
$A - B \subseteq C$	

Usando a estratégia de prova direta, temos:

Hipóteses	Provar
A, B, C são conjuntos	$x \in B$
$A - B \subseteq C$	
$x \in A - C$	

Note que se $x \in A - C$, temos que $x \in A$ e $x \notin C$[6]:

Hipóteses	Provar
A, B, C são conjuntos	$x \in B$
$A - B \subseteq C$	
$x \in A - C$	
$x \in A$	
$x \notin C$	

Aparentemente não há como deduzir que $x \in B$ a partir das hipóteses. Neste caso, podemos tentar uma prova por contradição.

Hipóteses	Provar
A, B, C são conjuntos	$\bot$
$A - B \subseteq C$	
$x \in A - C$	
$x \in A$	
$x \notin C$	
$x \notin B$	

Uma vez que $x \in A$ e $x \notin B$, temos que $x \in A - B$.

Hipóteses	Provar
A, B, C são conjuntos	$\bot$
$A - B \subseteq C$	
$x \in A - C$	
$x \in A$	
$x \notin C$	
$x \notin B$	
$x \in A - B$	

Como $x \in A - B$ e $A - B \subseteq C$, temos que $x \in C$, o que contradiz a suposição de que $x \notin C$, concluindo a demonstração. Apresentamos o texto desta demonstração a seguir.

Suponha A, B e C são conjuntos e que $A - B \subseteq C$.
  Suponha que $x \in A - C$.
    Suponha que $x \notin B$.
      Como $x \in A - C$, temos que $x \in A$ e $x \notin C$.
      Como $x \in A$ e $x \notin B$, temos que $x \in A - B$.
      Como $x \in A - B$ e $A - B \subseteq C$, temos que $x \in C$.
      Como $x \in C$ e $x \notin C$, temos uma contradição.
    Assim, temos que $x \in B$.
  Logo, se $x \in A - C$ então $x \in B$.
Portanto, A, B e C são conjuntos e que $A - B \subseteq C$ então se $x \in A - C$, temos que $x \in B$

[6] O leitor atento deve ter notado que isto é uma consequência da representação de pertinência a conjuntos definidos por set comprehension. Isto é, representamos $y \in \{x \in A \mid P(x)\}$ como $y \in A \wedge P(y)$.

■

Exemplo 86. Considere o seguinte teorema:

Suponha que $A \cap C \subseteq B$ e $a \in C$. Então $a \notin A - B$.

Este teorema possui como hipóteses os fatos que $A \cap C \subseteq B$ e que $a \in C$. A conclusão deste teorema pode ser expressa pela seguinte fórmula da teoria de conjuntos: $a \notin A - B$. Temos a seguinte versão inicial do rascunho:

Hipóteses	Conclusão
$A \cap C \subseteq B$	$a \notin A - B$.
$a \in C$	

Expressando a conclusão como uma fórmula da lógica, obtemos:

Hipóteses	Conclusão
$A \cap C \subseteq B$	$\neg(a \in A \wedge a \notin B)$.
$a \in C$	

Utilizando equivalências algébricas podemos mostrar que $\neg(a \in A \wedge a \notin B) \equiv a \in A \rightarrow a \in B$, conforme deduzido a seguir:

$$\begin{array}{ll} \neg(a \in A \wedge a \notin B) & \equiv \\ a \notin A \vee \neg a \notin B & \equiv \\ a \notin A \vee a \in B & \equiv \\ a \in A \rightarrow a \in B & \end{array}$$

Utilizando esta equivalência, obtemos:

Hipóteses	Conclusão
$A \cap C \subseteq B$	$a \in A \rightarrow a \in B$.
$a \in C$	

Agora, utilizaremos a estratégia de prova direta para implicação, obtendo:

Hipóteses	Conclusão
$A \cap C \subseteq B$	$a \in B$.
$a \in C$	
$a \in A$	

Uma vez que $a \in A$ e $a \in C$, temos que $a \in A \cap C$ e como $A \cap C \subseteq B$, podemos concluir que $a \in B$.

A seguir, apresentamos o texto desta demonstração.

Suponha que $A \cap C \subseteq B$ e $a \in C$.
  Suponha que $a \in A$.
    Como $a \in A$ e $a \in C$, temos que $a \in A \cap C$.
    Como $a \in A \cap C$ e $A \cap C \subseteq B$ então $a \in B$.
  Logo, se $a \in A$ então $a \notin A - B$.
Portanto, se $A \cap C \subseteq B$ e $a \in C$ então $a \notin A - B$.

Vale lembrar que manipulações usando álgebra booleana não devem fazer parte da demonstração final. Note que no texto deste exemplo, consideramos que a conclusão a ser provada era $a \in A \rightarrow a \in B$ ao invés da equivalente $a \notin A - B$. ■

Exemplo 87. Considere o seguinte teorema:

Suponha que A e B são conjuntos. Se $A \cap B = A$ então $A \subseteq B$.

Este teorema possui como hipóteses os fatos de que A e B são conjuntos e conclusão a fórmula:

$$A \cap B = A \rightarrow A \subseteq B$$

Logo, temos o seguinte rascunho inicial:

Hipóteses	Conclusão
A e B são conjuntos	$A \cap B = A \rightarrow A \subseteq B$

Usando prova direta, temos:

Hipóteses	Conclusão
A e B são conjuntos	$A \subseteq B$
$A \cap B = A$	

Para continuar a demonstração, temos que representar a expressão $A \subseteq B$ como uma fórmula da lógica de predicados.

Hipóteses	Conclusão
A e B são conjuntos	$\forall x. x \in A \rightarrow x \in B$
$A \cap B = A$	

Como a conclusão possui um quantificador universal, utilizaremos a estratégia de prova para este.

Hipóteses	Conclusão
A e B são conjuntos	$x \in B$
$A \cap B = A$	
x arbitrário	
$x \in A$	

Agora, como $A \cap B = A$ e $x \in A$, temos que $x \in A \cap B$ e, portanto, $x \in B$, conforme requerido.

O texto para esta demonstração é apresentado a seguir.

Suponha que $A \cap B = A$.
  Suponha x arbitrário.
    Suponha que $x \in A$.
      Como $x \in A$ e $A \cap B = A$, temos que $x \in A \cap B$.
      Como $x \in A \cap B$ temos que $x \in B$.
    Logo, se $x \in A$ então $x \in B$.
  Como x é arbitrário, temos que $A \subseteq B$.
Portanto, se $A \cap B = A$ então $A \subseteq B$.

■

Exemplo 88. Considere o seguinte teorema:

Suponha que B é um conjunto e $\mathcal{F}$ é uma família. Se $\bigcup \mathcal{F} \subseteq B$ então $\mathcal{F} \subseteq \mathcal{P}(B)$.

Este é o primeiro exemplo que utiliza operações sobre famílias de conjuntos e por isso iremos apresentá-lo em maiores detalhes. Iniciaremos, como de costume, com o rascunho.

Hipóteses	Conclusão
B é um conjunto	$\bigcup \mathcal{F} \subseteq B \rightarrow \mathcal{F} \subseteq \mathcal{P}(B)$.
$\mathcal{F}$ é uma família.	

Usando a técnica de prova direta, temos:

Hipóteses	Conclusão
B é um conjunto	$\mathcal{F} \subseteq \mathcal{P}(B)$.
$\mathcal{F}$ é uma família.	
$\bigcup \mathcal{F} \subseteq B$	

Representando a conclusão usando uma fórmula da lógica de predicados, temos:

Hipóteses	Conclusão
B é um conjunto	$\forall x. x \in \mathcal{F} \rightarrow x \in \mathcal{P}(B)$.
$\mathcal{F}$ é uma família.	
$\bigcup \mathcal{F} \subseteq B$	

Agora, usando a estratégia para o quantificador universal, e prova direta temos:

Hipóteses	Conclusão
B é um conjunto	$x \in \mathcal{P}(B)$.
$\mathcal{F}$ é uma família.	
$\bigcup \mathcal{F} \subseteq B$	
x arbitrário	
$x \in \mathcal{F}$	

O próximo passo da demonstração é representar $x \in \mathcal{P}(B)$ como uma fórmula da lógica, isso será feito em dois passos. Primeiro, representamos esta fórmula usando uma equivalente da teoria de conjuntos.

Hipóteses	Conclusão
B é um conjunto	$x \subseteq B$.
$\mathcal{F}$ é uma família.	
$\bigcup \mathcal{F} \subseteq B$	
x arbitrário	
$x \in \mathcal{F}$	

Agora, representando como uma fórmula da lógica obtemos:

Hipóteses	Conclusão
B é um conjunto	$\forall y. y \in x \rightarrow y \in B$
$\mathcal{F}$ é uma família.	
$\bigcup \mathcal{F} \subseteq B$	
x arbitrário	
$x \in \mathcal{F}$	

Usando novamente as estratégias de prova para implicação e quantificador universal, temos:

Hipóteses	Conclusão
B é um conjunto	$y \in B$
$\mathcal{F}$ é uma família.	
$\bigcup \mathcal{F} \subseteq B$	
x arbitrário	
$x \in \mathcal{F}$	
y arbitrário	
$y \in x$	

Nosso próximo passo nesta demonstração é utilizar a hipótese $\bigcup \mathcal{F} \subseteq B$.

Hipóteses	Conclusão
B é um conjunto	$y \in B$
$\mathcal{F}$ é uma família.	
$\forall z.z \in \bigcup \mathcal{F} \rightarrow z \in B$	
x arbitrário	
$x \in \mathcal{F}$	
y arbitrário	
$y \in x$	

Como $y \in x$ e $x \in \mathcal{F}$, temos que $y \in \bigcup \mathcal{F}$[7]. Usando esta dedução, obtemos:

Hipóteses	Conclusão
B é um conjunto	$y \in B$
$\mathcal{F}$ é uma família.	
$\forall z.z \in \bigcup \mathcal{F} \rightarrow z \in B$	
x arbitrário	
$x \in \mathcal{F}$	
y arbitrário	
$y \in x$	
$y \in \bigcup \mathcal{F}$	

Agora, basta utilizar uma eliminação do quantificador universal sobre a hipótese $\forall z.z \in \bigcup \mathcal{F} \rightarrow z \in B$ substituindo z por y. Com isso, obtemos:

Hipóteses	Conclusão
B é um conjunto	$y \in B$
$\mathcal{F}$ é uma família.	
$\forall z.z \in \bigcup \mathcal{F} \rightarrow z \in B$	
x arbitrário	
$x \in \mathcal{F}$	
y arbitrário	
$y \in x$	
$y \in \bigcup \mathcal{F}$	
$y \in \bigcup \mathcal{F} \rightarrow y \in B$	

Finalmente, concluímos esta demonstração utilizando a regra de eliminação da implicação. A seguir, apresentamos o texto correspondente a esta demonstração.

Suponha que $\bigcup \mathcal{F} \subseteq B$.
  Suponha x arbitrário.

[7] Este passo de demonstração é equivalente a utilizar a regra $\{\exists_I\}$ da dedução natural.

Suponha $x \in \mathcal{F}$.
Suponha y arbitrário.
Suponha $y \in x$.
Como $y \in x$ e $x \in \mathcal{F}$, temos que $y \in \bigcup \mathcal{F}$.
Como $y \in \bigcup \mathcal{F}$ e $\bigcup \mathcal{F} \subseteq B$, temos que $y \in B$.
Logo, se $y \in x$ então $y \in B$.
Como y é arbitrário, temos que $x \subseteq B$.
Logo, se $x \in \mathcal{F}$, então $x \in \mathcal{P}(B)$.
Como x é arbitrário, temos que $\mathcal{F} \subseteq \mathcal{P}(B)$.
Portanto, se $\bigcup \mathcal{F} \subseteq B$ então $\mathcal{F} \subseteq \mathcal{P}(B)$.

■

O leitor deve ter notado que todos os exemplos apresentados utilizam a definição de notações da teoria de conjuntos usando set comprehension e usam a seguinte regra para representação, em lógica, de pertinência:

$$y \in \{x \in A \mid P(x)\} \equiv y \in A \wedge P(y)$$

Sabendo-se como representar notações da teoria de conjuntos como expressões da lógica, a tarefa de demonstrar teoremas sobre conjuntos é basicamente utilizar as estratégias de prova de maneira quase mecânica.

5.6.1 Exercícios

1. Prove que, se $A \subseteq B$ e $B \subseteq C$, então $A \subseteq C$.
2. Prove que, se $\overline{A} \subseteq \overline{B}$ então $B \subseteq A$.
3. Prove que, se $A \subseteq B$ então $\mathcal{P}(A) \subseteq \mathcal{P}(B)$.
4. Prove que, $\mathcal{P}(A) \cap \mathcal{P}(B) = \mathcal{P}(A \cap B)$
5. Prove que, $\mathcal{P}(A) \cup \mathcal{P}(B) \subseteq \mathcal{P}(A \cup B)$
6. Prove que, se $(A - B) \cup (B - A) = A \cup B$ então $A \cap B = \emptyset$ (isto é A e B são disjuntos).
7. Prove que, se $A \cup B = A - B$ então $B = \emptyset$.
8. Prove que, se $A \cap B = A$ então $A \subseteq B$.
9. Suponha que $A - B \subseteq C \cap D$ e que $x \in A$. Prove que se $x \notin D$ então $x \in B$.
10. Suponha que $A \subseteq C$ e que B e C são disjuntos. Prove que se $x \in A$ então $x \notin B$.
11. Prove que se A e $B - C$ são disjuntos, então $A \cap B \subseteq C$.
12. Prove que se $\mathcal{F}$ é uma família de conjuntos e $A \in \mathcal{F}$, então $A \subseteq \bigcup \mathcal{F}$.
13. Prove que $A = \bigcup \mathcal{P}(A)$, para qualquer conjunto A.
14. Prove que se $\emptyset \in \mathcal{F}$ então $\bigcap \mathcal{F} = \emptyset$.

15. Seja $\mathcal{F}$ uma família de conjuntos. Define-se o conjunto $\bigcup!\mathcal{F}$ por:

$$\bigcup!\mathcal{F} = \{x \mid \exists!A.A \in \mathcal{F} \wedge x \in A\}$$

Prove que para qualquer família $\mathcal{F}$, $\bigcup!\mathcal{F} \subseteq \bigcup \mathcal{F}$.

5.7 Notas Bibliográficas

Grande parte deste capítulo consiste de definições da teoria de conjuntos que podem ser encontradas em qualquer livro de matemática discreta, como por exemplo [9].

Todos os exemplos de teoremas envolvendo conjuntos podem ser encontrados em [9].

6

Combinatória Elementar

> Counting pairs is the oldest trick in combinatorics... Every time we count pairs, we learn something from it.
>
> Gil Kalai, Matemático Israelense.

6.1 Motivação

A combinatória é o ramo da matemática que estabelece métodos para determinar o número de elementos de conjuntos finitos. Mas, qual o interesse de estudantes de computação em se determinar o número de elementos de um certo conjunto?

Uma das várias aplicações da combinatória em computação é o projeto e análise de algoritmos. Diversos problemas podem ser caracterizados por possuírem um conjunto de soluções e, em tais problemas, estamos interessados na "melhor" solução deste conjunto. Algoritmos que resolvem este tipo de problemas possuem um custo computacional proporcional ao tamanho do conjunto de soluções para o problema em questão. Logo, para sermos capazes de entender o comportamento de algoritmos para certos problemas, devemos ser capazes de determinar, de maneira precisa, o número de elementos de conjuntos finitos.

Desta forma, o objetivo deste capítulo é o estudo de técnicas para determinar o número de elementos de conjuntos finitos.

6.2 Noções Básicas de Combinatória

6.2.1 Princípio Multiplicativo

O princípio multiplicativo permite-nos determinar o número de elementos de um conjunto construído por tarefas[1] separadas.

Definição 41 (Princípio Multiplicativo)**.** Suponha que um procedimento possa ser dividido em uma sequência de duas tarefas. Se houver n_1 formas de se fazer

[1] Alguns autores chamam tarefas de "eventos".

a primeira tarefa e para cada uma destas houver n_2 formas de se fazer a segunda tarefa, então há $n_1 \times n_2$ formas de se concluir este procedimento. ■

A seguir apresentamos alguns exemplos que ilustram o uso do princípio multiplicativo.

Exemplo 89. Uma empresa que possui apenas dois empregados, Asdrúbal e Credirceu, alugou um andar de um prédio com 12 salas. De quantas maneiras podemos distribuir as salas deste andar para estes dois funcionários?

Solução: O procedimento para atribuir salas deve funcionar da seguinte maneira: Primeiramente, devemos selecionar uma das 12 salas para Asdrúbal e, na sequência, escolher uma das 11 salas restantes para Credirceu. Logo, temos um total de 12×11 possibilidades para distribuição de salas entre estes funcionários. ■

Exemplo 90. As cadeiras de um auditório devem ser etiquetadas com uma letra e um número inteiro positivo que não exceda a 100. Qual é o maior número de cadeiras que podem ser etiquetadas de maneira diferente?

Solução: Para etiquetarmos uma cadeira devemos selecionar uma das 26 letras do alfabeto e, para cada uma destas, devemos selecionar um número inteiro entre 1 e 100. Logo, temos um total de 26×100 possibilidades. ■

Exemplo 91. Há 32 computadores em uma sala de aula. Cada computador tem 24 portas para conexões de dispositivos. Quantas possibilidades existem nesta sala para a conexão de um certo dispositivo?

Solução: Note que o procedimento para conectar o dispositivo envolve escolher um dos 32 computadores e uma das 24 portas possíveis do computador escolhido. Logo, temos um total de 32×24 possibilidades de conexão. ■

Exemplo 92. Qual o valor da variável teste após a execução do seguinte trecho de código?

```
teste ← 0
for i_1 ← 1 to n_1 do
    for i_2 ← 1 to n_2 do
        ⋮
        for i_m ← 1 to n_m do
            teste ← teste + 1
        end for
    end for
end for
```

Solução: O valor inicial da variável teste é zero. Cada vez que uma repetição é feita, o valor de teste é acrescido de uma unidade. Denomine por T_i a tarefa de executar o i-ésimo laço. Logo, a tarefa de determinar o número de vezes que a variável teste é incrementada é equivalente a tarefa de contar T_i, $1 \leq i \leq m$. Como cada laço é executado n_i $(1 \leq i \leq m)$ vezes, temos que o número de repetições deste algoritmo é

$$\begin{array}{ll} T_1 \times T_2 \times ... \times T_m & = \\ n_1 \times n_2 \times ...n_m & \end{array}$$

Portanto, temos que o valor final da variável teste é $n_1 \times n_2 \times ... \times n_m$. ■

6.2.2 Princípio Aditivo

O princípio aditivo permite-nos determinar o número de elementos de um conjunto construído por tarefas independentes. Dizemos que duas tarefas t_1 e t_2 são independentes se a execução de t_1 não interfere no número de possibilidades de t_2, isto é, os conjuntos relativos a estas tarefas são disjuntos.

Definição 42 (Princípio Aditivo)**.** Se uma tarefa puder ser realizada em uma das n_1 formas ou em uma das n_2 formas, em que nenhum dos elementos do conjunto das n_1 formas pertence ao conjunto das n_2 formas (isto é, estes conjuntos são disjuntos), então há $n_1 + n_2$ formas de se realizar a tarefa. ■

A seguir apresentamos exemplos que ilustram a utilização deste princípio.

Exemplo 93. Suponha que um aluno de mestrado ou um calouro deve ser escolhido para participar de uma comissão em uma certa universidade. Sabendo-se que há 48 alunos de mestrado e 60 calouros, de quantas maneiras podemos escolher o representante desta comissão[2]?

Solução: Como há 48 estudantes de mestrado e 60 calouros e estes conjuntos de alunos são disjuntos, pelo princípio aditivo podemos concluir que o número de maneiras de escolher um representante é de $60 + 48$. ■

Exemplo 94. Diógenes, um estudante de Computação, tem interesse em participar de um projeto de iniciação científica. Sabendo-se que há 17 projetos de computação, 5 de engenharia, 2 de ciências básicas e 3 de administração, quantas possibilidades de escolha de projetos Diógenes possui? Considere que nenhum projeto pode ser classificado em duas áreas distintas.

Solução: Como nenhum dos grupos de projetos possui interseção, podemos utilizar o princípio aditivo para concluir que o número total de possibilidades é de $17 + 5 + 2 + 3$. ■

Exemplo 95. Qual o valor da variável teste após a execução do seguinte trecho de código?

```
teste ← 0
for i_1 ← 1 to n_1 do
    teste ← teste + 1
end for
for i_2 ← 1 to n_2 do
    teste ← teste + 1
end for
⋮
for i_m ← 1 to n_m do
    teste ← teste + 1
end for
```

Solução: O valor inicial da variável teste é zero. Note que a variável teste é incrementada a cada iteração de cada um destes laços. Como o i-ésimo laço executa n_i vezes, temos que a variável teste possuíra o valor final igual a:

$$\sum_{i=1}^{m} n_i = n_1 + n_2 + ... + n_m$$

[2]supondo que, evidentemente, alunos de mestrado não são considerados calouros.

■

6.2.3 Utilizando ambos os princípios

Diversos problemas de combinatória utilizam não apenas um dos dois princípios apresentados nesta seção. Muitos problemas podem ser resolvidos utilizando-se ambos os princípios. Nesta seção apresentaremos alguns exemplos que usam o princípio multiplicativo e aditivo simultaneamente.

Exemplo 96. Crébison deseja desenvolver um compilador para uma linguagem experimental em que todos os nomes de variáveis devem possuir três caracteres, sendo o primeiro uma letra e outros dois quaisquer caracteres alfanuméricos. Quantos nomes diferentes de variáveis são possíveis nesta linguagem projetada por Crébison?

Solução: Note que para o primeiro caractere, temos um total de 26 possibilidades (número de letras do alfabeto). Porém, para cada uma das outras posições temos 36 possibilidades, pois temos 26 letras do alfabeto e 10 dígitos, usando o princípio aditivo. Mas, a tarefa de se escolher um nome de variável pode ser dividida nas seguintes tarefas (em sequência): escolher uma letra, escolher um símbolo alfanumérico e escolher um símbolo alfanumérico. Logo, pelo princípio multiplicativo, temos que o número total de possíveis nomes de variáveis nesta linguagem é $26 \times 36 \times 36$. ■

Exemplo 97. Diosbaldo usa o sistema operacional X, que exige que todas as senhas definidas tenham tamanho 6, 7 ou 8 e que estas possuam pelo menos um dígito. Os demais caracteres podem ser qualquer símbolo alfa numérico. Quantas senhas diferentes podem ser construídas para este sistema operacional?

Solução: Evidentemente, o número total de senhas, S, pode ser expresso como $S = S_6 + S_7 + S_8$, em que S_i é o conjunto de senhas contendo i caracteres. Mas, isso nos deixa com a seguinte questão: como descobrir o valor de cada um dos S_i? Note que S_6 pode ser obtido a partir de todas as sequências de letras e números de tamanho 6 removendo aquelas que não possuem dígitos. Isto é:

$$S_6 = 36^6 - 26^6$$

pois, temos 36^6 possibilidades de sequências de letras e dígitos de tamanho 6 (note que utilizamos os princípios aditivo e multiplicativo para isto) e 26^6 são as sequências formadas apenas por letras. Logo, o número de sequências com pelo menos um dígito é $S_6 = 36^6 - 26^6$. Utilizando um raciocínio similar para S_7 e S_8, podemos concluir que S é igual a:

$$\begin{array}{ll} S_6 + S_7 + S_8 & = \\ (36^6 - 26^6) + (36^7 - 26^7) + (36^8 - 26^8) & \end{array}$$

■

6.2.4 Exercícios

1. Em uma universidade há 30 graduandos em Sistemas de Informação e 20 de Engenharia de Computação.

   (a) De quantas maneiras podemos escolher dois representantes de maneira que um seja de Sistemas de Informação e outro de Engenharia de Computação?
   (b) De quantas maneiras podemos escolher um representante que seja de Sistemas de Informação ou de Engenharia de Computação?

2. Uma avaliação de múltipla escolha contém 10 questões. Há quatro possíveis respostas para cada questão.
   (a) De quantas maneiras um estudante pode responder às questões do exame se este responder a todas elas?
   (b) De quantas maneiras o estudante pode responder às questões se ele pode deixar questões em branco?

3. Quantas sequências de 8 bits são possíveis?

4. Para os itens a seguir, considere os números inteiros positivos entre 100 e 999.
   (a) Quantos são pares?
   (b) Quantos são ímpares?
   (c) Quantos são divisíveis por 5?

5. Quantas sequências de oito letras do alfabeto da língua portuguesa são possíveis
   (a) se as letras puderem ser repetidas?
   (b) se as letras não puderem ser repetidas?
   (c) que comecem e terminem com a letra X?

6. Um palíndromo é uma sequência que pode ser lida da esquerda para direita da mesma forma que da direita para esquerda.
   (a) Quantos palíndromos existem para sequências de 8 bits?
   (b) Quantos palíndromos de tamanho 4 ou 5 podem ser formados utilizando-se as 26 letras do alfabeto?

7. Utilize o princípio multiplicativo para demonstrar que toda fórmula da lógica proposicional que possui n variáveis terá uma tabela verdade com 2^n linhas.

8. Utilize o princípio multiplicativo para demonstrar que existem 2^{2^n} tabelas verdades distintas para proposições envolvendo n variáveis.

6.3 Princípio da Inclusão-Exclusão

Um problema muito comum em combinatória é o de determinar o número de elementos de um conjunto que pode ser expresso como a união de outros.

Suponha que desejamos determinar o número de elementos de um conjunto A em que $A = B \cup C$. Podemos dizer que o número de elementos de A é simplesmente a soma dos números de elementos de B e C, isto é, $|A| = |B|+|C|$?

De modo geral, esta idéia não é válida, conforme apresentado no exemplo seguinte:

Exemplo 98. Considere os seguintes conjuntos: $B = \{1, 2, 3\}$, $C = \{3, 4, 5\}$ e $A = B \cup C$. Note que $|A| = 5$ e não simplesmente a soma da cardinalidade de B e C, que resultaria em 6. ■

Se $A = B \cup C$, não podemos dizer que $|A| = |B| + |C|$, uma vez que estaremos "contando" duas vezes os elementos de $B \cap C$, já que estes aparecem tanto em B quanto em C. Desta forma, para obter a cardinalidade correta de A, temos que "somar apenas uma vez" os elementos de $B \cap C$. Isso nos leva a seguinte equação:

$$|A| = |B \cup C| = |B| + |C| - |B \cap C|$$

Esta equação é apenas um caso específico do chamado princípio da inclusão exclusão, que é apresentado na próxima definição.

Definição 43 (Princípio da Inclusão-Exclusão)**.** Sejam $A_1, A_2, ..., A_n$ n conjuntos finitos. O número de elementos de $\bigcup_{i=1}^{n} A_i$ é dado pela seguinte equação:

$$\begin{array}{rcll} |\bigcup_{i=1}^{n} A_i| & = & \sum_{1 \leq i \leq ni} |A_i| & - \\ & & \sum_{1 \leq i < j \leq n} |A_i \cap A_j| & + \\ & & \sum_{1 \leq i < j < k \leq n} |A_i \cap A_j \cap A_k| & - \\ & & \sum_{1 \leq i < j < k < p \leq n} |A_i \cap A_j \cap A_k \cap A_p| & + \\ & & ... & \\ & & +(-1)^{n-1} |\bigcap_{1 \leq i \leq n} A_i| & \end{array}$$

■

Exemplo 99. Utilizando o princípio da inclusão-exclusão podemos definir uma fórmula para o número de elementos da união de 3 conjuntos:

$$|A \cup B \cup C| = |A| + |B| + |C| - |A \cap B| - |B \cap C| - |A \cap C| + |A \cap B \cap C|$$

De maneira similar, podemos definir uma equação para a união de 4 conjuntos:

$$\begin{array}{rcl} |A \cup B \cup C \cup D| & = & |A| + |B| + |C| + |D| \\ & & -|A \cap B| - |A \cap C| - |A \cap D| - |B \cap C| - |B \cap D| - |C \cap D| \\ & & +|A \cap B \cap C| + |B \cap C \cap D| + |A \cap C \cap D| + |A \cap B \cap D| \\ & & -|A \cap B \cap C \cap D| \end{array}$$

■

A seguir apresentamos exemplos que utilizam o princípio da inclusão-exclusão na prática.

Exemplo 100. Em uma classe de matemática discreta, todo estudante é graduando em engenharia de computação ou em sistemas de informação ou em ambos. O número de estudantes matriculados em engenharia de computação é 25, em sistemas, 13 e em ambos os cursos estão matriculados 8 alunos. Quantos estudantes há nesta classe?

Solução: Considere como A o conjunto de estudantes de engenharia de computação e B o conjunto de alunos de sistemas de informação. Logo, pelo princípio de inclusão-exclusão temos:

$$\begin{array}{rcl} |A \cup B| & = & |A| + |B| - |A \cap B| \\ |A \cup B| & = & 25 + 13 - 8 \\ |A \cup B| & = & 30 \end{array}$$

■

Exemplo 101. Um conjunto de 1232 alunos cursam espanhol, 879 francês e 114 russo. Além disso, 103 cursam espanhol e francês, 23 cursam espanhol e russo e 14 cursam francês e russo. Se 2092 estudantes cursam pelo menos uma das três línguas, quantos estudantes cursam as três?

*Solução:*Chame de S o conjunto de estudantes que cursam espanhol, F os que estudam francês e R os que estudam russo. Evidentemente, o conjunto de estudantes que cursa as 3 línguas é $S \cap R \cap F$. Logo, pelo princípio de inclusão-exclusão, temos:

$$\begin{array}{lcl} |S \cup R \cup F| & = & |S| + |R| + |F| - |S \cap R| - |R \cap F| - |S \cap F| + |S \cap R \cap F| \\ 2092 & = & 1232 + 114 + 879 - 23 - 103 - 14 + |S \cap R \cap F| \\ |S \cap R \cap F| & = & 7 \end{array}$$

Assim, temos que o número total de estudantes que cursam as três línguas é de 7 alunos. ■

6.3.1 Exercícios

1. Quantos números entre 1 e 3.600 são divisíveis por 3, 5 ou 7?

2. Quantos números entre 1 e 42.000 não são divisíveis por 2, por 3 e nem por 7?

3. Quantos elementos há na união de cinco conjuntos se os conjuntos contêm 10.000 elementos cada, cada par de conjuntos tem 1.000 elementos, cada trio possui 100 elementos, cada quatro conjuntos possui 10 elementos em comum e todos os 5 conjuntos possui 1 elemento em comum?

6.4 Princípio da Casa dos Pombos

Nesta seção apresentaremos um princípio que, apesar de bastante simples, permite a solução elegante de diversos problemas de contagem. De maneira simples, este princípio especifica que se mais de k objetos forem colocados em k caixas, então pelo menos uma caixa possuirá mais de um objeto. A definição formal deste princípio é apresentada a seguir.

Definição 44 (O Princípio da Casa dos Pombos)**.** Se n objetos são colocados em k caixas, então há pelo menos uma caixa com $\lceil \frac{n}{k} \rceil$[3]. ■

A seguir apresentamos exemplos que aplicam o princípio da casa dos pombos para a solução de alguns problemas.

Exemplo 102. Considere as seguintes afirmativas:

1. Em um grupo de 367 pessoas haverá pelo menos 2 que fazem aniversário no mesmo dia do ano.

2. Em um grupo de 27 palavras haverá pelo menos 2 que iniciam com a mesma letra.

[3]A função teto de a, $\lceil a \rceil$, retorna o menor inteiro maior que a.

Solução: Note que como um ano pode possuir 366 dias, temos que em um conjunto com 367 pessoas, pelo menos duas fazem aniversário no mesmo dia. Neste caso, consideramos como "caixas" o número de dias do ano e como "objetos" cada uma das 367 pessoas, colocando cada pessoa na caixa correspondente ao dia de seu aniversário. Para a segunda sentença, temos que o número de caixas é exatamente o número de letras do alfabeto, 26. Logo, se colocarmos cada palavra na caixa correspondente à sua letra inicial, temos que pelo princípio da casa dos pombos, pelo menos duas palavras do conjunto de 27 iniciam com a mesma letra. ■

Exemplo 103. Qual o número mínimo de estudantes em uma classe para que tenhamos certeza de que pelo menos 6 receberão a mesma nota, sabendo que apenas 5 notas diferentes são possíveis?

Solução: De acordo com o princípio da casa dos pombos, o número mínimo de estudantes pode ser obtido da seguinte maneira: Primeiro, consideramos que o número de caixas é o número de notas, 5. Logo, o número de estudantes é o menor valor n que satisfaz a equação $\lceil \frac{n}{5} \rceil = 6$. Para determinar n, podemos usar o seguinte raciocínio: Se temos 5 caixas, 25 alunos são suficientes para que todas as caixas possuam 5 alunos. Se somarmos 1, temos que uma caixa com certeza terá 6 elementos. Logo, o número mínimo de alunos para garantir que pelo menos 6 tirem a mesma nota será de 26 alunos. ■

6.4.1 Exercícios

1. Em um baralho padrão existem 52 cartas distribuídas entre 4 naipes. Cada naipe possui 13 cartas distintas.

   (a) Quantas cartas devem ser removidas do baralho para garantir que pelo menos 3 cartas do mesmo naipe sejam retiradas?

   (b) Quantas cartas devem ser removidas para se garantir que 2 cartas de mesmo tipo sejam retiradas?

   (c) Mostre que em um grupo de 5 números inteiros não necessariamente consecutivos, há pelo menos dois com o mesmo resto quando divididos por 4.

2. Há 38 horários diferentes de aulas em uma universidade. Se há 677 classes diferentes, quantas salas de aula diferentes são necessárias para acomodar esta demanda de ocupação?

3. Qual o menor número de cabos necessários para conectar oito computadores a quatro impressoras de maneira que quatro computadores diferentes estejam conectados a quatro impressoras diferentes.

6.5 Permutações, Combinações e Arranjos

Muitos problemas combinatórios podem ser resolvidos encontrando-se o número de maneiras de se organizar um número específico de elementos distintos de um conjunto em que a ordem destes elementos pode ser ou não relevante.

6.5.1 Arranjos e Permutações

Iniciamos esta seção apresentando os conceitos de arranjo e permutação.

Definição 45 (Arranjo e Permutação)**.** Um arranjo de r objetos de um conjunto de n objetos distintos, $A(n,r)$, é uma coleção ordenada de objetos. Se $n = r$ denominamos o esta coleção de permutação e representamos este conjunto por $P(n)$. ■

Pela definição anterior, podemos concluir que utilizamos arranjos e permutações sempre que estamos interessados em determinar o número de maneiras de se organizar objetos de maneira em que a ordem destes é relevante. O teorema seguinte deduz uma fórmula para $A(n,r)$ utilizando o princípio multiplicativo.

Teorema 6. *Seja n um inteiro positivo qualquer e r um inteiro tal que $1 \leq r \leq n$, então há $A(n,r) = \Pi_{1\leq i\leq (r-1)}(n-i)$ permutações de r objetos obtidas a partir de um conjunto contendo n destes objetos.*

Demonstração. Suponha n e r arbitrários. Suponha que $1 \leq r \leq n$. Como desejamos formar coleções de r objetos em que a ordem interessa, devemos, inicialmente, escolher o primeiro elemento, possuindo n opções. Para o segundo elemento, teremos $n-1$ opções e para o terceiro $n-2$. De maneira similar, para o r-ésimo objeto teremos $n-(r-1) = n-r+1$ possibilidades de escolha. Logo, pelo princípio multiplicativo, temos que

$$A(n,r) = \Pi_{1\leq i\leq (r-1)}(n-i)$$

□

Um corolário útil deste teorema é apresentado a seguir:

Corolário 1. *Sejam n, r inteiros arbitrários tais que $1 \leq r \leq n$. Então, $A(n,r) = \frac{n!}{(n-r)!}$.*

Demonstração. Suponha n, r inteiros arbitrários e que $1 \leq r \leq n$. Logo, pelo teorema 6, temos que $A(n,r) = \Pi_{1\leq i\leq (r-1)}(n-i)$. Mas:

$$\begin{array}{ll} \dfrac{n!}{(n-r)!} & = \\ \dfrac{n(n-1)(n-2)...(n-r+1)(n-r)!}{(n-r)!} & = \\ n(n-1)(n-2)...(n-r+1) & = \\ \Pi_{1\leq i\leq (r-1)}(n-i) & = \\ A(n,r) & \end{array}$$

conforme requerido. □

Evidentemente, pelo corolário 1, temos que o número de permutações de n objetos é igual a:

$$\begin{array}{lll} P(n) & = & \\ A(n,n) & = & \{\text{pela def. de } P(n)\} \\ \dfrac{n!}{(n-n)!} & = & \{\text{pelo corolário 1.}\} \\ n! & & \end{array}$$

Exemplo 104. De quantas maneiras podemos selecionar o primeiro, o segundo e o terceiro colocados em um concurso com 100 inscritos?

Solução: Note que neste caso, estamos interessados em construir coleções de 3 pessoas a partir de um conjunto de 100 inscritos em que a ordem destas 3 pessoas é importante (uma será o primeiro, outra o segundo, etc). Logo, desejamos determinar o número de arranjos de 100 elementos escolhidos 3 a 3, que de acordo com o teorema 6 é:

$$A(100,3) = \frac{100!}{97!}$$

■

Exemplo 105. Quantas permutações das letras ABCDEFGH contêm a sequência ABC?

Solução: Visto que as letras ABC devem aparecer na mesma ordem, estas devem ser consideradas como um "único bloco" nas permutações. Logo, devemos considerar permutações das 5 letras (D, E ,F ,G e H) e o bloco ABC. Logo, o número total de permutações de 6 elementos é $P(6) = 6!$. ■

6.5.2 Combinações

Nesta seção abordaremos a noção de combinação, que consiste de uma coleção não ordenada de objetos. A seguir apresentamos uma definição precisa deste conceito.

Definição 46 (Combinação)**.** Uma combinação de r objetos obtidos a partir de um conjunto de n elementos distintos é uma coleção não ordenada de r objetos. Combinações de n objetos escolhidos r a r são usualmente representadas por $C(n,r)$ ou $\binom{n}{r}$. ■

O seguinte teorema demonstra a fórmula para determinar o número de combinações.

Teorema 7. *Suponha n e r inteiros arbitrários tais que $0 \leq r \leq n$. Então, o número de combinações de n elementos escolhidos r a r é $C(n,r) = \binom{n}{r} = \frac{n!}{r!(n-r)!}$.*

Demonstração. Note que o número de arranjos de n objetos escolhidos r a r pode ser obtido a partir da formação das combinações $C(n,r)$ e, então, pela ordenação de cada uma destas, o que pode ser feito de $P(r)$ maneiras. Logo, temos que:

$$A(n,r) = C(n,r).P(r)$$

que implica que

$$C(n,r) = \frac{A(n,r)}{P(r)} = \frac{\frac{n!}{(n-r)!}}{r!} = \frac{n!}{r!(n-r)!}$$

□

Exemplo 106. Quantas mãos de poker podem ser retiradas de um baralho de 5 cartas a partir de um baralho com 52 cartas?

Solução: Neste caso estamos interessados em conjuntos de 5 cartas, obtidos a partir de um conjunto contendo 52 cartas, em que a ordem das cartas não é importante. Logo, o total de mãos é dado por $C(52, 5)$. ■

Exemplo 107. Quantas maneiras há para selecionar cinco jogadores de um time de basquete com 10 membros para disputar uma partida em outra escola?

Solução: A resposta é dada por $C(10, 5)$. ■

Exemplo 108. Quantos são os anagramas formados por duas vogais e cinco consoantes escolhidas a partir de um conjunto de 18 consoantes e 5 vogais?

Solução: A escolha das vogais se dá por $C(5, 2)$ e das consoantes em $C(18, 5)$. Porém, como estas podem ocorrer em qualquer posição, devemos permutar as 7 posições da coleção gerada. Logo, temos que o número total de anagramas será de: $C(5, 2)C(18, 3)P(7)$. ■

6.5.3 Exercícios

1. Demonstre a seguinte equivalência: $C(n, r) = C(n, n - r)$.
2. Quantos anagramas da palavra UNIFORMES iniciam com consoante e terminam em vogal?
3. Considere a palavra NÚMERO:
   (a) Quantos são seus anagramas?
   (b) Quantos anagramas começam com uma vogal?
   (c) Quantos começam com uma vogal e terminam com uma consoante?

6.6 Arranjos e Combinações com Repetições

Em muitos problemas de combinatória devemos formar coleções em que a repetição de elementos é permitida. Nesta seção analisaremos estratégias para lidar com tais problemas combinatórios.

6.6.1 Arranjos com Repetições

Arranjos com repetições são caracterizados por coleções ordenadas em que repetições de elementos são permitidas. O número de arranjos de r elementos escolhidos a partir de um conjunto de n, em que repetições são permitidas pode ser obtido facilmente usando a regra do produto, conforme o teorema seguinte.

Teorema 8. *O número de arranjos de r elementos escolhidos a partir de um conjunto de n elementos distintos é n^r.*

Demonstração. Para cada uma das r posições do arranjo temos um total de n possibilidades de escolha. Logo, pelo princípio multiplicativo, temos um total de n^r possíveis arranjos. □

Exemplo 109. Qual o total de placas de carros que podem ser construídas utilizando 7 símbolos, sendo os 3 primeiros letras e os 4 últimos dígitos?

Solução: Neste caso, temos que o total de placas é dado por $26^3 \times 10^4$. ■

6.6.2 Permutações com Repetições

Para o caso de permutações em que elementos podem ser repetidos, devemos ter cuidado para evitar a contagem de uma mesma coleção mais de uma vez.

Por exemplo, se considerarmos quantas permutações possui a palavra ELE como sendo 3! estaremos contando coleções mais de uma vez, pois, não é possível diferenciar entre as diferentes ocorrências da letra E. O próximo exemplo ilustra a intuição a ser utilizada para demonstrar a fórmula para o cálculo do número de permutações com repetições.

Exemplo 110. Quantas permutações diferentes podem ser obtidas para a palavra SUCCESS?

Solução:Como a palavra em questão possui símbolos repetidos, não podemos simplesmente calcular o número de permutações desta. Para obtermos o número de sequências diferentes que podem ser construídas utilizando-se estas letras, primeiro, devemos colocar os S's em 3 das 7 posições disponíveis, o que resulta em $C(7,3)$, o que deixa apenas 4 posições livres. Os dois C's podem ser distribuídos nas 4 posições restantes de $C(4,2)$, deixando 2 posições disponíveis. Finalmente, podemos distribuir o U de $C(2,1)$ e a letra E de $C(1,1)$ maneiras. Consequentemente, temos que pelo princípio multiplicativo temos que o total de sequências é igual a $C(7,3)C(4,2)C(2,1)C(1,1)$. ■

O padrão de raciocínio utilizado no exemplo anterior pode ser abstraído, conforme apresentado no teorema a seguir.

Teorema 9. *O número de permutações diferentes de n objetos, em que há n_1 objetos de tipo 1, n_2 objetos de tipo 2, ... e n_k objetos de tipo k, é:*

$$\frac{n!}{n_1!n_2!...n_k!}$$

Demonstração. Para determinar o número de permutações, primeiro devemos colocar n_1 objetos de tipo 1 de $C(n,n_1)$ maneiras, deixando $n-n_1$ posições livres. Então os objetos de tipo 2 podem ser colocados de $C(n-n_1,n_2)$ maneiras, deixando $n-n_1-n_2$ posições disponíveis. Continuamos este processo até posicionarmos os objetos de tipo n_k. Após esta etapa, pelo princípio multiplicativo, temos que o total de permutações diferentes é:

$$\begin{array}{ll} C(n,n_1)C(n-n_1,n_2)...C(n-n_1-...-n_{k-1},n_k) & = \\ \frac{n!}{n_1!(n-n_1)!}\frac{(n-n_1)!}{n_2!(n-n_1-n_2)!}...\frac{(n-n_1-...-n_k)!}{n_k!0!} & = \\ \frac{n!}{n_1!n_2!...n_k!} & \end{array}$$

□

6.6.3 Combinações com Repetições

Para determinar o número de combinações possíveis com repetições, como a ordem de elementos é irrelevante em combinações, devemos ser capazes de "separar" grupos de elementos diferentes em uma mesma combinação. Assim como fizemos para arranjos, primeiramente resolveremos um exemplo e na sequência, vamos abstrair o padrão de raciocínio utilizado para determinar uma fórmula para este tipo de combinações.

Exemplo 111. Há quantas maneiras de escolher cinco cédulas em uma caixa que contém cédulas de $1, 2, 5, 10, 20, 50, 100$ reais? Considere que a ordem de escolha é irrelevante, que cada nota de um mesmo valor é indistinguível de outras de mesmo valor e que há pelo menos cinco cédulas de cada valor.

Solução: Como temos 7 tipos de cédulas e desejamos selecionar 5 cédulas com repetições permitidas, temos que, de alguma maneira, agrupar os elementos indistinguíveis que farão parte da combinação gerada. Para isso, utilizaremos a seguinte analogia: considere a existência de uma caixa com separações, onde cada compartimento desta caixa seja destinado a um tipo de cédula. Abaixo, ilustramos esta caixa, como uma tabela:

1	2	5	10	20	50	100

Note que a caixa apresentada possui 6 separações. Como cada compartimento é destinado a um tipo de cédula, iremos colocar as cédulas de um mesmo valor em compartimento correspondente a este tipo de cédula. Abaixo, apresentamos a configuração da caixa ao selecionarmos 2 cédulas de 2 reais e 3 de 100 reais.

0	2	0	0	0	0	3

Veja que anotamos em cada compartimento a quantidade de cédulas que este possui. Perceba que esta caixa pode ser representada de maneira abstrata como sequências de "*" (para representar cédulas) e "|" para representar divisões da caixa. A configuração anterior pode ser representada pela seguinte sequência de carateres: "|**|||||***". Note que usando este truque da caixa, determinamos uma configuração de escolha das cédulas determinando as sequências não ordenadas formadas por 5 objetos a partir de um conjunto de 11 objetos (5 asteriscos e 6 barras). Isso é exatamente $C(11, 5)$. ■

O teorema seguinte generaliza as discussões anteriores.

Teorema 10. *O número de combinações de r elementos, permitindo repetições, a partir de um conjunto de n elementos é de $C(n + r - 1, r)$.*

Demonstração. Cada uma das combinações de um conjunto com n elementos, quando a repetição é permitida, pode ser representada por uma lista de $n - 1$ separadores ("| ") e r asteriscos. Então, os $n - 1$ separadores são utilizados para representar os n elementos diferentes, em que a i-ésima posição é igual a um asterisco toda vez que o i-ésimo elemento do conjunto aparecer na combinação. Logo, o número de combinações é dado por $C(n - 1 + r, r)$, pois cada lista corresponde a uma escolha de r posições para colocar os r asteriscos a partir de $n - 1 + r$ posições que contém r asteriscos e $n - 1$ separadores. □

6.6.4 Exercícios

1. Quantas sequências de 6 letras existem (com repetições permitidas)?
2. Há quantas maneiras de se selecionar três elementos não ordenados a partir de um conjunto com cinco elementos, em que repetições são permitidas?
3. Quantas permutações podem ser feitas com a palavra MISSISSIPI?

4. Quantas permutações podem ser feitas com a palavra ABRACADABRA?

5. Há quantas soluções possíveis para a equação

$$x_1 + x_2 + x_3 + x_4 = 17$$

em que x_1, x_2, x_3 e x_4 são inteiros não negativos?

6.7 Notas Bibliográficas

Noções de combinatória estão presentes em quase todos os livros de matemática discreta, variando um pouco a terminologia utilizada. Neste capítulo utilizamos exemplos de [6].

7

Relações

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

Henry Ford, Pioneiro da indústria automobilística.

7.1 Motivação

Existem diversos tipos de relações em nosso cotidiano. Algumas destas descrevem como membros de uma família estão relacionados entre si: pais, filhos, irmãos, irmãs, sobrinhos, etc. Outras especificam, por exemplo, que certas cidades pertecem a um determinado país: por exemplo, Londres está na Inglaterra, e Paris na França. Ou podemos ter uma relação que descreve quais automóveis são montados por um certo fabricante. Relações são utilizadas na matemática para descrever como dois números se relacionam: por exemplo, dados dois números x e y temos que $x \geq y$, $x < y$, em que $\geq$ e $<$ são relações entre números.

Relações estão presentes em diversos ramos da computação, pois a terminologia da teoria de relações permite descrever conceitos de maneira precisa. Talvez, a aplicação mais famosa de relações em ciência da computação são os bancos de dados relacionais. Porém, relações formam a base teórica de muitas outras áreas como a semântica de linguagens de programação, demonstração de terminação de algoritmos, representação de informação armazenada em máquinas de busca, teoria de grafos, etc. Uma vez que relações são ubíquas e importantes, é útil definí-las como objetos matemáticos e descrever suas propriedades. O objetivo deste capítulo é apresentar a teoria de relações e a demonstração de alguns resultados importantes desta.

Nota 1. Neste capítulo assumimos que o leitor já possui a maturidade para compreender e demonstrar teoremas. Portanto, na maioria das demonstrações o rascunho será completamente omitido. Porém, recomenda-se que este seja "reconstruído" pelo leitor para um maior entendimento do conteúdo. ■

7.2 Pares Ordenados e Produto Cartesiano

Em capítulos anteriores, lidamos com conjuntos em que cada elemento é um "componente" deste. Porém, como você aprendeu em outros cursos, existem conjuntos formados por pares de números que representam pontos em um plano. Nesta seção, vamos introduzir formalmente o conceito de par ordenado e como podemos construir conjuntos de pares utilizando uma operação conhecida como produto cartesiano. As definições seguintes apresentam estes conceitos.

Definição 47 (Par ordenado)**.** Sejam A e B conjuntos quaisquer em que $a \in A$ e $b \in B$. Dizemos que (a, b) é um par ordenado em que o primeiro elemento é $a \in A$ e o segundo $b \in B$. ■

A operação sobre conjuntos que permite a criação de pares ordenados é o chamado produto cartesiano, que é definido a seguir.

Definição 48 (Produto Cartesiano)**.** Sejam A e B dois conjuntos quaisquer. O produto cartesiano de A por B, $A \times B$, é definido como:

$$A \times B = \{(a, b) \mid a \in A \wedge b \in B\}$$

■

Exemplo 112. Sejam $A = \{1, 2, 3\}$ e $B = \{4, 5, 6\}$. Temos:

$$A \times B = \{(1,4), (1,5), (1,6), (2,4), (2,5), (2,6), (3,4), (3,5), (3,6)\}$$

Evidentemente, temos que $(1, 4) \in A \times B$. Além disso, para o par ordenado $(1, 4)$ temos que 1 é o primeiro elemento (um elemento do conjunto A) e 4 é o segundo (um elemento de B). ■

Uma boa maneira de atestarmos a compreensão de um novo conceito matemático é demonstrando teoremas sobre este.

Teorema 11. *Sejam A, B e C conjuntos arbitrários. Então $A \times (B \cap C) = (A \times B) \cap (A \times C)$.*

Demonstração. Suponha que A, B e C são conjuntos arbitrários. Suponha p arbitrário.

($\rightarrow$) Suponha $p \in A \times (B \cap C)$. Pela definição de produto cartesiano, temos que $p = (a, b)$ em que $a \in A$ e $b \in B \cap C$. Já que $b \in B \cap C$, temos que $b \in B$ e $b \in C$. Como $a \in A$ e $b \in B$, temos que $(a, b) \in A \times B$. De maneira similar, já que $a \in A$ e $b \in C$, temos que $(a, b) \in A \times C$. Logo, $(a, b) \in (A \times B) \cap (B \times C)$. Portanto, se $p \in A \times (B \cap C)$ então $(A \times B) \cap (A \times C)$.

($\leftarrow$) Suponha $p \in (A \times B) \cap (A \times C)$. Assim, temos que $p \in A \times B$ e $p \in A \times C$. Pela definição de produto cartesiano, temos que $p = (a, b)$ em que $a \in A$, $b \in B$ e $b \in C$. Já que $b \in B$ e $b \in C$, temos que $b \in B \cap C$. Mas, como $a \in A$ e $b \in B \cap C$, temos que $(a, b) \in A \times (B \cap C)$. Logo, se $p \in (A \times B) \cap (A \times C)$ então $p \in A \times (B \cap C)$.

Como p é arbitrário, temos que $A \times (B \cap C) = (A \times B) \cap (A \times C)$. Portanto, para todos conjuntos A, B e C temos que $A \times (B \cap C) = (A \times B) \cap (A \times C)$. □

Comentário 1. Um ponto crucial desta demonstração é a utilização das hipóteses que um elemento p que pertence ao produto cartesiano de dois conjuntos deve ser um par em que o primeiro elemento pertence ao primeiro conjunto e o segundo elemento, ao segundo conjunto.

No caso do teorema anterior, em um momento temos que $p \in A \times (B \cap C)$, então $p = (a, b)$ em que $a \in A$ e $b \in B \cap C$.

Além deste detalhe, toda a demonstração consiste apenas de uso de técnicas de provas que já vimos no capítulos 4 e 5.

Evidentemente, como o produto cartesiano de conjuntos é apenas um conjunto de pares ordenados, todas as notações da teoria de conjuntos (apresentadas no capítulo 5) são aplicáveis. ■

Abaixo apresentamos outra demonstração similar.

Teorema 12. *Seja A um conjunto qualquer. Então, $A \times \emptyset = \emptyset$.*

Demonstração. Suponha p arbitrário.

($\rightarrow$) Suponha que $p \in A \times \emptyset$. Como $p \in A \times \emptyset$, existem $a \in A$ e $b \in \emptyset$ tais que $p = (a, b)$. Mas, não existe $b \in \emptyset$. Logo, o resultado desejado é provado por contradição.

($\leftarrow$) Suponha que $p \in \emptyset$. Como não existe $p \in \emptyset$, por contradição, o resultado é provado.

Como p é arbitrário, temos que $A \times \emptyset = \emptyset$ □

Comentário 2. A chave da demonstração anterior é o uso do fato de que não existe elemento $x \in \emptyset$, o que nos permite concluir a demonstração usando contradição. ■

7.2.1 Exercícios

1. Prove os seguintes teoremas:
   (a) Seja A um conjunto qualquer. Então $A \times \emptyset = \emptyset$.
   (b) Sejam A e B conjuntos quaisquer. Se $A \times B = B \times A$ se e somente se $A = \emptyset$ ou $B = \emptyset$ ou $A = B$.

7.3 Introdução às Relações

Matematicamente, especificamos que dois objetos a e b estão relacionados dizendo que o par (a, b) pertence ao conjunto de pares que descreve uma propriedade de interesse sobre estes objetos. Usamos relações (conceito matemático) para expressar relacionamentos entre objetos modelados matematicamente como elementos de conjuntos.

Definição 49 (Relação)**.** Suponha que A e B são conjuntos quaisquer. Denominamos o conjunto $R \subseteq A \times B$ uma relação de A em B. ■

A seguir apresentamos alguns exemplos de relações.

Exemplo 113. Considere os seguintes conjuntos $A = \{1, 2, 3\}$ e $B = \{4, 5, 6\}$. Temos que $R = \{(1, 5), (3, 4)\}$ é uma relação de A em B, já que $R \subseteq A \times B$.

Outro exemplo de relação, agora envolvendo um conjunto infinito de pares, é:

$$G = \{(x, y) \in \mathbb{R} \times \mathbb{R} \mid x < y\}$$

esta relação representa, utilizando pares, o conceito de "menor" sobre números reais. ■

Relações não necessariamente são formadas apenas por conjuntos numéricos. O próximo exemplo mostra relações sobre conjuntos não numéricos.

Exemplo 114. Considere os seguintes conjuntos que poderiam ser utilizados para modelar um sistema de informação em uma universidade:

- S : conjunto de todos os estudantes da universidade.
- C: conjunto de todos os cursos de graduação da universidade.
- D: conjunto de todas as disciplinas oferecidas em cursos da universidade.
- P: conjunto de todos os professores que lecionam na universidade.

Utilizando estes conjuntos, temos as seguintes relações:

- $R = \{(e, c) \in S \times C \mid \text{O estudante } e \text{ está matriculado no curso } c\}$.
- $R_1 = \{(p, d) \in P \times D \mid \text{O professor } p \text{ leciona a disciplina } d\}$.

Evidentemente, estas relações estariam representadas por mecanismos apropriados de bancos de dados relacionais em um sistema de informação de uma universidade. A primeira relação modela as informações sobre qual é o curso em que um aluno está matriculado e a segunda, qual disciplina um professor leciona. ■

A seguir, apresentamos alguns conceitos provavelmente já conhecidos pelo leitor, mas em um contexto de funções e não de relações[1].

Definição 50 (Domínio, Imagem, Inversa)**.** Suponha que R é uma relação de A em B. Então o domínio de R é o conjunto:

$$dom(R) = \{a \in A \mid \exists b.b \in B \wedge (a, b) \in R\}$$

A imagem[2] de R é definida pelo seguinte conjunto:

$$ran(R) = \{b \in B \mid \exists a.a \in A \wedge (a, b) \in R\}$$

Finalmente, a relação inversa de R, $R^{-1} \subseteq B \times A$, é:

$$R^{-1} = \{(b, a) \mid \exists a.\exists b.a \in A \wedge b \in B \wedge (a, b) \in R\}$$

■

[1] Veremos, posteriormente, que funções são apenas um tipo especial de relações.

[2] Normalmente, livros denotam o conjunto imagem usando a abreviação *ran* de *range* de imagem em inglês.

Exemplo 115. Considere:

- $A = \{1, 2, 3, 4\}$ e $B = \{6, 7, 8, 9, 0\}$.
- $R = \{(1, 6), (3, 0), (2, 9)\}$.

Temos:

- $dom(R) = \{1, 2, 3\}$
- $ran(R) = \{0, 6, 9\}$
- $R^{-1} = \{(6, 1), (0, 3), (9, 2)\}$

■

Encerraremos esta seção com um conceito importante: o de composição de relações. Este conceito permite a construção de uma nova relação a partir de duas relações existentes. A seguir definimos formalmente este conceito e apresentamos alguns exemplos na sequência.

Definição 51 (Composição de Relações)**.** Sejam $R \subseteq A \times B$ e $S \subseteq B \times C$ duas relações sobre conjuntos A, B, C e D. A relação composta de S e R, $S \circ R$, é uma relação de A em C definida como:

$$S \circ R = \{(a, c) \mid \exists b.b \in B \wedge (a, b) \in R \wedge (b, c) \in S\}$$

■

Exemplo 116. Considere os seguintes conjuntos que poderiam ser utilizados para modelar um sistema de informação em uma universidade:

- S : conjunto de todos os estudantes da universidade.
- C: conjunto de todos os cursos de graduação da universidade.
- D: conjunto de todas as disciplinas oferecidas em cursos da universidade.
- P: conjunto de todos os professores que lecionam na universidade.

e as seguintes relações

- $R_1 = \{(p, d) \in P \times D \mid p \text{ leciona a disc. } d\}$.
- $R_2 = \{(d, c) \in D \times C \mid d \text{ está no curso } c\}$.
- $R_3 = \{(e, d) \in S \times D \mid e \text{ está matr. em } d\}$.

Temos que a relação $R_3 \circ R_1 \subseteq S \times P$ é definida como:

$R_3 \circ R_1 = \{(e, p) \in S \times P \mid \text{o professor } p \text{ leciona alguma disciplina para o aluno } e\}$

De maneira similar, podemos definir uma relação que especifica que um certo aluno está matriculado em um curso, usando composição e as relações R_2 e R_3:

$R_2 \circ R_3 = \{(e, c) \mid \text{o aluno } e \text{ está matriculado em alguma disciplina do curso } c\}$

■

Finalizaremos esta seção com alguns teoremas envolvendo as definições apresentadas. Novamente vale ressaltar que cabe ao leitor a tarefa de reconstruir o rascunho para um melhor entendimento do conteúdo apresentado.

Teorema 13. *Suponha que* $R \subseteq A \times B$. *Então,* $(R^{-1})^{-1} = R$.

Demonstração. Suponha p arbitrário.

$(\rightarrow)$: Suponha que $p \in (R^{-1})^{-1}$. Se $R \subseteq A \times B$, então $R^{-1} \subseteq B \times A$. Já que $R^{-1} \subseteq B \times A$ então $(R^{-1})^{-1} \subseteq A \times B$.Como $p \in (R^{-1})^{-1}$, temos que existem $a \in A$ e $b \in B$ e $p = (a, b)$. Se $(a, b) \in (R^{-1})^{-1}$, então $(b, a) \in R^{-1}$ e, portanto, pela definição de relação inversa, temos que $(a, b) \in R$. Logo, se $(a, b) \in (R^{-1})^1$ então $(a, b) \in R$.

$(\leftarrow)$: Suponha que $p \in R$. Como $R \subseteq A \times B$, temos que existem $a \in A$ e $b \in B$ tais que $p = (a, b)$. Pela definição de inversa, temos que se $(a, b) \in R$ temos que $(b, a) \in R^{-1}$ e $(a, b) \in (R^{-1})^{-1}$. Logo, se $(a, b) \in R$ temos que $(a, b) \in (R^{-1})^{-1}$.

Como p é arbitrário, temos que $(R^{-1})^{-1} = R$. □

Comentário 3. A demonstração do teorema anterior utiliza a representação lógica da igualdade de dois conjuntos de pares ordenados (relações). Formalmente, definimos a igualdade de dois conjuntos A e B da seguinte maneira:

$$A = B \equiv \forall x.x \in A \leftrightarrow x \in B$$

Além disso, utilizamos a definição de inversa de uma relação, que consiste em "trocar" a ordem dos elementos de um par ordenado. Se par $(x, y) \in R$ então temos que $(y, x) \in R^{-1}$. O restante da demonstração consiste em uso das estratégias de prova para o quantificador universal e o conectivo bicondicional. ■

Teorema 14. *Suponha que* $R \subseteq A \times B$, $S \subseteq B \times C$ *e* $T \subseteq C \times D$. *Então,* $T \circ (S \circ R) = (T \circ S) \circ R$.

Demonstração. Suponha que $R \subseteq A \times B$, $S \subseteq B \times C$ e $T \subseteq C \times D$. Suponha p arbitrário.

$(\rightarrow)$: Suponha que $p \in T \circ (S \circ R)$. Como $R \subseteq A \times B$, $S \subseteq B \times C$ e $T \subseteq C \times D$, temos que $S \circ R \subseteq A \times C$ e $T \circ (S \circ R) \subseteq A \times D$. Assim, como $T \circ (S \circ R) \subseteq A \times D$, temos que existem $a \in A$ e $d \in D$ tais que $p = (a, d)$. Pela definição de composição, temos que para $(a, d) \in T \circ (S \circ R)$, deve existir $c \in C$ tal que $(a, c) \in S \circ R$ e $(c, d) \in T$. Mas, para $(a, c) \in S \circ R$ deve existir $b \in B$ tal que $(a, b) \in R$ e $(b, c) \in S$. Logo, pela definição de composição, temos que $(b, d) \in T \circ S$. Novamente, por composição, podemos concluir que $(a, d) \in (T \circ S) \circ R$. Logo, se $p \in T \circ (S \circ R)$ então $p \in (T \circ S) \circ R$.

$(\leftarrow)$: Suponha que $p \in (T \circ S) \circ R$. Como $R \subseteq A \times B$, $S \subseteq B \times C$ e $T \subseteq C \times D$, temos que $T \circ S \subseteq B \times D$ e $(T \circ S) \circ R \subseteq A \times D$. Assim, como $(T \circ S) \circ R \subseteq A \times D$, temos que existem $a \in A$ e $d \in D$ tais que $p = (a, d)$. Pela definição de composição, temos que para $(a, d) \in (T \circ S) \circ R$ deve existir $b \in B$ tal que $(b, d) \in T \circ S$ e $(a, b) \in R$. Por sua vez, para

$(b, d) \in T \circ S$, deve existir $c \in C$ tal que $(b, c) \in S$ e $(c, d) \in T$. Novamente, por composição, temos que $(a, c) \in S \circ R$ e que $(a, d) \in T \circ (S \circ R)$. Logo, se $p \in (T \circ S) \circ R$ então $p \in T \circ (S \circ R)$.

Como p é arbitrário, temos que $T \circ (S \circ R) = (T \circ S) \circ R$. □

Comentário 4. Neste teorema utilizou-se extensivamente a definição de composição de relações. Se $R \subseteq A \times B$ e $S \subseteq B \times C$, então $S \circ R \subseteq A \times C$ é definido como:

$$S \circ R = \{(a, c) \mid \exists b.b \in B \wedge (a, b) \in R \wedge (b, c) \in S\}$$

A partir desta definição, utilizamos a hipótese envolvendo o quantificador existencial para deduzir cada um dos pares que pertencem as relações R, S e T que foram utilizados para construir o par $p = (a, d)$ utilizado na demonstração. ■

7.3.1 Exercícios

1. Sejam $A = \{1, 2, 3\}$, $B = \{4, 5, 6\}$, $R = \{(1, 4), (1, 5), (2, 5), (3, 6)\}$ e $S = \{(4, 5), (4, 6), (5, 4), (6, 6)\}$. Note que $R \subseteq A \times B$ e $S \subseteq B \times B$. Encontre as seguintes relações:
   (a) $S \circ R$
   (b) $S \circ S$
   (c) $S^{-1} \circ R$
   (d) $R^{-1} \circ S$
2. Seja R uma relação sobre um conjunto A. Prove que $R \circ R^{-1} \subseteq i_A$, em que $i_A = \{(x, x) \mid x \in A\}$.
3. Sejam A e B dois conjuntos quaisquer.
   (a) Prove que para toda relação $R \subseteq A \times B$, $R \circ i_A = R$, em que $i_A = \{(x, x) \mid x \in A\}$.
   (b) Prove que para toda relação $R \subseteq A \times B$, $i_B \circ R = R$, em que $i_B = \{(x, x) \mid x \in B\}$.

7.4 Relações Binárias

Nesta seção apresentaremos propriedades de um tipo especial de relação: as relações binárias, cuja definição apresentamos a seguir.

Definição 52 (Relação Binária)**.** Seja A um conjunto qualquer. Dizemos que R é uma relação binária sobre A se $R \subseteq A \times A$. ■

Exemplo 117. As seguintes relações são relações binárias sobre os seguintes conjuntos $A = \{1, 2\}$, $\mathbb{N}$, P (conjunto de todas as pessoas) e subconjuntos de um conjunto B ($\mathcal{P}(B)$).

- $R = \{(1, 2), (1, 1)\}$.
- $G = \{(x, y) \in \mathbb{N} \times \mathbb{N} \mid x > y\}$.
- $I = \{(x, y) \in P \times P \mid x \text{ é irmão de } y\}$.

- $S = \{(x, y) \in \mathcal{P}(B) \times \mathcal{P}(B) \,|\, x \subseteq y\}$.

■

Relações binárias são interessantes por possuírem diversas propriedades que permitem que possamos classificá-las e usar diversos resultados sobre estas propriedades. Antes de apresentarmos estas propriedades, vamos introduzir uma notação para representar o fato que um certo par pertence a uma relação R.

Notação 1. Seja $R \subseteq A \times A$ uma relação binária qualquer sobre um conjunto A. Representaremos o fato de que $(x, y) \in R$ como xRy.

A seguir definimos estas propriedades.

Definição 53 (Relação Reflexiva)**.** Seja $R \subseteq A \times A$ uma relação binária qualquer. Dizemos que R é uma relação reflexiva se

$$\forall x.x \in A \rightarrow xRx.$$

■

Exemplo 118. Abaixo apresentamos diversos exemplos de relações reflexivas.

- $R = \{(x, y) \in \mathbb{N} \times \mathbb{N} \,|\, x \leq y\}$ é uma relação reflexiva pois todo número $n \in \mathbb{N}$ é menor ou igual a si próprio.
- $R_1 = \{(p, q) \,|\, \text{as palavras } p \text{ e } q \text{ iniciam com a mesma letra do alfabeto.}\}$ é uma relação reflexiva pois toda palavra p inicia com a mesma letra que ela própria.
- $R_2 = \{(x, y) \in \mathcal{P}(A) \times \mathcal{P}(A) \,|\, x \subseteq y\}$ é uma relação reflexiva pois todo conjunto x é subconjunto de si próprio.

■

Definição 54 (Relação Irreflexiva)**.** Seja $R \subseteq A \times A$ uma relação binária qualquer. Dizemos que R é uma relação irreflexiva se

$$\forall x.x \in A \rightarrow \neg(xRx).$$

■

Exemplo 119. São exemplos de relações irreflexivas.

- $R = \{(x, y) \in \mathbb{R} \times \mathbb{R} \,|\, x < y\}$, é uma relação irreflexiva pois, para qualquer número $x \in \mathbb{R}$, temos que não é verdade que $x < x$.
- $R_1 = \{(a, b) \,|\, \text{A pessoa } a \text{ é pai da pessoa } b\}$, é uma relação irreflexiva pois, não é possível uma pessoa ser pai dela própria.

■

Definição 55 (Relação Simétrica)**.** Seja $R \subseteq A \times A$ uma relação binária qualquer. Dizemos que R é uma relação simétrica se

$$\forall x.\forall y.x \in A \wedge y \in A \wedge xRy \rightarrow yRx.$$

■

Exemplo 120. São exemplos de relações simétricas.

- $R = \{(p, q) \mid$ a pessoa p é irmã(o) da pessoa $q.\}$ é uma relação simétrica já que, para quaisquer p e q, se p é irmão de q então q também é irmão de p.
- $R_1 = \{(p, q) \mid$ as palavras p e q iniciam com a mesma letra do alfabeto.$\}$ é uma relação simétrica já que, para quaisquer palavras p e q, se p inicia com a mesma letra que q, então q inicia com a mesma letra que p.
- $R_2 = \{(x, y) \in \mathcal{P}(A) \times \mathcal{P}(A) \mid \exists z.z \in x \wedge z \in y\}$ é uma relação simétrica, pois se um elemento z pertence a um conjunto x e a um conjunto y então este mesmo elemento pertence ao conjunto y e x.

■

Definição 56 (Relação Transitiva)**.** Seja $R \subseteq A \times A$ uma relação binária qualquer. Dizemos que R é uma relação transitiva se

$$\forall x.\forall y.\forall z.x \in A \wedge y \in A \wedge z \in A \wedge xRy \wedge yRz \rightarrow xRz$$

■

Exemplo 121. São exemplos de relações transitivas.

- $R = \{(x, y) \in \mathbb{N} \times \mathbb{N} \mid x \leq y\}$ é uma relação transitiva, pois se $x \leq y$ e $y \leq z$ então $x \leq z$.
- $R_1 = \{(p, q) \mid$ as palavras p e q iniciam com a mesma letra do alfabeto.$\}$ é uma relação transitiva já que, para quaisquer palavras p, q e r, se p inicia com a mesma letra que q e q inicia com a mesma letra que r então p inicia com a mesma letra que r.
- $R_2 = \{(x, y) \in \mathcal{P}(A) \times \mathcal{P}(A) \mid x \subseteq y\}$ é uma relação transitiva, pois se um conjunto x está contido em um conjunto y e y está contido em um conjunto z, então x está contido em z.

■

Definição 57 (Relação Anti-simétrica)**.** Seja $R \subseteq A \times A$ uma relação binária qualquer. Dizemos que R é uma relação anti-simétrica se

$$\forall x.\forall y.x \in A \wedge y \in A \wedge xRy \wedge yRx \rightarrow x = y$$

■

Exemplo 122. São exemplos de relações anti-simétricas.

- $R = \{(x, y) \in \mathbb{N} \times \mathbb{N} \mid x \leq y\}$ é uma relação anti-simétrica, pois se $x \leq y$ e $y \leq x$ então $x = y$.
- $R_2 = \{(x, y) \in \mathcal{P}(A) \times \mathcal{P}(A) \mid x \subseteq y\}$ é uma relação anti-simétrica, pois se um conjunto x está contido em um conjunto y e y está contido em x, então os conjuntos x e y são iguais (pela definição de igualdade de conjuntos).

■

Agora que estas definições foram apresentadas juntamente com alguns exemplos, as utilizaremos para demonstrar alguns teoremas. Vamos demonstrar um dos teoremas em detalhes (apresentando o rascunho e a construção passo-a-passo do texto) e os outros dois mostraremos apenas o texto final.

Teorema 15. *Suponha que R é uma relação binária sobre um conjunto A. Então, Se R é reflexiva então $i_A \subseteq R$, em que $i_A = \{(x, x) \mid x \in A\}$.*

Exemplo 123. Demonstraremos o primeiro item em detalhes. A partir do enunciado do primeiro item, temos a seguinte configuração inicial do rascunho.

Hipóteses	Conclusão
$R \subseteq A \times A$	R é reflexiva $\rightarrow i_A \subseteq R$

Utilizando a estratégia de prova direta, temos a seguinte configuração do rascunho.

Hipóteses	Conclusão
$R \subseteq A \times A$	$i_A \subseteq R$
R é reflexiva	

Utilizando a definição de subconjunto, temos

Hipóteses	Conclusão
$R \subseteq A \times A$	$\forall p.p \in i_A \rightarrow p \in R$
R é reflexiva	

Agora, aplicando as estratégias de prova para o quantificador universal e implicação (nesta ordem) temos

Hipóteses	Conclusão
$R \subseteq A \times A$	$p \in R$
R é reflexiva	
p arbitrário	
$p \in i_A$	

Se $p \in i_A$ então existe y tal que $p = (y, y)$.

Hipóteses	Conclusão
$R \subseteq A \times A$	$p \in R$
R é reflexiva	
p arbitrário	
$p \in i_A$	
$\exists y.y \in A \wedge p = (y, y)$	

Usando a estratégia de hipóteses para o quantificador existencial, temos

Hipóteses	Conclusão
$R \subseteq A \times A$	$p \in R$
R é reflexiva	
p arbitrário	
$p \in i_A$	
$\exists y.y \in A \wedge p = (y, y)$	
$y \in A$	
$p = (y, y)$	

Utilizando a definição de relação reflexiva, temos:

Hipóteses	Conclusão
$R \subseteq A \times A$	$p \in R$
R é reflexiva	
p arbitrário	
$p \in i_A$	
$\exists y.y \in A \wedge p = (y, y)$	
$y \in A$	
$p = (y, y)$	
$\forall x.x \in A \rightarrow xRx$	

Agora, basta usar a eliminação do quantificador universal (substituindo x por y), temos:

Hipóteses	Conclusão
$R \subseteq A \times A$	$p \in R$
R é reflexiva	
p arbitrário	
$p \in i_A$	
$\exists y.y \in A \wedge p = (y, y)$	
$y \in A$	
$p = (y, y)$	
$\forall x.x \in A \rightarrow xRx$	
$y \in A \rightarrow yRy$	

Usando as hipóteses $y \in A$ e $y \in A \rightarrow yRy$, concluímos a demonstração do teorema.

Agora, vamos construir o texto deste teorema passo-a-passo. Primeiramente, mostramos a parte do texto correspondente a primeira implicação deste teorema.

Suponha que R seja uma relação reflexiva.
    [Prova de $i_A \subseteq R$].
Portanto, se R é uma relação reflexiva então $i_A \subseteq R$.

Agora, utilizando a definição de $\subseteq$ em termos do quantificador universal, temos:

Suponha que R seja uma relação reflexiva.
    Suponha p arbitrário.
        [Prova de $p \in i_A \rightarrow p \in R$].
    Como p é arbitrário, temos que $i_A \subseteq R$.
Portanto, se R é uma relação reflexiva então $i_A \subseteq R$.

Usando prova direta, temos

Suponha que R seja uma relação reflexiva.
    Suponha p arbitrário.
        Suponha que $p \in i_A$.
            [Prova de $p \in R$].
        Logo, se $p \in i_A$ então $p \in R$.
    Como p é arbitrário, temos que $i_A \subseteq R$.
Portanto, se R é uma relação reflexiva então $i_A \subseteq R$.

Agora, concluímos o texto utilizando as hipóteses.

Suponha que R seja uma relação reflexiva.
Suponha p arbitrário.
Suponha que $p \in i_A$.
Como $p \in i_A$, existe $y \in A$ tal que $p = (y, y)$.
Como R é reflexiva e $y \in A$, temos que yRy.
Logo, se $p \in i_A$ então $p \in R$.
Como p é arbitrário, temos que $i_A \subseteq R$.
Portanto, se R é uma relação reflexiva então $i_A \subseteq R$.

■

Agora, mais dois teoremas sobre relações. Estes serão apresentados sem detalhes[3].

Teorema 16. *Seja R uma relação binária sobre um conjunto A. Então, Se R é simétrica então $R = R^{-1}$.*

Demonstração. Suponha que R seja uma relação simétrica. Suponha p arbitrário.

$(\rightarrow)$: Suponha que $p \in R$. Como $R \subseteq A \times A$, então existem $x, y \in A$ tais que $p = (x, y)$. Uma vez que xRy e R é simétrica, temos que yRx. Já que yRx, pela definição de inversa, temos que $xR^{-1}y$. Logo, se $p \in R$ então $p \in R^{-1}$.

$(\leftarrow)$: Suponha que $p \in R^{-1}$. Como $R \subseteq A \times A$, então existem $x, y \in A$ tais que $p = (x, y)$. Uma vez que $xR^{-1}y$, pela definição de inversa, temos que yRx. Como yRx e R é simétrica, temos que xRy. Logo, se $p \in R^{-1}$ então $p \in R$.

Como p é arbitrário, temos que $R = R^{-1}$. Portanto, se R é uma relação simétrica, temos que $R = R^{-1}$. □

Teorema 17. *Seja R uma relação binária sobre um conjunto A. Então, Se R é transitiva então $R \circ R \subseteq R$.*

Demonstração. Suponha que R é uma relação transitiva. Suponha p arbitrário. Suponha que $p \in R \circ R$. Como $R \subseteq A \times A$, temos que existem $a, c \in A$ tais que $p = (a, c)$. Como $(a, c) \in R \circ R$, pela definição de composição de relações, temos que existe $b \in A$ tal que aRb e bRc. Como R é transitiva, aRb e bRc, temos que aRc. Logo, se $p \in R \circ R$ então $p \in R$. Como p é arbitrário, temos que $R \circ R \subseteq R$. Portanto, se R é uma relação transitiva então $R \circ R \subseteq R$. □

7.4.1 Exercícios

1. Seja $A = \{\text{banana, abacate, melancia, ovo, ócio}\}$ e $R = \{(x, y) \in A \times A \mid \text{a palavra } x \text{ tem alguma letra em comum com a palavra } y\}$.
   (a) Liste os pares que formam a relação R.
   (b) Quais propriedades (reflexiva, irreflexiva, simétrica, transitiva, anti-simétrica) possui a relação R?

[3] *Dica do professor amigo:* ***Entenda*** *todas essas demonstrações!*

2. Demonstre os seguintes teoremas.

   (a) Suponha que R é uma relação binária sobre um conjunto A. Se R é reflexiva então $R \subseteq R \circ R$.

   (b) Suponha que R é uma relação binária sobre um conjunto A. Se R é reflexiva então R^{-1} também é reflexiva.

   (c) Sejam R_1 e R_2 duas relações binárias sobre um conjunto A. Então se R_1 e R_2 são relações simétricas, então $R_1 \cup R_2$ e $R_1 \cap R_2$ também são simétricas.

7.5 Relações de Ordem

7.5.1 Introdução

Usando as definições de propriedades de relações, apresentadas na seção anterior, podemos notar que diversas relações possuem características similares. Note as relações seguintes:

1. $R = \{(x, y) \in \mathbb{R} \times \mathbb{R} \mid x \leq y\}$
2. $S = \{(x, y) \in \mathcal{P}(A) \times \mathcal{P}(A) \mid x \subseteq y\}$

são ambas relações reflexivas, transitivas e anti-simétricas. Relações com estas propriedades são denominadas ordens parciais e possuem características que permitem entendê-las como um critério de ordenação entre elementos de um conjunto. A próxima definição apresenta estes conceitos.

Definição 58 (Pré-Ordem e Ordens Parciais)**.** Seja $R \subseteq A \times A$ uma relação. Dizemos que R é uma pré-ordem se R for reflexiva e transitiva. Uma ordem parcial é uma relação reflexiva, transitiva (pré-ordem) e anti-simétrica. ■

Exemplo 124. São exemplos de ordens parciais (e, evidentemente, de pré-ordens) as seguintes relações.

1. $R = \{(x, y) \in \mathbb{R} \times \mathbb{R} \mid x \leq y\}$.
2. $S = \{(x, y) \in \mathcal{P}(A) \times \mathcal{P}(A) \mid x \subseteq y\}$.
3. $T = \{(x, y) \in \{1, 2\} \times \{1, 2\} \mid |x| \leq |y|\}$.

■

Relações de ordem especificam uma maneira de compararmos elementos de um conjunto. A próxima definição torna este conceito preciso.

Definição 59 (Elementos Comparáveis)**.** Seja $R \subseteq A \times A$ uma relação de ordem qualquer (pré-ordem, ordem parcial, ordem total, lexicográfica) e $a, b \in A$. Dizemos que a e b são comparáveis em R se aRb ou bRa. ■

O nome ordem "parcial" deve-se ao fato de que nem sempre todos elementos de um conjunto A são comparáveis de acordo com uma ordem parcial R. Como exemplo, o par $(\{1, 2\}, \emptyset) \notin S$, pois $\{1, 2\} \not\subseteq \emptyset$. Por sua vez, para quaisquer números reais x, y temos que $x \leq y$ e $x \not\leq y$. Quando uma ordem parcial permite a comparação entre quaisquer elementos de um conjunto, dizemos que esta é uma ordem total.

Definição 60 (Ordem Total)**.** Seja $R \subseteq A \times A$ uma relação. Dizemos que R é uma ordem total se R for uma ordem parcial e adicionalmente a seguinte condição é verdadeira:

$$\forall x.\forall y.x \in A \wedge y \in A \wedge (xRy \vee yRx)$$

■

Evidentemente, relações como $\leq$ e $\geq$ definidas sobre conjuntos numéricos são ordens totais pois, sempre é possível comparar dois números para determinarmos qual destes é o maior (ou menor).

Exemplo 125. Para ilustrar estes conceitos, vamos considerar as seguintes relações definidas sobre $A = \{1, 2\}$:

- $S = \{(x, y) \in \mathcal{P}(A) \times \mathcal{P}(A) \,|\, x \subseteq y\}$.
- $T = \{(x, y) \in A \times A \,|\, |x| \leq |y|\}$.

É fácil constatar que ambas estas relações são ordens parciais. Porém, note que apenas T é total, pois todo par de conjuntos pode ser comparado com respeito ao tamanho destes, mas o mesmo não acontece com a noção de subconjunto, pois temos que o seguinte par não pertence a relação S: $(\{1, 2\}, \{1\})$. Logo, temos que S não é uma ordem total, pois não atende a condição

$$\forall x.\forall y.x \in A \wedge y \in A \wedge (xSy \vee ySx)$$

■

Outros tipos importantes de relações são as chamadas ordens estritas e ordens lexicográficas. A primeira é um tipo de relação de ordem em que possui características similares a relação de $<$ e $>$ para números e ordens lexicográficas constituem ordens parciais para n-uplas de valores. Apresentaremos a definição de ordem lexicográfica apenas para pares. Deixamos como exercício para o leitor sua extensão para n-uplas quaisquer.

Definição 61 (Ordem Estrita)**.** Seja $R \subseteq A \times A$ uma relação. Dizemos que R é uma ordem estrita se R é irreflexiva, transitiva e anti-simétrica. ■

Note que toda ordem parcial R pode ser "transformada" em uma ordem estrita eliminando os pares $i_A = \{(x, x) \,|\, x \in A\}$, isto é, se R é uma ordem parcial, então $R - i_A$ é uma ordem estrita.

Definição 62 (Ordem Lexicográfica)**.** Seja $\sqsubset \subseteq A \times A$ uma relação de ordem parcial sobre A. Definimos a ordem lexicográfica induzida por $\sqsubset$, $R \subseteq (A \times A) \times (A \times A)$, entre pares de elementos de A, como:

$$R = \{((x, y), (x', y') \,|\, x \sqsubset x' \wedge [x = x' \vee y \sqsubset y']\}$$

■

7.5.2 Exercícios

1. Considere as relações apresentadas no exemplo 125.
   (a) Liste os pares que formam as relações S e T.
   (b) Denomine por X um conjunto de pares que se presentes na relação T a tornariam uma ordem total. Determine o menor conjunto X tal que $X \cup T$ é uma ordem total.
2. Seja R uma ordem parcial sobre um conjunto A qualquer. Prove que R^{-1} também é uma ordem parcial.
3. Seja R uma ordem parcial sobre um conjunto A qualquer. Prove que $R - i_A$ é uma ordem estrita.
4. A definição 62 apresentou como definir uma ordem lexicográfica para pares de valores de um certo conjunto A. Apresente uma definição similar de uma ordem para triplas de valores de um conjunto A.

7.5.3 Elementos Máximos e Mínimos

Considere o seguinte conjunto $A = \{\text{me, tame, men, mental, mentalist}\}$ e a seguinte relação sobre este:

$$R = \{(x, y) \in A \times A \mid x \text{ é uma subpalavra de } y\}$$

É fácil mostrar que a relação R é uma ordem parcial sobre este conjunto A (prove isto!). Conforme já mencionado em diversos momentos, relações de ordem especificam critérios de comparação (ordem) entre os elementos do conjunto sobre o qual a relação está definida.

Desta forma, temos que, como a palavra me é subpalavra de tame, men, mental e mentalist. Se considerarmos que um par xRy denota que x é "menor" que y de acordo com a ordem parcial R, temos que o elemento me é o menor de todos os elementos do conjunto A. Elementos com esta propriedade são ditos elementos mínimos de um conjunto. A próxima definição formaliza este conceito.

Definição 63 (Elemento Mínimo e Máximo)**.** Seja R uma relação de ordem parcial sobre um conjunto A, $B \subseteq A$ e $b \in B$. Dizemos que b é um elemento mínimo de B, com respeito a relação R, se

$$\forall x.x \in B \rightarrow bRx$$

De maneira similar, dizemos que b é um elemento máximo de B se

$$\forall x.x \in B \rightarrow xRb$$

■

Definição 64 (Elementos Minimal e Maximal)**.** Seja R uma relação de ordem parcial sobre um conjunto A, $B \subseteq A$ e $b \in B$. Dizemos que b é um elemento minimal de B, com respeito a relação R, se

$$\neg\exists x.x \in B \wedge xRb \wedge x \neq b$$

De maneira similar, dizemos que b é um elemento maximal de B se

$$\neg\exists x.x \in B \wedge bRx \wedge x \neq b$$

■

A seguir apresentamos alguns exemplos que ilustram estas definições.

Exemplo 126. Considere os seguintes problemas.

- Seja $L = \{(x, y) \in \mathbb{R}\times\mathbb{R} \,|\, x \leq y\}$, que evidentemente é uma ordem parcial. Seja $B = \{x \in \mathbb{R} \,|\, x \geq 7\}$. O conjunto B possui elementos mínimos / minimais? E o conjunto $C = \{x \in \mathbb{R} \,|\, x > 7\}$?
- Seja $S = \{(X, Y) \in \mathcal{P}(\mathbb{N}) \times \mathcal{P}(\mathbb{N}) \,|\, X \subseteq Y\}$, que evidentemente é uma ordem parcial. Seja $\mathcal{F} = \{X \in \mathcal{P}(\mathbb{N}) \,|\, 2 \in X \wedge 3 \in X\}$. O conjunto $\mathcal{F}$ possui elementos mínimos e minimais?

Solução: É evidente que $7 \leq x$, para todo $x \in B$. Logo, 7 é um elemento mínimo de B. Como não existe $x \in B$, $x \neq 7$ e $xR7$, temos que 7 também é um elemento minimal de B.

Note que para o segundo item, temos que o conjunto $\{2, 3\}$ é subconjunto de todo $X \in \mathcal{F}$. Como $\{2, 3\} \in \mathcal{F}$, temos que este é o elemento mínimo e minimal deste conjunto (porquê?). ■

Vale a pena ressaltar que todo elemento mínimo (máximo) é minimal (maximal), mas a recíproca não é verdadeira, além disso, se um conjunto possui um elemento mínimo (máximo), este é o único minimal (maximal) do conjunto em questão. Estes fatos serão demonstrados formalmente pelos teoremas a seguir [4].

Teorema 18. *Suponha R uma ordem parcial sobre A e $B \subseteq A$. Se B possui um elemento mínimo, então este é único.*

Hipóteses	Conclusão
	B possui mínimo $\rightarrow$ este mínimo é único
B possui mínimo	este mínimo é único

Evidentemente, a sentença "B possui mínimo", pode ser representada por um quantificador existencial:

Hipóteses	Conclusão
	B possui mínimo $\rightarrow$ este mínimo é único
B possui mínimo	este mínimo é único
$\exists b.b \in B \wedge b$ é o mínimo de B	este mínimo é único

Como temos uma hipótese envolvendo um quantificador existencial, podemos introduzir uma nova variável para representar o elemento mínimo do conjunto B:

[4] **Entenda** esses teoremas e suas demonstrações. — Dica do seu professor camarada.

Hipóteses	Conclusão
	B possui mínimo $\rightarrow$ este mínimo é único
B possui mínimo	este mínimo é único
$\exists b.b \in B \wedge b$ é o mínimo de B	
$b \in B$	b é o único mínimo de B
b é mínimo de B	

A sentença "b é o único mínimo de B" pode ser representada como "todo mínimo de B é igual a b". Com isso temos:

Hipóteses	Conclusão
	B possui mínimo $\rightarrow$ este mínimo é único
B possui mínimo	este mínimo é único
$\exists b.b \in B \wedge b$ é o mínimo de B	
$b \in B$	b é o único mínimo de B
b é mínimo de B	
	$\forall c.c$ é mínimo de $B \rightarrow b = c$

Utilizando as técnicas de prova para quantificadores universais e implicações:

Hipóteses	Conclusão
	B possui mínimo $\rightarrow$ este mínimo é único
B possui mínimo	este mínimo é único
$\exists b.b \in B \wedge b$ é o mínimo de B	
$b \in B$	b é o único mínimo de B
b é mínimo de B	
	$\forall c.c$ é mínimo de $B \rightarrow b = c$
c arbitrário	$b = c$
c é mínimo de B	

Expandindo as definições de mínimo:

Hipóteses	Conclusão
	B possui mínimo $\rightarrow$ este mínimo é único
B possui mínimo	este mínimo é único
$\exists b.b \in B \wedge b$ é o mínimo de B	
$b \in B$	b é o único mínimo de B
$\forall x.x \in B \rightarrow bRx$	
	$\forall c.c$ é mínimo de $B \rightarrow b = c$
c arbitrário	$b = c$
$\forall y.y \in B \rightarrow cRy$	

Como $c \in B$ e $\forall x.x \in B \rightarrow bRx$, podemos concluir que bRc. De maneira similar, como $b \in B$ e $\forall y.y \in B \rightarrow cRy$, temos que cRb. Uma vez que R é uma ordem parcial, temos que R é anti-simétrica, sendo assim, como bRc e cRb, temos que $b = c$, como requerido. O texto desta demonstração é apresentado a seguir.

Demonstração. Suponha que b é o elemento mínimo de B e que $c \in B$ arbitrário também é mínimo de B. Como b é mínimo e $c \in B$, temos que bRc. Da mesma forma, como c é mínimo de B e $b \in B$, temos que cRb. Como R é uma ordem parcial, temos que $b = c$. Já que c é arbitrário, podemos concluir que $b \in B$ é o único elemento mínimo de B. Portanto, se B possui um elemento mínimo, este é único. □

Teorema 19. *Suponha R uma ordem parcial sobre A, $B \subseteq A$ e que $b \in B$ é o mínimo de B. Então b é minimal e é o único minimal de B.*

Hipóteses	Conclusão
b é mínimo de B	b é minimal de $B \wedge$ b é o único minimal de B

Primeiro, vamos provar que b é minimal de B. Usando as definições de mínimo e minimal:

Hipóteses	Conclusão
$\forall x.x \in B \rightarrow bRx$	$\neg\exists y.y \in B \wedge yRb \wedge y \neq b.$

Utilizando álgebra booleana, temos que $\neg\exists y.y \in B \wedge yRb \wedge y \neq b = \forall y.yRb \rightarrow y = b$.

Hipóteses	Conclusão
$\forall x.x \in B \rightarrow bRx$	$\forall y.yRb \rightarrow y = b.$

Utilizando as técnicas de prova para quantificadores universais e implicações:

Hipóteses	Conclusão
$\forall x.x \in B \rightarrow bRx$	$\forall y.yRb \rightarrow y = b$
y arbitário	$y = b$
yRb	

Mas, como $\forall x.x \in B \rightarrow bRx$ e $y \in B$ temos que bRy. Uma vez que R é anti-simétrica, yRb e bRy, temos que $b = y$, conforme requerido. Com isso, provamos a primeira parte, que $b \in B$ é minimal. Agora resta provar que b é o único minimal de B.

Hipóteses	Conclusão
$\forall x.x \in B \to bRx$	$\forall c.c$ é minimal de $B \to b = c$

Utilizando as técnicas de prova para quantificadores universais, implicações e a definição de minimal, temos:

Hipóteses	Conclusão
$\forall x.x \in B \to bRx$	$b = c$
c arbitário	
$\neg\exists y.yRc \wedge y \neq c$	

Por álgebra booleana, temos que:

Hipóteses	Conclusão
$\forall x.x \in B \to bRx$	$b = c$
c arbitário	
$\forall y.yRc \to y = c$	

Como $c \in B$ e $\forall x.x \in B \to bRx$, temos que bRc. Finalmente, como bRc e $\forall y.yRc \to y = c$, podemos concluir que $b = c$, conforme requerido. O texto desta demonstração é apresentado a seguir.

Demonstração. Suponha que $b \in B$ é mínimo e $y \in B$ arbitrário tal que yRb. Como b é mínimo, temos que bRy. Uma vez que yRb, bRy e R é anti-simétrica, temos que $y = b$. Portanto, b é um elemento minimal de B. Suponha c arbitrário tal que c é minimal de B. Como b é mínimo, temos que bRc. Uma vez que c é minimal, temos que cRb e já que R é uma ordem parcial, temos que $c = b$. Como c é arbitário, temos que b é o único minimal de B. □

Teorema 20. *Seja R uma ordem total sobre A, $B \subseteq A$, $b \in B$. Se b é um elemento minimal de B, então b é o mínimo de B.*

Para este teorema, temos a seguinte configuração inicial do rascunho.

Hipóteses	Conclusão
R é uma ordem total	b é o mínimo de B
$B \subseteq A$	
b é minimal de B	

Expandindo as definições de mínimo e minimal, temos:

Hipóteses	Conclusão
R é uma ordem total	$\forall x.x \in B \to bRx$
$B \subseteq A$	
$\neg\exists y.y \in B \wedge yRb \wedge y \neq b$	

Utilizando as estratégias de prova para o quantificador universal e implicação (nesta ordem) temos:

Hipóteses	Conclusão
R é uma ordem total	bRx
$B \subseteq A$	
$\neg\exists y.y \in B \wedge yRb \wedge y \neq b$	
x arbitrário	
$x \in B$	

É óbvio que $x = b \vee x \neq b$. Usando este fato, temos:

Hipóteses	Conclusão
R é uma ordem total	bRx
$B \subseteq A$	
$\neg\exists y.y \in B \wedge yRb \wedge y \neq b$	
x arbitrário	
$x \in B$	
$x = b \vee x \neq b$	

Como R é reflexiva, para $x = b$, o resultado é imediato. Logo, vamos considerar que $x \neq b$.

Hipóteses	Conclusão
R é uma ordem total	bRx
$B \subseteq A$	
$\neg\exists y.y \in B \wedge yRb \wedge y \neq b$	
x arbitrário	
$x \in B$	
$x \neq b$	

Como R é uma ordem total temos que $xRb \vee bRx$.

Hipóteses	Conclusão
R é uma ordem total	bRx
$B \subseteq A$	
$\neg\exists y.y \in B \wedge yRb \wedge y \neq b$	
x arbitrário	
$x \in B$	
$x \neq b$	
$xRb \vee bRx$	

Agora, dividindo esta prova em casos, temos:

Hipóteses	Conclusão
R é uma ordem total	bRx
$B \subseteq A$	
$\neg\exists y.y \in B \wedge yRb \wedge y \neq b$	
x arbitrário	
$x \in B$	
$x \neq b$	
$xRb \vee bRx$	
Caso xRb:	bRx
Caso bRx:	bRx

O segundo caso é trivial. Para o primeiro, como xRb e $x \neq b$, temos que $\exists y.y \in B \wedge yRb \wedge y \neq b$, o que contradiz a hipótese $\neg\exists y.y \in B \wedge yRb \wedge y \neq b$, concluindo a demonstração deste teorema.

A seguir apresentamos o texto deste teorema.

Demonstração. Suponha que R é uma ordem total sobre A, $B \subseteq A$, $b \in B$. Suponha que b é um minimal de B. Suponha x arbitrário. Suponha que $x \in B$. Se $x = b$, como R é reflexiva, temos que bRx. Suponha que $x \neq b$. Uma vez que R é uma ordem total, temos que bRx ou xRb. Considere os casos:

- Caso bRx: imediato.
- Caso xRb: Como xRb e $x \neq B$, existe um valor $y \in B$ tal que yRb e $y \neq B$, o que contradiz a suposição de que b é minimal de B. Logo, bRx.

Como x é arbitrário, podemos concluir que b é o mínimo de B. □

7.5.4 Limites Inferiores e Superiores

Nesta seção estenderemos os conceitos de elementos mínimos e máximos para o que chamamos de limites inferiores e superiores, conceitos amplamente utilizados em diversos ramos da computação.

Definição 65 (Limite Inferior e Limite Superior)**.** Seja R uma ordem parcial sobre A, $B \subseteq A$ e $a \in A$. Dizemos que a é um limite inferior de B se

$$\forall x.x \in B \rightarrow aRx$$

De maneira similar, dizemos que a é um limite superior de B se

$$\forall x.x \in B \rightarrow xRa$$

■

Note que a única diferença entre limites inferiores (superiores) e elementos mínimos (máximos) de um conjunto é que os primeiros não necessariamente devem ser elementos do conjunto em questão. Além disso, Limites inferiores (superiores) para um conjunto não são únicos, conforme mostraremos no exemplo seguinte.

Exemplo 127. Considere a seguinte relação $R = \{(x, y) \in \mathbb{R} \times \mathbb{R} \,|\, x \leq y\}$ e os conjuntos $B = \{x \in \mathbb{R} \,|\, x \geq 7\}$ e $C = \{x \in \mathbb{R} \,|\, x \leq 7\}$. Temos que os elementos do intervalo $(-\infty, 7]$ são todos limites inferiores do conjunto B e o intervalo $[7, +\infty)$ é o conjunto de limites superiores de C. ■

Note que o número de limites inferiores (superiores) para um certo conjunto pode ser infinito. Logo, faz sentido em falarmos do conjunto de limites inferiores (superiores) de um conjunto qualquer. No exemplo anterior, temos que os intervalores $(-\infty, 7]$ e $[7, +\infty)$ são os conjuntos de limites inferiores de B e superiores de C, respectivamente. Em ambos os conjuntos, $(-\infty, 7]$ e $[7, +\infty)$, temos que 7 é o elemento máximo do conjunto de limites inferiores de B e, além disso, 7 também é o elemento mínimo do conjunto de limites superiores de C. A definição seguinte apresenta uma caracterização formal destes elementos máximos (mínimos) de conjuntos de limites inferiores (superiores).

Definição 66 (Maior Limite Inferior e Menor Limite Superior)**.** Seja R uma ordem parcial sobre A, $B \subseteq A$. Seja I o conjunto de limites inferiores de B e S o conjunto de limites superiores de B. Se I possui um elemento máximo, dizemos que este é o maior limite inferior de B. Caso S possua um elemento mínimo, dizemos que este é o menor limite superior de B. ■

Exemplo 128. Considere a seguinte relação $R = \{(x, y) \in \mathbb{R} \times \mathbb{R} \,|\, x \leq y\}$ e os conjuntos $B = \{x \in \mathbb{R} \,|\, x \geq 7\}$ e $C = \{x \in \mathbb{R} \,|\, x \leq 7\}$. Temos que os elementos do intervalo $(-\infty, 7]$ são todos limites inferiores do conjunto B e o intervalo $[7, +\infty)$ é o conjunto de limites superiores de C. Logo, o maior limite inferior de B é 7 e o menor limite superior de C é também 7. ■

7.5.5 Exercícios

1. Seja R uma ordem parcial sobre A, $B \subseteq A$ e $b \in B$.
   (a) Prove que se b é o elemento mínimo de B, então b é o maior limite inferior de B.
   (b) Prove que se b é o elemento máximo de B, então b é o menor limite superior de B.
2. Suponha que R é uma ordem parcial sobre um conjunto A e $B \subseteq A$. Prove que $R \cap (B \times B)$ é uma ordem parcial.
3. Suponha que R é uma ordem parcial sobre um conjunto parcial. Prove que R^{-1} é uma ordem parcial sobre A.
4. Suponha que A é um conjunto, $\mathcal{F} \subseteq \mathcal{P}(A)$, $\mathcal{F} \neq \emptyset$ e
$$R = \{(x, y) \times \mathcal{P}(A) \times \mathcal{P}(A) \,|\, x \subseteq y\}$$
   (a) Prove que $\bigcup \mathcal{F}$ é o menor limite superior de $\mathcal{F}$ para a ordem parcial R.
   (b) Prove que $\bigcap \mathcal{F}$ é o maior limite inferior de $\mathcal{F}$ para a ordem parcial R.

7.6 Relações de Equivalência

7.6.1 Introdução

Se relações de ordem podem ser consideradas como generalizações dos conceitos de "maior" e "menor" para números, relações de equivalência são consideradas generalizações do conceito de igualdade em matemática. A definição de relação de equivalência é apresentada a seguir.

Definição 67 (Relação de Equivalência)**.** Seja $R \subseteq A \times A$ uma relação. Dizemos que R é uma relação de equivalência se R for reflexiva, transitiva e simétrica. ■

A seguir apresentamos alguns exemplos de relações de equivalência [5].

[5] Recomendo **fortemente** que você tente provar o porquê cada uma destas relações é uma relação de equivalência.

Exemplo 129. São relações de equivalências:

- Seja P o conjunto de todas as pessoas do planeta. As seguintes relações sobre P são relações de equivalência.
  - $R = \{(x, y) \in P \times P \,|\, \text{as pessoas } x \text{ e } y \text{ possuem a mesma idade}\}$.
  - $R_1 = \{(x, y) \in P \times P \,|\, \text{as pessoas } x \text{ e } y \text{ possuem a mesma profissão}\}$.
  - $R_2 = \{(x, y) \in P \times P \,|\, \text{as pessoas } x \text{ e } y \text{ possuem o mesmo modelo de carro}\}$.
- Seja A um conjunto qualquer. A seguinte relação sobre $\mathcal{P}(A)$ é uma relação de equivalência:
  - $R = \{(x, y) \in \mathcal{P}(A) \times \mathcal{P}(A) \,|\, |x| = |y|\}$.

■

De maneira intuitiva, se R é uma relação de equivalência sobre um conjunto A e xRy, podemos dizer que x e y são iguais de acordo com o critério definido por R. Por exemplo, considere a relação

$$R_1 = \{(x, y) \in P \times P \,|\, \text{as pessoas } x \text{ e } y \text{ possuem a mesma profissão}\}$$

definida sobre o conjunto de pessoas, P. Se Asdrúbal e Criosvaldo são ambos advogados, então (Asdrúbal, Criosvaldo) $\in R_1$, pois ambos possuem a mesma profissão e, portanto, Asdrúbal e Criosvaldo podem ser considerados "iguais" de acordo com a relação R_1. Desta forma, podemos considerar que esta relação R_1 divide o conjunto de todas as pessoas em uma família de conjuntos em que cada um destes conjuntos possui todas as pessoas que exercem uma certa profissão. Estes conceitos são formalizados pelas definições seguintes.

Definição 68 (Classes de Equivalência)**.** Seja R uma relação de equivalência sobre um conjunto A e $x \in A$. A classe de equivalência de x com respeito a R, $[x]$, é definida como:

$$[x] = \{y \in A \,|\, yRx\}$$

O conjunto de todas as classes de equivalência de elementos de um conjunto A, A/R, é definida como:

$$A/R = \{[x] \,|\, x \in A\}$$

■

Como exemplo destes conceitos, considere:

Exemplo 130. Seja $A = \{1, 2\}$ e $R = \{(x, y) \in \mathcal{P}(A) \times \mathcal{P}(A) \,|\, |x| = |y|\}$. Temos que a relação R é composta pelos seguintes pares:

$$R = \left\{ \begin{array}{ll} (\emptyset, \emptyset) & , \\ (\{1\}, \{1\}), (\{1\}, \{2\}) & , \\ (\{2\}, \{1\}), (\{2\}, \{2\}) & , \\ (\{1, 2\}, \{1, 2\}) & \end{array} \right\}$$

Com isso, temos que $[\{1\}] = \{\{1\}, \{2\}\} = [\{2\}]$, isto é, a classe de equivalência de 1, $[1]$, é igual a classe de equivalência de 2, $[2]$. ■

Note que, no exemplo anterior, as classes de equivalência de dois elementos distintos ($\{1\}$ e $\{2\}$) é o mesmo conjunto, $\{\{1\},\{2\}\}$. Este fato é válido para qualquer relação de equivalência e é demonstrado pelos teoremas seguintes[6].

Teorema 21. *Seja R uma relação de equivalência sobre um conjunto A e $x \in A$. Então $x \in [x]$.*

Demonstração. Suponha que R é uma relação de equivalência sobre um conjunto A e que $x \in A$. Como R é uma relação reflexiva e $x \in A$, temos que xRx. Já que xRx, pela definição de $[x]$, temos que $x \in [x]$. Portanto, se R é uma relação de equivalência sobre um conjunto A e que $x \in A$ então $x \in [x]$. □

Teorema 22. *Seja R uma relação de equivalência sobre um conjunto A. Então, para todo $x, y \in A$ temos que $y \in [x]$ se e somente se $[x] = [y]$.*

Demonstração. Suponha x, y arbitrários. Suponha $x, y \in A$. Suponha que $y \in [x]$. Suponha z arbitrário.

($\rightarrow$) : Suponha que $z \in [x]$. Pela definição de classe de equivalência, temos que zRx. Uma vez que $y \in [x]$, temos que yRx. Como R é simétrica e yRx, temos que xRy. Como R é transitiva, zRx e xRy temos que zRy e, assim, $z \in [y]$. Logo, se $z \in [x]$ então $z \in [y]$.

($\leftarrow$) : Suponha que $z \in [y]$. Como $y \in [x]$, temos que yRx. Já que $z \in [y]$, temos que zRy. Uma vez que zRy e yRx, já que R é transitiva, temos que zRx. Logo, se $z \in [y]$ então $z \in [x]$.

Como z é arbitrário, temos que $[x] = [y]$. Logo, se $y \in [x]$ então $[x] = [y]$. Assim, se $x, y \in A$ então se $y \in [x]$ então $[x] = [y]$. Como x, y são arbitrários temos que para todo $x, y \in A$ temos que $y \in [x]$ se e somente se $[x] = [y]$. □

Na próxima seção apresentaremos o conceito de partição de um conjunto e como este é relacionado ao conceito de classes de equivalência.

7.6.2 Partições e Classes de Equivalência

Partições são um tipo especial de famílias de conjuntos que possuem certas propriedades definidas a seguir.

Definição 69 (Partição)**.** Seja A um conjunto qualquer e $\mathcal{F} \subseteq \mathcal{P}(A)$ uma família de conjuntos. Dizemos que $\mathcal{F}$ é uma partição de A se as seguintes condições são verdadeiras.

1. $\bigcup \mathcal{F} = A$
2. $\mathcal{F}$ é disjunta par a par. Dizemos que uma família é par a par disjunta se $\forall X.\forall Y.X \neq Y \rightarrow X \cap Y = \emptyset$.
3. $\forall X.X \in F \rightarrow X \neq \emptyset$

■

Exemplo 131. Seja $A = \{1,2,3,4\}$, $\mathcal{F} = \{\{2\},\{1,3\},\{4\}\}$ e $\mathcal{G} = \{\{2\},\{1,3\},\{1,4\}\}$. Temos que a família $\mathcal{F}$ é partição de A, pois:

[6] É altamente recomendável que você faça os rascunhos destas demonstrações!

- $\bigcup \mathcal{F} = \{2\} \cup \{1,3\} \cup \{4\} = \{1,2,3,4\} = A$.
- Todos os conjuntos de $\mathcal{F}$ são disjuntos par a par.
- Nenhum conjunto de $\mathcal{F}$ é vazio.

Por sua vez, a família $\mathcal{G}$ não é uma partição de A pois, esta não é disjunta par a par. ■

De maneira intuitiva, toda relação de equivalência sobre um conjunto produz uma partição que corresponde aos conjuntos de elementos "iguais entre si" de acordo com o critério da relação. Mais ainda, toda partição de um conjunto gera uma relação de equivalência sobre este. Estes resultados são apresentados (e demonstrados) nos teoremas seguintes.

Teorema 23. *Seja R uma relação de equivalência sobre A. Então, A/R é uma partição de A.*

Demonstração. Para demonstrar que A/R devemos provar as 3 partes da definição de partição (definição 69).

1. Suponha x arbitrário.

   ($\rightarrow$) : Suponha que $x \in \bigcup A/R$. Como $x \in \bigcup A/R$, temos que existe y tal que $x \in [y]$ e $[y] \in A/R$. Se $x \in [y]$, temos que xRy e, logo, $x \in A$. Logo, se $x \in \bigcup A/R$ então $x \in A$.

   ($\leftarrow$) : Suponha que $x \in A$. Como $x \in A$, pelo teorema 21, temos que $x \in [x]$ e, portanto, $x \in \bigcup A/R$. Logo, se $x \in A$ então $x \in \bigcup A/R$.

   Portanto, como x é arbitrário temos que $\bigcup A/R = A$.

2. Suponha X e Y arbitrários. Suponha que $X, Y \in A/R$. Suponha que $X \cap Y \neq \emptyset$. Como $X \cap Y \neq \emptyset$, existe z tal que $z \in X$ e $z \in Y$. Como $X \in A/R$, existe $x \in A$ tal que $X = [x]$. Como $Y \in A/R$, existe $y \in A$ tal que $Y = [y]$. Como $z \in [x]$ e $z \in [y]$, pelo teorema 22, temos que $[x] = [y]$. Logo, se $X \neq Y$ então $X \cap Y = \emptyset$. Assim, se $X, Y \in A/R$ temos que se $X \neq Y$ então $X \cap Y = \emptyset$. Como X, Y são arbitrários, temos que A/R é disjunta par a par.

3. Suponha X arbitrário. Suponha que $X \in A/R$. Se $X \in A/R$, existe $x \in A$ tal que $X = [x]$. Pelo teorema 21, temos que se $x \in A$ então $x \in [x]$. Logo, $X \neq \emptyset$. Assim, se $X \in A/R$ então $X \neq \emptyset$. Como X é arbitrário, temos que o conjunto A/R não possui como elemento o conjunto vazio.

Portanto, temos que A/R é uma partição do conjunto A. □

Comentário 5. Ao invés de apresentar um rascunho da demonstração anterior, iremos comentar alguns pontos chaves desta que permitirão o leitor interessado a construir o rascunho.

Para mostrar que o conjunto

$$A/R = \{[x] \mid x \in A\}$$

é uma partição de um conjunto A, devemos provar as 3 partes da definição de partição. Isto é, devemos demonstrar que:

1. $\bigcup A/R = A$
2. A/R é uma família disjunta par a par.
3. $\forall X.X \in A/R \to X \neq \emptyset$

Para o primeiro item, utilizamos a definição da igualdade de conjuntos. Logo, devemos provar que

$$\forall x.x \in \bigcup A/R \leftrightarrow x \in A.$$

Para mostrar que $x \in \bigcup A/R \to x \in A$, usamos a definição de $x \in \bigcup A/R$: deve existir um conjunto $X \in A/R$ tal que $x \in X$. Porém, como A/R é o conjunto de classes de equivalências de A, temos que existe $y \in A$ tal que $X = [y]$ e, assim, temos que $x \in [y]$ e, portanto, temos que xRy o que nos permite concluir que $x \in A$, conforme requerido.

A implicação

$$x \in A \to x \in \bigcup A/R$$

é uma consequência do teorema 21.

No segundo item, devemos demonstrar que A/R é uma família disjunta par a par, isto é:

$$\forall X.\forall Y.X \in A/R \wedge Y \in A/R \to X \neq Y \to X \cap Y = \emptyset$$

Para demonstrar esta fórmula procedemos como usual: supomos X e Y arbitrários e que $X, Y \in A/R$, nos deixando com a seguinte implicação:

$$X \neq Y \to X \cap Y \neq \emptyset$$

Note que a implicação anterior é equivalente a

$$\neg(X = Y) \to \neg\exists x.x \in X \cap Y$$

e, portanto, como ambos os lados desta são negações, isto sugere que usemos uma prova pela contrapositiva. Logo, supomos que $X \cap Y \neq \emptyset$ e mostramos que $X = Y$. O restante da dedução envolve o uso do teorema 22 e das definições de classes de equivalência e A/R.

Finalmente, a terceira parte da demonstração é uma consequência imediata do teorema 21, que pode ser usado para concluir que nenhum conjunto de A/R é vazio. ■

Para demonstrar que toda partição produz uma relação de equivalência, primeiro devemos provar alguns resultados "auxiliares". Estes serão enunciados sem demonstração (prová-los fica como exercício) e os utilizaremos na prova deste teorema.

Teorema 24. *Suponha que A é um conjunto e $\mathcal{F}$ é uma partição de A. Então $R = \bigcup_{X \in \mathcal{F}}(X \times X)$ é uma relação de equivalência sobre A.*

Teorema 25. *Suponha que A é um conjunto, $\mathcal{F}$ é uma partição de A e $R = \bigcup_{X \in \mathcal{F}}(X \times X)$. Se $X \in \mathcal{F}$ e $x \in X$ então $X = [x]$.*

Finalmente, mostramos como uma partição produz uma relação de equivalência.

Teorema 26. *Suponha que A é um conjunto e que $\mathcal{F}$ é uma partição de A. Então, existe uma relação de equivalência sobre A tal que $A/R = \mathcal{F}$.*

Demonstração. Seja $R = \bigcup_{X \in \mathcal{F}}(X \times X)$. Pelo teorema 24, temos que R é uma relação de equivalência. Para mostrar que $A/R = \mathcal{F}$, suponha X arbitrário.

($\rightarrow$) : Suponha que $X \in A/R$. Como $X \in A/R$, temos que existe $x \in A$ tal que $X = [x]$. Como $x \in A$ e $\bigcup \mathcal{F} = A$ (pois $\mathcal{F}$ é uma partição de A), temos que $x \in A$. Logo, $x \in \bigcup \mathcal{F}$ e, portanto, deve existir $Y \in \mathcal{F}$ tal que $x \in Y$. Mas, pelo teorema 25, temos que $[x] = Y$. Assim, $X \in \mathcal{F}$. Como X é arbitrário, temos que $A/R \subseteq \mathcal{F}$.

($\leftarrow$) : Suponha $X \in \mathcal{F}$. Como F é uma partição, temos que $X \neq \emptyset$ e, portanto, existe $x \in X$. Pelo teorema 25, temos que $X = [x] \in A/R$. Logo, $\mathcal{F} \subseteq A/R$.

Assim $\mathcal{F} = A/R$, conforme requerido. □

7.6.3 Exercícios

1. Seja $A = \{1, 2, 3, 4\}$.
   (a) Apresente duas partições de A.
   (b) Para cada uma as partições apresentadas por você, apresente a relação de equivalência por elas gerada.
2. Suponha que R é uma relação reflexiva e transitiva sobre um conjunto A. Prove que $R \cap R^{-1}$ é uma relação de equivalência.
3. Prove os teoremas 24 e 25.

7.7 Fechos de Relações

Um problema comum em se manipular relações é o de que nem sempre as relações modeladas possuem certas propriedades de interesse (reflexivas, simétricas e transitivas). Nestes contextos é útil considerar o que chamamos de fechos de uma relação, que consiste na menor relação (com respeito a $\subseteq$) de forma a possuir certa propriedade. A próxima definição apresenta o conceito de fecho de uma relação

Definição 70 (Fecho de uma relação)**.** Seja R uma relação binária sobre um conjunto A. A relação S é o fecho de R com respeito a propriedade ρ se:

1. S possui a propriedade ρ.
2. $R \subseteq S$.
3. $\forall T.R \subseteq T \wedge T$ possui a propriedade $\rho \rightarrow S \subseteq T$, isto é S é a "menor" relação que contém S e possui a propriedade ρ.

■

As próximas seções apresentam os fechos para três propriedades de relações.

7.7.1 Fecho Reflexivo

Seja R uma relação sobre um conjunto A. Se R não é reflexiva é porque existe $a \in A$ tal que $\neg aRa$. Neste caso, para garantir que R é reflexiva basta fazer a união desta relação com

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

O fato de que o conjunto $R \cup i_A$ é o fecho reflexivo de uma relação R é expresso pelo teorema seguinte.

Teorema 27. *Seja R uma relação sobre um conjunto A. Então $R \cup i_A$, em que $i_A = \{(x, x) \mid x \in A\}$, é o fecho reflexivo de R.*

7.7.2 Fecho Simétrico

Seja R uma relação sobre um conjunto A. Se R não é simétrica é porque existem $a, b \in A$ tais que aRb e $\neg bRa$. Para tornar R uma relação simétrica, basta garantir que todo par presente em R possua o seu respectivo inverso. Isto é, dada uma relação R, o seu fecho simétrico é dado por $R \cup R^{-1}$.

Teorema 28. *Seja R uma relação sobre um conjunto A. Então $R \cup R^{-1}$ é o fecho simétrico de R.*

7.7.3 Fecho Transitivo

Seja R uma relação sobre um conjunto A. Note que para R não ser uma relação transitiva, temos que devem existir $a, b, c \in A$ tais que aRb, bRc e $\neg aRc$. Para entender melhor o fecho transitivo de uma relação, é conveniente considerarmos um exemplo simples.

Exemplo 132. Seja $A = \{1, 2, 3, 4\}$ e $R = \{(1, 2), (2, 3), (3, 4)\}$. Temos que R não é uma relação transitiva, pois:

- $1R2$ e $2R3$ mas $\neg 1R3$;
- $2R3$ e $3R4$ mas $\neg 2R4$.

Assim, podemos tentar tornar R transitiva incluindo os pares $(1, 3)$ e $(2, 4)$, obtendo a relação R' abaixo:

$$R' = \{(1, 2), (2, 3), (3, 4), (1, 3), (2, 4)\}$$

que ainda não é transitiva, já que $1R3$ e $3R4$ e $\neg 1R4$. Incluindo o par $1R4$, temos

$$R'' = \{(1, 2), (2, 3), (3, 4), (1, 3), (2, 4), (1, 4)\}$$

que é transitiva. Note que a necessidade do par $1R4$ surgiu quando da inclusão do par $1R3$. Os pares que faltam a relação R para que esta seja transitiva são os pares da forma aRc em que aRb e bRc, isto é, pares $(a, c) \in R \circ R = R^2$. Assim, ao acrescentarmos estes pares, estamos construindo a relação $R' = R \cup R^2$. De maneira similar, os pares que faltam a R' devem estar na relação $R' \circ R'$ que é igual a:

$$(R \cup R^2)^2 = R^2 \cup R^3 \cup R^4$$

■

Simplificando, temos que os pares que faltam para uma certa relação R se tornar transitiva são formados pela composição desta relação. O fecho transitivo de uma relação R, R^*, é definido como:

$$R^* = \bigcup_{i=1^\infty} R^i$$

Este resultado é formalizado pelo seguinte teorema.

Teorema 29. *Seja R uma relação sobre um conjunto A. O fecho transitivo de R é dado por:*

$$R^* = \bigcup_{i=1^\infty} R^i$$

Não discutiremos este teorema por este ser demonstrável utilizando uma técnica que veremos posteriormente: indução matemática.

7.7.4 Exercícios

1. Prove os teoremas 27 e 28.

7.8 Notas Bibliográficas

Relações são a base de diversos conceitos e áreas da ciência da computação. A fundamentação matemática de bancos de dados relacionais é feita sobre diversas operações sobre relações. Além disso, algoritmos para a solução de diversos problemas de inteligência artificial podem ser modelados como algoritmos sobre grafos, que essencialmente são relações.

Grande parte deste capítulo é baseado em [9].

8

Funções

Um matemático é uma função que
a partir de café produz teoremas.

Paul Ërdos — Matemático
Húngaro.

8.1 Motivação

Funções são provavelmente um dos conceitos matemáticos de maior importância para a Computação, já que funções podem ser entendidas como um modelo abstrato de algoritmo: a partir de uma ou mais entradas, uma função produz um único resultado. Além disso, este resultado é completamente determinado pela entrada: se aplicarmos repetidamente uma função ao mesmo argumento sempre obteremos o mesmo resultado.

A característica mais importante do conceito de função para a computação é que estas são um mecanismo de abstração. De maneira simples, para usarmos uma função precisamos apenas saber sua interface (o que ela recebe como parâmetros e o que ela retorna como resultado) e não como esta é implementada internamente. Este conceito de abstração é amplamente utilizado em computação.

Iniciaremos este capítulo definindo funções como um caso especial de relações e então consideraremos uma maneira "algorítmica" de se definir funções.

8.2 Introdução às funções

Como já dito na seção anterior, matematicamente, funções são apenas um caso especial de relações.

Definição 71 (Função)**.** Sejam A e B dois conjuntos quaisquer e $f \subseteq A \times B$. Dizemos que f é uma função se:

$$\forall x.x \in A \rightarrow \exists! b.b \in B \wedge (a, b) \in f$$

■

Notação 2. Utilizaremos a seguinte notação para representar que a relação $f \subseteq A \times B$ é uma função:

$$f : A \to B$$

e a notação $f(x) = y$ se $(x, y) \in f$ e f é uma função.

Como uma função é um caso especial de relação, temos que todas as definições de relações (domínio, imagem, contra-domínio) são aplicáveis.

Exemplo 133. Apresentaremos alguns exemplos de funções.

- Sejam $A = \{1, 2, 3\}$ e $B = \{4, 5, 6\}$ e $f = \{(1, 3), (2, 6), (3, 5)\}$. Temos que a relação f é uma função de A em B.
- Sejam $A = \{1, 2, 3\}$ e $B = \{4, 5, 6\}$ e $g = \{(1, 4), (2, 6), (1, 5), (3, 6)\}$. Temos que g não é uma função, pois $(1, 4) \in g$ e $(1, 5) \in g$.
- Sejam C o conjunto de todas as cidades e P o conjunto de todos os países. A relação $d = \{(c, p) \in C \times P \mid \text{a cidade } c \text{ está no país } p.\}$ é uma função, pois toda cidade pertence a um único país.
- Seja P o conjunto de todas as pessoas. A relação $h = \{(p, m) \in P \times \mathcal{P}(P) \mid \text{A pessoa } p \text{ é pai (ou mãe) das pessoas no conjunto } m\}$ é uma função, já que toda pessoa possui um único conjunto de filhos.

■

Dizemos que duas funções $f : A \to B$ e $g : A \to B$ são iguais se estas denotam o mesmo conjunto de pares. Este fato é expresso pelo teorema seguinte.

Teorema 30 (Extensionalidade)**.** *Suponha que $f : A \to B$, $g : A \to B$ e que $\forall x.x \in A \to f(x) = g(x)$. Então, $f = g$.*

Demonstração. Suponha (a, b) arbitrário.

$(\to)$: Suponha $(a, b) \in f$ arbitrário. Como $\forall x.x \in A \to f(x) = g(x)$ e $f(a) = b$, temos que $g(a) = b$. Logo, se $f(a) = b$ então $g(a) = b$.

$(\leftarrow)$: Suponha $(a, b) \in g$ arbitrário. Como $\forall x.x \in A \to f(x) = g(x)$ e $g(a) = b$, temos que $f(a) = b$. Logo, se $g(a) = b$ então $f(a) = b$.

Como (a, b) é arbitrário, temos que $f = g$. □

8.3 Funções, algoritmicamente

Matematicamente, funções são apenas um conjunto de pares ordenados. Porém, de um ponto de vista computacional, o conceito de função é distinto, pois este corresponde a noção de algoritmo. O problema com a versão "matemática" do conceito de função é que este não envolve variáveis cruciais em computação como consumo de memória e tempo de execução.

O objetivo desta seção é apresentar classes de funções que correspondem a algoritmos e mostrar que nem toda função pode ser programada em uma linguagem de programação.

8.3.1 Funções definidas recursivamente

No decorrer deste texto, apresentamos algumas funções recursivas: funções para determinar o conjunto de sub-fórmulas, variáveis livres de fórmulas da lógica. Podemos classificar uma função como sendo recursiva se esta está de acordo com a próxima definição.

Definição 72 (Funções de Recursivas Primitivas)**.** Seja A um conjunto qualquer. Dizemos que uma função $f : \mathbb{N} \to A \to A$ é definida recursivamente se a definição de f possui a seguinte estrutura, em que g, h são funções recursivas primitivas:

$$\begin{array}{lcl} f(0,x) & = & g\,x \\ f(n,x) & = & h(f(n-1,x), n-1, x) \end{array}$$

■

Funções definidas por recursão primitiva são garantidas de sempre terminar, isto é, atendem o critério de terminação[1]. Não demonstraremos este fato por este envolver conceitos que estão além dos propósitos desta disciplina. A seguir, apresentamos alguns exemplos de funções recursivas primitivas.

Exemplo 134. A função que a partir de um número natural retorna o seu quadrado é uma função recursiva primitiva, conforme apresentado a seguir.

$$g(x) \quad = \quad x \times x$$

■

O leitor deve estar se perguntando: "como esta função pode ser considerada primitiva recursiva se esta não usa recursão?". O fato é que toda função não recursiva atende o critério de função recursiva primitiva[2]. O próximo exemplo apresenta uma função recursiva primitiva que usa recursividade.

Exemplo 135. A função que calcula o fatorial de um número natural pode ser escrita da seguinte maneira, em que $\perp$ representa um valor semanticamente inválido (uma exceção).

$$\begin{array}{lcl} fact(n) & = & f(n, \perp) \end{array}$$

$$\begin{array}{lcl} f(0,x) & = & 1 \\ f(n,x) & = & n \times f(n-1,x) \end{array}$$

Esta função está de acordo com a definição 72 pois:

- A função h da definição pode ser representada pela função de multiplicação, que é primitiva recursiva.
- A função g da definição pode ser representada pela função constante que sempre retorna 1.

[1] Se você não se lembra deste critério, recomendo que releia o capítulo 1.

[2] Isto é fácil de se entender, informalmente. A idéia é que toda função recursiva primitiva possui **todas** as chamadas recursivas no formato da definição 72. Como definições não recursivas não possuem chamadas recursivas, estas atendem este critério por vacuidade.

Como exemplo, vamos considerar o cálculo de $fact(4)$, apresentado abaixo.

$$\begin{array}{ll} fact(4) & = \\ f(4, \perp) & = \\ 4 \times f(3, \perp) & = \\ 4 \times 3 \times f(2, \perp) & = \\ 4 \times 3 \times 2 \times f(1, \perp) & = \\ 4 \times 3 \times 2 \times 1 \times f(0, \perp) & = \\ 4 \times 3 \times 2 \times 1 \times 1 & = \\ 24 & \end{array}$$

■

Exemplo 136. A seguinte função não pode ser considerada como recursiva primitiva.

$$f(x) \quad = \quad \begin{cases} 0 & \text{Se } x = 0 \\ 1 & \text{Se } x = 1 \\ 1 + f(x \div 2) & \text{Se } x \text{ é par} \\ f(x) & \text{Caso contrário} \end{cases}$$

A definição apresentada não pode ser considerada uma função, já que esta não termina para números ímpares maiores que 1. ■

8.3.2 Funções totais e parciais

Normalmente, em matemática considera-se apenas funções totais, isto é funções que são definidas para todos os valores de um domínio. Porém, em computação, este conceito não é adequado. Neste sentido, considera-se também a noção de função parcial, que definimos a seguir.

Definição 73 (Funções totais e parciais)**.** Seja $f : A \to B$ uma função. Dizemos que f é uma função total se $dom(f) = A$. Por sua vez, dizemos que f é parcial se $dom(f) \subset A$. ■

Se f é uma função parcial e $x \in dom(f)$ então dizemos que $f(x)$ é definida. Caso $x \notin dom(f)$, dizemos que $f(x)$ é indefinida. Em algumas situações, ao invés de simplesmente dizermos que $f(x)$ é indefinida, associamos a x um valor não pertencente ao contradomínio da função f, que normalmente é representado como $\perp$. Neste caso, dizemos que $f(x)$ é indefinida se $f(x) = \perp$.

Exemplo 137. Considere a seguinte função, $f : \mathbb{N} \to \mathcal{C}$, em que $\mathcal{C}$ representa o conjunto de símbolos (letras) do alfabeto latino.

$$f(i) \quad = \quad c, \text{ em que } c \text{ é a } i\text{-ésima letra do alfabeto}$$

Evidentemente, temos que esta função é parcial, visto que não é definida para todo $n \in \mathbb{N}$. O domínio de f é:

$$dom(f) = \{x \in \mathbb{N} \mid 1 \leq x \leq 26\}$$

já que o alfabeto latino possui 26 letras. Com isso, temos que f é indefinida para todo elemento de $\{x \in \mathbb{N} \mid x \geq 27\}$. ■

Em matemática, é simples descobrir para quais elementos uma função é ou não definida. Porém, em computação, este nem sempre é o caso. Como funções são modelos abstratos de programas, determinar o domínio de uma função (isto é, determinar se esta vai ser aplicada somente a valores considerados válidos) é tarefa de um compilador. Uma forma limitada deste tipo de análise é feita considerando tipos de valores e parâmetros de uma função. Em linguagens de programação modernas, compiladores são capazes de validar que somente valores de tipos adequados sejam argumentos para chamadas de funções. Normalmente, erros devido a passagem inválida de parâmetros podem ocasionar erros em tempo de execução[3]. Porém, mesmo valores de tipo adequados podem ocasionar erros em tempo de execução. Para isso, basta que o valor, apesar de possuir o tipo correto especificado para o parâmetro da função, não pertença ao domínio desta. Em computação, valores indefinidos para um determinado valor do domínio, são usualmente representados utilizando exceções que, se não tratadas, causam a interrupção do código.

8.4 Composição de funções

Nesta seção, vamos reapresentar o conceito de composição (agora de funções) e provar que esta operação é associativa.

Definição 74 (Composição de Funções)**.** Sejam $f : A \to B$ e $g : B \to C$. Definimos $g \circ f : A \to C$ como:

$$g \circ f(x) = g(f(x))$$

■

Teorema 31 (Associatividade da composição de funções)**.** *Sejam $f : A \to B$, $g : B \to C$ e $h : C \to D$. Temos que $h \circ (g \circ f) = (h \circ g) \circ f$.*

Demonstração. Suponha x arbitrário. Suponha que $x \in A$. Temos:

$$\begin{array}{lcl} ((h \circ g) \circ f)(x) & = & \\ (h \circ g)(f(x)) & = & \{\text{pela def. de } \circ\} \\ h(g(f(x))) & = & \{\text{pela def. de } \circ\} \\ h((g \circ f)(x)) & = & \{\text{pela def. de } \circ\} \\ (h \circ (g \circ f))(x) & & \end{array}$$

□

8.5 Propriedades de funções

Nesta seção descrevemos algumas propriedades de funções que provavelmente já são conhecidas pelo leitor.

[3]Programadores C/C++ estão habituados a este tipo de erro caracterizado pela mensagem: "`segmentation fault.`"

8.5.1 Funções injetoras

O requisito para uma relação ser considerada uma função é que esta associe cada elemento do domínio a um único elemento do contradomínio. Por sua vez, dizemos que uma função é injetiva se cada elemento da imagem desta função é o resultado de aplicá-la a somente um elemento do domínio. A definição seguinte formaliza este conceito.

Definição 75 (Função injetora)**.** Dizemos que $f : A \to B$ é uma função injetora se a seguinte fórmula é verdadeira:

$$\forall x.\forall y.x \in A \wedge y \in A \to x \neq y \to f(x) \neq f(y)$$

■

Exemplo 138. Sejam $A = \{1, 2\}$, $B = \{3, 4, 5\}$ e $f : A \to B$ definida como $f = \{(1, 3), (2, 5)\}$. Como cada imagem possui um único valor associado no domínio, temos que f é uma função injetora. ■

Exemplo 139. Seja $f : \mathbb{N} \to \mathbb{N}$ definida como $f(x) = x^2$. Temos que esta função é injetora, já que esta associa a cada quadrado perfeito de $\mathbb{N}$ a sua raiz. ■

O leitor atento deve notar que esta função é injetora quando definida sobre $\mathbb{N}$. Para os conjuntos $\mathbb{Z}, \mathbb{R}$ esta não o é, uma vez que, para quaisquer números x e $-x$, o quadrado destes sempre será $|x|^2$. De maneira mais precisa, temos que se $f : \mathbb{Z} \to \mathbb{Z}$ ou $f : \mathbb{R} \to \mathbb{R}$, f não é uma função injetora.

A classe das funções injetoras são fechadas sobre a operação de composição, isto é, a composição de duas funções injetoras é também uma função injetora. Este resultado é formalizado no teorema seguinte.

Teorema 32. *Sejam $f : A \to B$ e $g : B \to C$ duas funções injetoras. Então $g \circ f : A \to C$ também é uma função injetora.*

Demonstração. Suponha que $f : A \to B$ e $g : B \to C$ são funções injetoras. Suponha x, y arbitrários. Suponha $x, y \in A$. Suponha que $x \neq y$. Suponha, por contradição, que $(g \circ f)(x) = (g \circ f)(y)$. Pela definição de composição de funções, temos que $g(f(x)) = g(f(y))$, mas como g é injetora, temos que $f(x) = f(y)$. Mas, como f também é injetora, temos que $x = y$, o que contradiz a suposição de que $x \neq y$. Logo, $(g \circ f)(x) \neq (g \circ f)(y)$. Assim, se $x \neq y$ então $(g \circ f)(x) \neq (g \circ f)(y)$. Logo, se $x, y \in A$ e $x \neq y$ então $(g \circ f)(x) \neq (g \circ f)(y)$. Como x, y são arbitrários, temos que $g \circ f : A \to C$ é uma função injetora. Portanto, se $f : A \to B$ e $g : B \to C$ são funções injetoras, então temos que $g \circ f : A \to C$ é uma função injetora. □

Utilizando o conceito de funções injetoras, podemos expressar de maneira mais elegante o chamado princípio da casa dos pombos.

Teorema 33 (Princípio da casa dos pombos)**.** *Sejam A e B dois conjuntos finitos tais que $|A| > |B|$, $|A| > 1$ e $f : A \to B$ uma função total. Então:*

$$\exists a_1.\exists a_2.a_1 \in A \wedge a_2 \in A \wedge a_1 \neq a_2 \wedge f(a_1) = f(a_2)$$

Demonstração. Como f é total, temos que todo $x \in A$ é tal que existe $y \in B$ de forma que $f(x) = y$. Porém, como $|A| > |B|$, temos que existirão $a_1, a_2 \in A$, $a_1 \neq a_2$ tais que $f(a_1) = f(a_2)$. □

8.5.2 Funções sobrejetoras

Dizemos que uma função é sobrejetora se sua imagem é todo o seu contradomínio. Formalizamos esse conceito na próxima definição.

Definição 76 (Função sobrejetora)**.** Dizemos que uma função $f : A \to B$ é uma função sobrejetora se a seguinte fórmula é verdadeira:

$$\forall b.b \in B \to \exists a.a \in A \wedge f(a) = b$$

■

Exemplo 140. Seja $A = \{1, 2, 3\}$, $B = \{4, 5\}$ e $f : A \to B$ definida como $f = \{(1, 4), (2, 5), (3, 5)\}$. Temos que f é uma função sobrejetora, uma vez que sua imagem é igual ao conjunto B (contradomínio de f). Note que apesar de ser sobrejetora, a função f não é injetora uma vez que $f(2) = 5 = f(3)$. ■

Exemplo 141. A função $f : \mathbb{N} \to \mathbb{N}$ definida como $f(x) = x^2$ não é sobrejetora, uma vez que não existe $x \in \mathbb{N}$ tal que $x^2 = 3$. Na verdade, esta função possui como imagem o conjunto de quadrados perfeitos de $\mathbb{N}$. ■

Assim como as funções injetoras, a classe das funções sobrejetoras também é fechada sobre a operação de composição. Esse fato é expresso pelo teorema seguinte.

Teorema 34. *Sejam $f : A \to B$ e $g : B \to C$ duas funções sobrejetoras. Então, $g \circ f : A \to C$ é uma função sobrejetora.*

8.5.3 Funções bijetoras

Nesta seção apresentaremos uma classe importante de funções: as funções bijetoras.

Definição 77 (Função bijetora)**.** Dizemos que $f : A \to B$ é uma função bijetora se f é uma função injetora e sobrejetora. ■

Caso uma função é bijetora, dizemos que está é uma correspondência de um para um, conforme definido pelo teorema a seguir.

Teorema 35. *Seja $f : A \to B$ uma função bijetora. Então, $|dom(f)| = |ran(f)|$.*

Demonstração. Suponha que $|dom(f)| > |ran(f)|$. Como $|dom(f)| > |ran(f)|$, temos que f não pode ser injetora (pelo princípio da casa dos pombos). Suponha que $|dom(f)| < |ran(f)|$. Desta forma, temos que f não pode ser sobrejetora. Assim, temos que se f é bijetora então $|dom(f) = |ran(f)|$. □

Evidentemente, temos que a classe de funções bijetoras é fechada sobre a operação de composição de funções. Este fato é expresso pelo teorema seguinte.

Teorema 36. *Sejam $f : A \to B$ e $g : B \to C$ funções bijetoras. Então, $g \circ f : A \to C$ é uma função bijetora.*

Demonstração. Consequência imediata dos teoremas 32 e 34. □

8.5.4 Função inversa

Como funções são apenas um caso especial de relações, podemos utilizar a noção de inversa de uma relação à funções. Porém, nem sempre a inversa de uma função será uma função. Os próximos exemplos ilustram estes problemas.

Exemplo 142. Sejam $A = \{1, 2, 3\}$, $B = \{3, 4, 5\}$ e $f = \{(1, 3), (2, 3), (3, 5)\}$. Temos que a inversa da função f é

$$f^{-1} = \{(3, 1), (3, 2), (5, 3)\}$$

não pode ser considerada uma função já que não existe uma única imagem para o valor $3 \in B$. ■

Note que este problema é devido ao fato de que a função $f : A \to B$ não era injetora. Será que isto é suficiente para garantir que a inversa de uma função seja também uma função? Veremos que somente o fato de uma função ser injetora não é suficiente para que sua inversa também seja uma função. O próximo exemplo ilustra este caso.

Exemplo 143. Sejam $A = \{1, 2, 3\}$, $B = \{4, 5, 6, 7\}$ e $f = \{(1, 4), (2, 5), (3, 6)\}$. Desta forma temos que a inversa de f é dada pelo seguinte conjunto:

$$f^{-1} = \{(4, 1), (5, 2), (6, 3)\}$$

que não pode ser considerada uma função de $B \to A$ pois, $7 \in B$, mas $7 \notin dom(f^{-1})$ e, portanto, f^{-1} não pode ser considerada uma função de $B \to A$. ■

Note que o motivo que evitou que a inversa da função f do exemplo anterior fosse considerada uma função é que a imagem desta não era igual ao seu contradomínio, isto é, $f : A \to B$ não era uma função sobrejetora.

Desta forma, podemos concluir que a condição necessária e suficiente para que uma função $f : A \to B$ possua uma função inversa é que f seja uma função bijetora. Este fato é expresso pelo teorema seguinte:

Teorema 37. *Seja $f : A \to B$ uma função bijetora. Então $f^{-1} : B \to A$.*

Demonstração. Suponha que $f : A \to B$ seja uma função bijetora. Suponha b arbitrário. Suponha $b \in B$. Para mostrar que f^{-1} é uma função devemos mostrar que existe um único valor $a \in A$ tal que $f(b) = a$.

1. Existência: Como f é sobrejetora, temos que existe $a \in A$ tal que $f(a) = b$ e, portanto, $f^{-1}(b) = a$.
2. Unicidade: Suponha que $f(a_1) = b$ e $f(a_2) = b$. Como f é injetora, temos que se $f(a_1) = f(a_2)$ então $a_1 = a_2$.

□

Outra propriedade importante de funções bijetoras e suas inversas é que a composição destas produz como resultado a função identidade. O seguinte teorema expressa esse fato.

Teorema 38. *Seja $f : A \to B$ uma função bijetora e $f^{-1} : B \to A$ sua inversa. Então, temos que $f^{-1} \circ f = i_A$ e $f \circ f^{-1} = i_B$.*

8.5.5 Exercícios

1. Responda o que se pede

   (a) Seja $A = \{1, 2, 3\}$, $B = \{4\}$ e $f = \{(1, 4), (2, 4), (3, 4)\}$. Neste caso, $f : A \to B$? Justifique.

   (b) Seja $A = \{1\}$, $B = \{2, 3, 4\}$ e $f = \{(1, 2), (1, 3), (1, 4)\}$. Neste caso, $f : A \to B$? Justifique.

   (c) Seja C o conjunto de todos automóveis registrados no Brasil e S o conjunto de todas as strings de tamanho finito formadas por letras e números. Então $L = \{(c, s) \mid s \text{ é a placa do automóvel } c\}$ é uma função?

2. Sejam $f, g : \mathbb{R} \to \mathbb{R}$ funções definidas como:

$$f(x) = \frac{1}{x^2+2} \qquad g(x) = 2x - 1$$

   Encontre fórmulas para $(f \circ g)(x)$ e $(g \circ f)(x)$.

3. Suponha $f : A \to B$ e $C \subseteq A$. O conjunto $f \cap (C \times B)$ é uma relação de C em B, chamada de restrição de f com respeito a C, e é representado como $f \uparrow C$. Isto é: $f \uparrow C = f \cap (B \times C)$.

   - Prove que $f \uparrow C : C \to B$ e que para todo $c \in C$, $f(c) = (f \uparrow C)(c)$.
   - Prove que se f é injetora, então $f \uparrow C$ também é.
   - Prove que se f é sobretora, então $f \uparrow C$ também é.

4. Suponha $f : A \to B$ e S uma relação binária sobre B. Defina R como sendo:

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

   (a) Prove que se S é reflexiva, então R também é.

   (b) Prove que se S é transitiva, então R também é.

   (c) Prove que se S é simétrica, então R também é.

5. Prove os teoremas 34 e 38.

8.6 Notas Bibliográficas

Parte III

Indução e Recursividade

9

Indução Matemática

> Induction makes you fell guilty for getting something out of nothing, and it is artificial, but it is one of the greatest ideas of civilization.
>
> Helbert S. Wilf, Matemático Norte-americano.

9.1 Motivação

Na ciência e filosofia usamos, essencialmente, dois tipos de raciocínio distintos: o dedutivo e indutivo. O raciocínio dedutivo é governado por leis da lógica e foi tema de grande parte do curso de matemática discreta. Se certo fato é deduzido usando lógica, este é irrefutável, visto que, sistemas dedutivos para lógica são corretos e completos. O raciocínio indutivo, por sua vez, é o que usamos quando inferimos um padrão de comportamento futuro a partir de experiências realizadas no passado. Este tipo de raciocínio, apesar de útil em ciências experimentais, não é de interesse para os objetivos deste texto. Em especial, estamos interessados na chamada indução matemática, uma técnica de demonstração muito utilizada em diversas áreas da computação. O objetivo deste capítulo é apresentar esta técnica de prova e como esta é usada para demonstração de diversos fatos em matemática e em ciência da computação.

9.2 Introdução à indução matemática

A técnica de indução matemática pode ser utilizada para demonstrar conclusões que possuam a seguinte estrutura:

$$\forall n.n \in \mathbb{N} \rightarrow P(n)$$

Evidentemente, para demonstrar a fórmula anterior devemos provar que a propriedade P é verdadeira para cada um dos valores do conjunto

$$\mathbb{N} = \{0, 1, 2, 3...\}$$

Como existem infinitos valores em $\mathbb{N}$, não podemos simplesmente verificar se a propriedade em questão é verdadeira para cada um destes valores. Então, como demonstrar tais propriedades? A chave da indução matemática está na própria estrutura do conjunto de números naturais. Note que para listar todos os valores de $\mathbb{N}$, tudo o que temos que fazer é iniciar com $0 \in \mathbb{N}$ e repetidamente somar 1, produzindo assim, um novo número natural. Assim, para mostrar que todos os números naturais possuem uma certa propriedade P, basta:

- Mostrar que 0 possui a propriedade P, isto é, $P(0)$ é verdadeiro.
- Mostrar que sempre que um número natural n possuir a propriedade P então o sucessor de n, $n+1$, também possuirá essa propriedade. Isto é, devemos provar que $\forall n.n \in \mathbb{N} \wedge P(n) \rightarrow P(n+1)$.

Exatamente esta observação motiva a seguinte estratégia de prova.

Estratégia de Prova 14 (Para provar uma conclusão da forma $\forall n.n \in \mathbb{N} \rightarrow P(n).$)**.** Prove que a seguinte fórmula é verdadeira.

$$P(0) \wedge \forall n.n \in \mathbb{N} \wedge P(n) \rightarrow P(n+1)$$

Texto: Caso Base: [Prova de $P(0)$].
Passo Indutivo:[Prova de $\forall n.n \in \mathbb{N} \wedge P(n) \rightarrow P(n+1)$].

■

Porém, como somente a demonstração destes dois passos pode comprovar que $\forall n.n \in \mathbb{N} \rightarrow P(n)$? A idéia é bastante simples. Note que ao usarmos indução matemática, provamos as seguintes fórmulas:

- $P(0)$
- $\forall n.n \in \mathbb{N} \wedge P(n) \rightarrow P(n+1)$

Observe que temos que para qualquer número natural n, $P(n) \rightarrow P(n+1)$. Como esta fórmula é verdadeira para todo $n \in \mathbb{N}$, temos que esta é verdadeira também para $n = 0$. Eliminando o quantificador universal substituindo n por 0, obtemos

$$0 \in \mathbb{N} \wedge P(0) \rightarrow P(1)$$

Porém, é óbvio que $0 \in \mathbb{N}$ e como provamos que $P(0)$, por eliminação da implicação podemos concluir que $P(1)$ também é verdadeiro. Repetindo este processo para $n = 1$, temos que a seguinte implicação é obtida a partir da eliminação do quantificador universal:

$$1 \in \mathbb{N} \wedge P(1) \rightarrow P(2)$$

o que nos permite deduzir que $P(2)$ é verdadeiro. Note que podemos repetir esse processo para concluir que P é verdadeira para qualquer $n \in \mathbb{N}$. A seguir, apresentaremos um exemplo simples de uma propriedade provável por indução matemática.

Teorema 39. *Para todo $n \in \mathbb{N}$, temos que $\sum_{k=0}^{n} 2^k = 2^{n+1} - 1$*

Comentário 6. Note que este teorema pode ser expresso pela seguinte fórmula:

$$\forall n.n \in \mathbb{N} \wedge \sum_{k=0}^{n} 2^k = 2^{n+1} - 1$$

em que a propriedade $P(n)$ é

$$\sum_{k=0}^{n} 2^k = 2^{n+1} - 1$$

logo, de acordo com a estratégia de prova 14, devemos provar as seguintes fórmulas:

- $P(0)$
- $\forall n.n \in \mathbb{N} \wedge P(n) \rightarrow P(n+1)$

em que $P(0)$ é dada por

$$\sum_{k=0}^{0} 2^k = 2^{0+1} - 1$$

que é facilmente demonstrada como verdadeira pela seguinte equação:

$$\begin{array}{lll} \sum_{k=0}^{0} 2^k & = & \\ 2^0 & = & \{\text{pela def. da noção } \Sigma\} \\ 1 & = & \\ 2-1 & = & \\ 2^{0+1}-1 & & \end{array}$$

Logo, $P(0) = \sum_{k=0}^{0} 2^k = 2^{0+1} - 1$ é verdadeiro.

No próximo passo, temos que demonstrar o passo indutivo, que é dado pela seguinte fórmula:

$$\forall n.n \in \mathbb{N} \wedge \left(\sum_{k=0}^{n} 2^k = 2^{n+1} - 1 \right) \rightarrow \left(\sum_{k=0}^{n+1} 2^k = 2^{n+2} - 1 \right)$$

Para demonstrar o passo indutivo, supomos $n \in \mathbb{N}$ arbitrário e que $\sum_{k=0}^{n} 2^k = 2^{n+1} - 1$. Usualmente, damos o nome de hipótese de indução a suposição de que a propriedade que desejamos provar é verdadeira para um número n qualquer, isto é que $P(n)$ é verdadeira.

Para concluir a prova, resta mostrar que a seguinte igualdade é verdadeira:

$$\sum_{k=0}^{n+1} 2^k = 2^{n+2} - 1$$

que é facilmente demonstrável usando a hipótese de indução e álgebra. A demonstração desta equação é apresentada a seguir.

$$\begin{array}{lll} \sum_{k=0}^{n+1} 2^k & = & \\ \sum_{k=0}^{n} 2^k + 2^{n+1} & = & \{\text{pela definição de } \Sigma\} \\ 2^{n+1} - 1 + 2^{n+1} & = & \{\text{pela hipótese de indução}\} \\ 2^{n+2} - 1 & & \end{array}$$

A seguir, apresentamos a versão final do texto desta demonstração. ■

Demonstração.

Caso Base: Para $n = 0$, temos:

$$\begin{array}{lll} \sum_{k=0}^{0} 2^k & = & \\ 2^0 & = & \{\text{pela def. da notação } \Sigma\} \\ 1 & = & \\ 2-1 & = & \\ 2^{0+1}-1 & & \end{array}$$

Passo indutivo: Suponha n arbitrário. Suponha $n \in \mathbb{N}$ e que $\sum_{k=0}^{n} 2^k = 2^{n+1} - 1$. Temos:

$$\begin{array}{lll} \sum_{k=0}^{n+1} 2^k & = & \\ \sum_{k=0}^{n} 2^k + 2^{n+1} & = & \{\text{pela definição de } \Sigma\} \\ 2^{n+1} - 1 + 2^{n+1} & = & \{\text{pela hipótese de indução}\} \\ 2^{n+2} - 1 & & \end{array}$$

□

Apresentaremos mais alguns exemplos de demonstração por indução.

Teorema 40. *Para todo $n \in \mathbb{N}$, $3 \mid (n^3 - n)$.*

Comentário 7. Novamente, utilizaremos indução matemática. Note que o enunciado do teorema é dado por:

$$\forall n.n \in \mathbb{N} \rightarrow 3 \mid (n^3 - n)$$

em que a propriedade a ser demonstrada é

$$3 \mid (n^3 - n)$$

Note que $3 \mid (n^3 - n)$ é na verdade uma fórmula envolvendo um quantificador existencial.

$$3 \mid (n^3 - n) \equiv \exists k.k \in \mathbb{N} \wedge n^3 - n = 3k$$

Logo, temos que mostrar a veracidade das seguintes fórmulas, para demonstrar o teorema usando indução:

- $3 \mid (0^3 - 0)$
- $\forall n.n \in \mathbb{N} \wedge 3 \mid (n^3 - n) \rightarrow 3 \mid ((n+1)^3 - (n+1))$

A demonstração de $3 \mid (0^3 - 0)$ envolve provar a seguinte fórmula

$$\exists k.k \in \mathbb{N} \wedge 3k = 0$$

que é evidentemente verdadeira, basta fazer que $k = 0$.

Para o passo indutivo, devemos demonstrar que

$$\forall n.n \in \mathbb{N} \wedge 3 \mid (n^3 - n) \rightarrow 3 \mid ((n+1)^3 - (n+1))$$

Iniciamos a demonstração supondo $n \in \mathbb{N}$ arbitrário e que $3 \mid (n^3 - n)$ e devemos provar que

$$\exists k.k \in \mathbb{N} \wedge (n+1)^3 - (n+1) = 3k$$

Logo, devemos encontrar um valor de $k \in \mathbb{N}$ que torne a fórmula anterior verdadeira. Como $3 \mid (n^3 - n)$, existe $a \in \mathbb{N}$ tal que $n^3 - n = 3a$. Para encontrar uma "dica" de qual seria este valor de k, vamos desenvolver o polinômio $(n+1)^3 - (n+1)$:

$$\begin{array}{ll} (n+1)^3 - (n+1) & = \\ n^3 + 3n^2 + 3n + 1 - (n+1) & = \\ n^3 + 3n^2 + 3n + 1 - n - 1 & = \\ n^3 - n + 3n^2 + 3n & \end{array}$$

Porém, pela hipótese de indução, temos que $n^3 - n = 3a$. Com isso, temos:

$$\begin{array}{ll} n^3 - n + 3n^2 + 3n & = \\ 3a + 3n^2 + 3n & = \\ 3(a + n^2 + n) & \end{array}$$

que obviamente é divisível por 3. Logo, para concluir a demonstração, basta escolher $k = a + n^2 + n$. O texto final da prova é apresentado a seguir. ■

Demonstração.

Caso base: Seja $k = 0$. Temos que $0^3 - 0 = 3.0$, conforme requerido.

Passo indutivo: Suponha $n \in \mathbb{N}$ arbitrário e que $3 \mid n^3 - n$. Como $3 \mid n^3 - n$, existe $a \in \mathbb{N}$ tal que $n^3 - n = 3a$. Seja $k = a + n^2 + n$. Temos:

$$\begin{array}{lll} 3k & = & \\ 3(a + n^2 + n) & = & \{\text{por } k = a + n^2 + n\} \\ 3a + 3n^2 + 3n & = & \\ (n^3 - n) + 3n^2 + 3n & = & \{\text{pela hipótese de indução}\} \\ n^3 - n + 3n^2 + 3n + 1 - 1 & = & \\ (n^3 + 3n^2 + 3n + 1) - n - 1 & = & \\ (n+1)^3 - (n+1) & & \end{array}$$

Logo, $3 \mid (n+1)^3 - (n+1)$, conforme requerido.

□

Terminamos essa seção com um exemplo de demonstração por indução que não envolve equações, mas sim inequações.

Teorema 41. *Para todo $n \in \mathbb{N}$, $n \geq 5$ temos que $2^n > n^2$.*

Demonstração.

Caso base: Para $n = 5$, temos que $2^5 > 25$, conforme requerido.

Passo indutivo: Suponha $n \in \mathbb{N}$ arbitrário e que $2^n > n^2$. Temos:

$$\begin{array}{lll} 2^{n+1} & = & \\ 2^n + 2^n & > & \\ n^2 + n^2 & > & \{\text{pela hipótese de indução.}\} \\ n^2 + 2n + 1 & = & \\ (n+1)^2 & & \end{array}$$

Logo, $2^{n+1} > (n+1)^2$, conforme requerido.

□

Comentário 8. A demonstração anterior utiliza praticamente a mesma estrutura das anteriores, com duas diferenças:

1. O caso base deve ser para $n = 5$, uma vez que o teorema é válido para valores $n \geq 5$.
2. No passo indutivo, provamos a desigualdade desejada como uma sequência de igualdades / desigualdades. Este tipo de demonstração, comum em matemática, é válido para qualquer relação binária transitiva (note que tanto $>$, quanto $=$ são transitivas).

■

9.2.1 Exercícios

1. Prove os seguintes teoremas.
   (a) Para todo $n \in \mathbb{N}$, $\sum_{i=0}^{n} 3^i = \frac{3^{n+1}-1}{2}$
   (b) Para todo $n \in \mathbb{N}$, $\sum_{i=0}^{n} i^2 = \frac{n(n+1)(2n+1)}{6}$
   (c) Para todo $n \in \mathbb{N}$, $\sum_{i=1}^{n}(8i-5) = 4n^2 - n$
   (d) Para todo $n \geq 1$, $5 \mid (n^5 - n)$
   (e) Para todo $n \geq 1$, $6 \mid (7^n - 1)$
   (f) Para todo $n \geq 1$, $6 \mid (n^3 + 5n)$
   (g) Para todo $n \in \mathbb{N}$, $2 \mid (n^2 + n)$

9.3 Indução Forte

Em algumas situações, a suposição de que a propriedade a ser provada é válida para um número $n \in \mathbb{N}$ não é forte o suficiente para concluirmos a demonstração. Para ilustrar esse problema, vamos apresentar um teorema e tentar prová-lo usando a técnica de indução vista na seção anterior. Na sequência, apresentaremos a técnica de indução forte e a justificaremos de maneira informal, como feito para a indução simples. Finalmente, concluiremos esta seção demonstrando o teorema utilizado como motivador para a indução forte e mais alguns exemplos desta técnica de prova.

Utilizaremos o chamado teorema fundamental da aritmética que afirma que todo número inteiro $n > 1$ é primo ou produto de primos. O enunciado formal deste teorema é apresentado a seguir:

Teorema 42. *Para todo $n \in \mathbb{N}$, se $n > 1$ então n é primo ou é um produto de números primos.*

Note que o enunciado deste teorema é dado pela seguinte fórmula:

$$\forall n.n \in \mathbb{N} \rightarrow n > 1 \rightarrow n \text{ é primo} \vee n \text{ é produto de primos}$$

Como esta fórmula possui o formato exigido para demonstrações usando indução, podemos tentar provar o caso base ($n = 2$) e o passo indutivo. Porém, no passo indutivo chegamos em um possível "beco-sem-saída":

> Caso base: Para $n = 2$, temos que 2 é primo e portanto, 2 é primo ou produto de primos.
>
> Passo indutivo. Suponha $n \in \mathbb{N}$, $n > 1$ arbitrário e que n é primo ou produto de primos. Considere os seguintes casos:
>
> > n é primo: [Prova de $n + 1$ é primo ou produto de primos]
> >
> > n é produto de primos: [Prova de $n+1$ é primo ou produto de primos]

Note que no caso de n ser primo, não podemos garantir que $n + 1$ será sempre primo ou produto de primos, visto que para $n = 2$, temos que $n+1 = 3$ é primo. Por sua vez, se supormos que n é um produto de primos, temos que $n + 1$ pode ser ou não primo. Então como concluir esta demonstração?

Note que se um número n não é primo, necessariamente existem a e b tais que $1 < a, b < n$ e $n = a.b$, em que a e b podem ser ou não números primos. Se ambos forem primos, temos que n será um produto de primos. Por sua vez, se a ou b for um produto de primos, temos que n também o será. Apesar de correto, não podemos utilizar este raciocínio em uma prova por indução, pois a hipótese de indução supõe que a propriedade que desejamos provar é verdadeira para um valor fixo n qualquer e não para qualquer valor.

Nestas situações em que precisamos de uma hipótese de indução que seja válida não apenas para um valor, devemos usar a indução forte que nos permite supor que a propriedade a ser demonstrada é válida para todos os números naturais menores que um número n. O princípio de indução forte é apresentado na estratégia de prova seguinte.

Estratégia de Prova 15 (Para provar $\forall n.n \in \mathbb{N} \rightarrow P(n)$)**.** Prove que a seguinte fórmula é verdadeira:

$$\forall n.n \in \mathbb{N} \rightarrow (\forall k.k \in \mathbb{N} \wedge k < n \rightarrow P(k)) \rightarrow P(n)$$

■

O ponto chave de uma demonstração usando indução está em sua hipótese de indução:

$$\forall k.k \in \mathbb{N} \wedge k < n \rightarrow P(k)$$

que especifica que a propriedade P é verdadeira não para apenas um número qualquer, mas sim para todos valores $k < n$. Ao observarmos a fórmula da estratégia de indução forte, notamos que esta não possui "casos base". Então, como esta pode ser equivalente a $\forall n.n \in \mathbb{N} \rightarrow P(n)$? Para a indução convencional, mostramos que esta é equivalente a $\forall n.n \in \mathbb{N} \rightarrow P(n)$ usando o caso

base e uma sequência de eliminações do quantificador universal e implicações[1]. Para mostrar que indução forte é uma técnica de prova válida, utilizaremos uma estratégia similar a usada para indução convencional. Para a fórmula

$$\forall n.n \in \mathbb{N} \to (\forall k.k \in \mathbb{N} \land k < n \to P(k)) \to P(n)$$

ser verdadeira, esta deverá o ser para todos os valores de $n \in \mathbb{N}$. Logo, para $n = 0$, esta também será verdadeira. Substituindo n por 0 obtemos:

$$0 \in \mathbb{N} \to (\forall k.k \in \mathbb{N} \land l < 0 \to P(k)) \to P(0)$$

Porém, como não existe $k \in \mathbb{N}$ tal que $k < 0$, temos que $k < 0$ é equivalente a $\bot$. Logo, usando álgebra, temos:

$$\begin{array}{ll} 0 \in \mathbb{N} \to (\forall k.k \in \mathbb{N} \land k < 0 \to P(k)) \to P(0) & \equiv \\ 0 \in \mathbb{N} \to (\forall k.k \in \mathbb{N} \land \bot \to P(k)) \to P(0) & \equiv \\ 0 \in \mathbb{N} \to (\forall k.\bot \to P(k)) \to P(0) & \equiv \\ 0 \in \mathbb{N} \to (\top \to P(0)) & \equiv \\ 0 \in \mathbb{N} \to (\neg\top \lor P(0)) & \equiv \\ 0 \in \mathbb{N} \to (\bot \lor P(0)) & \equiv \\ 0 \in \mathbb{N} \to P(0) & \end{array}$$

que é equivalente a $P(0)$, por eliminação da implicação usando o fato de que $0 \in \mathbb{N}$. Logo, para $n = 0$, a fórmula da indução forte é equivalente a $P(0)$. Seguindo o mesmo raciocínio, para $n = 1$ temos:

$$1 \in \mathbb{N} \to (\forall k.k \in \mathbb{N} \land l < 1 \to P(k)) \to P(1)$$

mas, como o único valor de $k \in \mathbb{N}$ tal que $k < 1$ é $k = 0$, temos que a fórmula anterior é equivalente a:

$$\begin{array}{ll} 1 \in \mathbb{N} \to (\forall k.k \in \mathbb{N} \land l < 1 \to P(k)) \to P(1) & \equiv \\ 1 \in \mathbb{N} \to (0 < 1 \to P(0)) \to P(1) & \equiv \\ 1 \in \mathbb{N} \to (\top \to P(0)) \to P(1) & \equiv \\ 1 \in \mathbb{N} \to (\neg\top \lor P(0)) \to P(1) & \equiv \\ 1 \in \mathbb{N} \to (\bot \lor P(0)) \to P(1) & \equiv \\ 1 \in \mathbb{N} \to P(0) \to P(1) & \end{array}$$

que é equivalente a $P(0) \to P(1)$. Como temos que $P(0)$ (fórmula equivalente para $n = 0$), podemos concluir $P(1)$, usando eliminação da implicação. Repetindo esse mesmo raciocínio para $n = 2$, obtemos a fórmula

$$P(0) \land P(1) \to P(2)$$

que pode ser usada para concluir $P(2)$. Repetindo esse processo, temos que a propriedade P será verdadeira para qualquer $n \in \mathbb{N}$. Assim, temos que

$$\forall n.n \in \mathbb{N} \to (\forall k.k \in \mathbb{N} \land k < n \to P(k)) \to P(n)$$

realmente é equivalente a $\forall n.n \in \mathbb{N} \to P(n)$. Agora que justificamos a técnica de indução forte, a utilizaremos para demonstrar o teorema 42.

[1] Se você não lembra deste argumento, sugiro ler a seção anterior.

Demonstração. Suponha $n \in \mathbb{N}$ arbitrário e que para todo $k \in \mathbb{N}$, $k < n$, k é primo ou produto de primos. Evidentemente, se n é primo o resultado é imediato. Portanto, suponha que n não é primo. Logo, existem $a, b \in \mathbb{N}$ tais que $1 < a, b < n$ e $n = ab$. Como $a < n$ e $b < n$, pela hipótese de indução, temos que estes são primos ou produto de primos. Considere os seguintes casos:

1. a e b são primos. Logo, $n = a.b$ é um produto de primos.

2. a e b não são primos[2]. Logo, n = $a.b$ é um produto de primos.

Como os casos cobrem todas as possibilidades, temos que n é um produto de primos. □

Como um segundo exemplo da técnica de indução forte, apresentaremos outro resultado da teoria de números envolvendo o algoritmo de divisão de dois números inteiros.

Teorema 43. *Para todos $n, m \in \mathbb{N}$, se $m > 0$ então existem q e r tais que $n = mq + r$ e $r < m$.*

Note que q e r denotam o quociente e o resto da divisão, respectivamente. O teorema pode ser expresso pela seguinte fórmula:

$$\forall m.m \in \mathbb{N} \wedge m > 0 \rightarrow \forall n.n \in \mathbb{N} \rightarrow m > 0 \rightarrow \exists q.\exists r.q \in \mathbb{N} \wedge r \in \mathbb{N} \wedge n = mq + r \wedge r < m$$

Iniciamos esta demonstração supondo $m \in \mathbb{N}$ arbitrário, $m > 0$ e, na sequência, usamos indução forte para demonstrar

$$\forall n.n \in \mathbb{N} \rightarrow \exists q.\exists r.q \in \mathbb{N} \wedge r \in \mathbb{N} \wedge n = mq + r \wedge r < m$$

Em seguida, supomos $n \in \mathbb{N}$ arbitrário e que para todo $k < n$, $\exists q.\exists r.q \in \mathbb{N} \wedge r \in \mathbb{N} \wedge k = mq + r \wedge r < m$, o que, usando eliminação nos permite deduzir as hipóteses de que $k = mq + r$ e $r < m$. Para finalizar a demonstração, resta provar que

$$\exists q.\exists r.q \in \mathbb{N} \wedge r \in \mathbb{N} \wedge n = mq + r \wedge r < m$$

Se $n < m$, então basta fazer $q = 0$ e $r = n$ e o resultado é imediato. Para $n \geq m$, temos que encontrar valores de q e r tais que $n = mq + r$ e $r < m$. Note que como $n \geq m$, não podemos fazer que $r = n$. É óbvio que neste caso deveremos usar a hipótese de indução, mas para isso, devemos encontrar um valor de $k < n$ e a partir deste encontrar q e r. Qual será esse valor de k? Se nos atentarmos ao fato de que a divisão, $n \div m$, consiste em subtrair m de n sucessivamente, um possível valor para k é $n - m$, que será menor que n, visto que $m > 0$. Usando este valor o resultado desejado é quase imediato, como pode ser visto na demonstração a seguir.

Demonstração. Suponha $m \in \mathbb{N}$ arbitrário tal que $m > 0$. Suponha $n \in \mathbb{N}$ arbitrário e que para todo $k < n$ temos que existem q' e r' tais que $k = q'm + r'$ e $r' < m$. Considere os casos:

1. $n < m$. Seja $q = 0$ e $r = n$. Com isso, temos que $n = q.m + r$ e $r < m$, conforme requerido.

[2]Isto é, pela hipótese de indução algum dos dois (ou ambos) são um produto de primos

2. $n \geq m$. Seja $k = n - m < n$. Como $n \geq m$, temos que k é um número natural. Pela hipótese de indução, existem q' e r' tais que $k = mq' + r'$ e $r' < m$. Então, $n - m = mq' + r'$ e, portanto, $n = mq' + r' + m = m(q' + 1) + r'$. Assim, sejam $q = q' + 1$ e $r = r'$. Então, temos que $n = mq + r$ e $r < m$, conforme requerido.

□

Como um próximo exemplo, provaremos uma propriedade possuída por todo subconjunto não vazio de números naturais.

Teorema 44 (Princípio da boa ordenação)**.** *Todo conjunto não vazio de números naturais possui um elemento mínimo.*

Representamos este teorema é representado pela seguinte fórmula

$$\forall S.S \subseteq \mathbb{N} \to S \neq \emptyset \to S \text{ possui um elemento mínimo}$$

Aparentemente, este teorema não pode ser demonstrado por indução forte, uma vez que este não possui a estrutura

$$\forall n.n \in \mathbb{N} \to P(n)$$

Porém, se supormos $S \subseteq \mathbb{N}$ arbitrário, e representarmos

$$S \neq \emptyset \to S \text{ possui um elemento mínimo}$$

pela contrapositiva, temos a seguinte implicação:

$$S \text{ não possui um elemento mínimo} \to S = \emptyset$$

Supondo que S não possui mínimo, temos que provar que $S = \emptyset$, que é equivalente a dizer que $\forall n.n \in \mathbb{N} \to n \notin S$, o que é uma fórmula que pode ser demonstrada por indução, conforme apresentado na demonstração abaixo.

Demonstração. Suponha $S \subseteq \mathbb{N}$ arbitrário. Suponha que S não possui um elemento mínimo. Suponha $n \in \mathbb{N}$ arbitrário e que para todo $k < n$, $k \notin S$. Se $n \in S$, temos que S possui um elemento mínimo, o que contraria a suposição de que S não possui mínimo. Logo, se $S \neq \emptyset$, S possui um elemento mínimo. □

Agora, utilizaremos o princípio da boa ordenação para demonstrar mais um fato sobre números.

Teorema 45. $\sqrt{2}$ *é um número irracional.*

Lembre-se que dizemos que um número n é racional se existem p e q tais que $n = \frac{p}{q}$. Para mostrar que $\sqrt{2}$ é irracional, devemos supor que existem p, q tais que $\sqrt{2} = \frac{p}{q}$ e obter uma contradição a partir deste fato. É fácil ver que a partir de $\sqrt{2} = \frac{p}{q}$ podemos deduzir que $p^2 = 2q^2$ e, portanto, p^2 é par. Porém, se n^2 é par, temos que n é par, logo temos que a fração $\frac{p}{q}$ poderia ser simplificada. Aparentemente, este raciocínio não levaria a lugar algum. Porém, ao usarmos esta idéia em conjunto com o princípio da boa ordenação, obtemos a contradição desejada.

Demonstração. Suponha que $\sqrt{2}$ é número racional. Logo, existem p, q tais que $\sqrt{2} = \frac{p}{q}$. Seja Q o seguinte conjunto:

$$Q = \{q \in \mathbb{N}^+ \mid \exists p.p \in \mathbb{N}^+ \wedge \frac{p}{q} = \sqrt{2}\}$$

Porém se $\sqrt{2}$ é racional, Q não é vazio e, portanto, pelo princípio da boa ordenação (teorema 44), Q possuirá um elemento mínimo. Seja q o mínimo de Q. Então, podemos escolher $p \in \mathbb{N}^+$ tal que $\frac{p}{q} = \sqrt{2}$. Assim, temos que p^2 e p são pares. Logo, existe x tal que $p = 2x$. Substituindo $p = 2x$ em $p^2 = 2q^2$, temos que $4x^2 = 2q^2$ e, portanto, $q^2 = 2x^2$ e, portanto q^2 e q são pares. Logo, $\sqrt{2} = \frac{x}{y}$, em que $q = 2y$. Logo, $y \in Q$. Porém, como $y < q$, temos que este fato contradiz a suposição de que q é o mínimo de Q. Logo, $\sqrt{2}$ é irracional. □

9.3.1 Exercícios

1. Prove que $\sqrt{3}$ é irracional.

9.4 Paradoxos e Indução Matemática

A técnica de indução matemática é muito útil para demonstrar propriedades sobre números naturais. Porém, esta pode também ser usada para "provar" paradoxos, como o seguinte teorema:

Teorema 46. *Todos os cavalos possuem a mesma cor.*

Demonstração. A prova será por indução sobre o número de cavalos.

> Caso base: Para $n = 1$, temos que todos os cavalos do conjunto contendo $n = 1$ cavalos possuem a mesma cor.
>
> Passo indutivo: Suponha n arbitrário e que todos os n cavalos, $C_1, ..., C_n$ possuem a mesma cor. Para mostrar que todos os cavalos $C_1, ..., C_{n+1}$ possuem a mesma cor, considere os seguintes conjuntos A e B ambos contendo n cavalos:
>
> $$A = \{C_1, ..., C_n\}$$
> $$B = \{C_2, ..., C_n\}$$
>
> Como $|A| = |B| = n$, temos que todos os cavalos de A possuem a mesma cor x e todos os cavalos de B possuem a mesma cor y. Porém, como $C_2 \in A$ e $C_2 \in B$, temos que as cores x e y são iguais. Portanto, todos os cavalos possuem a mesma cor.

□

Evidentemente, o teorema anterior possui uma falha, pois existem cavalos das mais variadas cores. A falha deste teorema é que este não é válido para conjuntos contendo 2 cavalos. Note que se considerarmos um conjunto possuindo apenas os cavalos a e b, quando dividirmos este em dois conjuntos A e B, teremos $A = \{a\}$ e $B = \{b\}$, não possuindo, portanto, uma interseção.

Outro exemplo de uma falsa prova por indução é a seguinte:

Teorema 47. *Todo número natural é igual a* 0.

Demonstração. Suponha $n \in \mathbb{N}$ arbitrário e que para todo $k < n$, $k = 0$. Considere os casos:

Caso $n = 0$. Neste caso, o resultado é imediato.

Caso $n \neq 0$. Logo, existe m tal que $n = m + 1$. Pela hipótese de indução, todo $k < n$ é igual a 0, logo, $m = 0$ e $1 = 0$. Assim, temos que $n = m + 1 = 0 + 0 = 0$, conforme requerido.

□

Assim, como o exemplo anterior, este teorema é evidentemente falso. Note que este falha para $n = 1$, pois o único valor de $k < 1$ é zero e, portanto, não podemos concluir que $1 = 0$ como usado na "prova" anterior.

9.5 Notas Bibliográficas

Indução matemática é tema de todo livro de matemática discreta. Os exemplos e definições deste capítulo foram obtidos de [9].

10

Recursividade

Recursive. adj. See
RECURSIVE.

Stan Kelly-Bootie — The Devil's
DP Dictionary

10.1 Motivação

Tanto em matemática, quanto na ciência da computação, diversas operações são definidas recursivamente, isto é, alguns valores iniciais para esta operação são dados e os demais são obtidos aplicando-se uma ou mais regras sucessivamente. Um exemplo de função recursiva é a definição do fatorial, apresentada abaixo:

$$\begin{aligned} 0! &= 1 \\ n! &= n \times (n-1)! \end{aligned}$$

Como valor inicial, temos que o fatorial de 0 é 1 e, demais valores são obtidos pela segunda equação da definição.

De certa forma, provas por indução possuem uma estrutura similar a definições recursivas: apresenta-se provas de fatos elementares (casos base) e usa-se uma regra (passo indutivo) para mostrar que o fato em questão é válido para elementos diferentes dos considerados nos casos base. Neste capítulo, veremos como a indução é utilizada para demonstrar propriedades sobre definições recursivas.

10.2 Funções Recursivas

Existem diversas maneiras de se definir funções. Podemos definir uma função usando uma expressão que caracteriza a relação entre o domíno e sua imagem (método usualmente utilizado na matemática). Outra maneira de se definir uma função é através do uso de composição, que permite a definição de funções utilizando definições prévias. Esta forma de definir funções é o mais próximo do que idealmente deve ser feito em computação, visto que fornece o mais elevado nível de abstração, o de composição de interfaces. Existe, ainda uma terceira forma

de se definir uma função: utilizando recursão. Como um primeiro exemplo, considere a seguinte função $f : \mathbb{N} \to \mathbb{N}$ definida como

$$\begin{cases} f(0) &= 1 \\ f(n) &= 2n + f(n-1) \end{cases}$$

Note que esta definição especifica um valor inicial para f, $f(0) = 1$ e os demais valores são obtidos a partir de valores "anteriores" desta função. Como exemplo, considere o cálculo de $f(5)$, apresentado abaixo:

$$\begin{array}{ll} f(5) & = \\ 2.5 + f(4) & = \\ 10 + (2.4 + f(3)) & = \\ 10 + (8 + (2.3 + f(2))) & = \\ 10 + (8 + (6 + 2.2 + f(1))) & = \\ 10 + (8 + (6 + (4 + (2.1 + f(0))))) & = \\ 10 + (8 + (6 + (4 + (2 + 1)))) & = \\ 31 & \end{array}$$

Apesar de simples compreensão, o uso de funções recursivas possui o inconveniente de que o cálculo desta para valores elevados do domínio pode consumir muito tempo. Considere calcular $f(2000)$. Este cálculo ocasionaria 2000 chamadas recursivas. Porém, muitas vezes, podemos encontrar uma função g, equivalente a f, sem recursividade. Existem diversas técnicas para solucionar este tipo de problema e apresentaremos a mais simples destas baseada em indução matemática. Inicialmente, montamos uma pequena tabela de valores para f:

n	$f(n)$
0	1
1	3
2	7
3	13
4	21
5	31
6	43

Após pensar um pouco, podemos conjecturar que a função

$$g(n) = n(n+1) + 1$$

é equivalente a f, uma vez que esta possui os mesmos valores que f, conforme tabela abaixo:

n	$f(n)$	$g(n)$
0	1	1
1	3	3
2	7	7
3	13	13
4	21	21
5	31	31
6	43	43

Porém, somente construir e verificar esta tabela para alguns valores não é suficiente para mostrar que $f(n) = n(n+1) + 1$. Para isso, devemos provar que:

$$\forall n.n \in \mathbb{N} \to f(n) = n(n+1) + 1$$

que pode ser provado por indução matemática, conforme apresentado no teorema seguinte.

Teorema 48. *Seja $f(n)$ uma função definida como:*

$$\begin{cases} f(0) &= 1 \\ f(n) &= 2n + f(n-1) \end{cases}$$

então $f(n) = n(n+1) + 1$.

Demonstração.

Caso base ($n = 0$): Temos que $f(0) = 1 = 0(0+1)+1$, conforme requerido.

Passo indutivo: Suponha $n \in \mathbb{N}$ arbitrário e que $f(n) = n(n+1) + 1$. Temos:

$$\begin{array}{lll} f(n+1) & = & \\ 2(n+1) + f(n) & = & \text{pela definição de } f(n) \\ 2(n+1) + n(n+1) + 1 & = & \text{pela hipótese de indução} \\ (n+1)[(n+1)+1] + 1 & & \end{array}$$

Logo, $f(n+1) = (n+1)[(n+1)+1] + 1$ conforme requerido.

□

De maneira geral, podemos obter uma fórmula fechada (isto é, sem recursividade) para uma função recursiva $f(n)$ usando os seguintes passos:

1. Construir uma tabela contendo alguns valores da função $f(n)$.
2. "Adivinhar", a partir da tabela construída no passo anterior, qual função não recursiva produz os mesmos resultados para os valores da tabela.
3. Provar, usando indução matemática, que a fórmula fechada encontrada é realmente equivalente a função em questão.

A seguir, mostraremos mais exemplo desta técnica encontrando uma fórmula fechada para a seguinte função recursiva.

$$\begin{cases} f(0) &= 0 \\ f(n) &= 2f(n-1) + 1 \end{cases}$$

Inicialmente, construiremos uma tabela contendo alguns valores de $f(n)$:

n	$f(n)$
0	0
1	1
2	3
3	7
4	15
5	31
6	63

Se observarmos os valores da tabela, podemos perceber que estes são próximos de potências perfeitas de 2, logo, podemos conjecturar que a fórmula fechada para $f(n)$ é $2^n - 1$. Constataremos este fato provando por indução.

Teorema 49. *Seja $f(n)$ a função definida como*

$$\begin{cases} f(0) &= 0 \\ f(n) &= 2f(n-1)+1 \end{cases}$$

então $f(n) = 2^n - 1$.

Demonstração.

Caso base: Para $n = 0$, temos $f(0) = 0 = 1 - 1 = 2^0 - 1$.

Passo indutivo: Suponha $n \in \mathbb{N}$ arbitrário e que $f(n) = 2^n - 1$. Temos que:

$$\begin{array}{lll} f(n+1) & = & \\ 2f(n)+1 & = & \\ 2(2^n-1)+1 & = & \{\text{pela hipótese de indução}\} \\ 2^{n+1}-2+1 & = & \\ 2^{n+1}-1 & & \end{array}$$

Logo, $f(n+1) = 2^{n+1} - 1$.

□

10.2.1 Conjunto Potência, Recursivamente

No capítulo 5, apresentamos a definição do conjunto potência (ou conjunto das partes) de um conjunto A:

$$\mathcal{P}(A) = \{X \mid X \subseteq A\}$$

É fácil mostrar que $|\mathcal{P}(A)| = 2^n$ se $|A| = n$, usando o princípio multiplicativo (veja no capítulo 6).

Porém, como provar este resultado usando indução? Pode-se argumentar que basta utilizar indução sobre o tamanho do conjunto. Logo, no caso base, para $n = 0$, consideramos o conjunto vazio e obtemos $\mathcal{P}(\emptyset) = \{\emptyset\}$.

No caso indutivo, devemos considerar o cálculo do conjunto potência de um conjunto A com pelo menos um elemento. Isto é:

$$A = B \cup \{a\} \;\; a \notin B$$

Essas observações levam a seguinte função recursiva que, a partir de um conjunto qualquer, produz o conjunto potência deste.

$$\begin{cases} \mathcal{P}(\emptyset) &= \{\emptyset\} \\ \mathcal{P}(B \cup \{a\}) &= \mathcal{P}(B) \cup \{X \cup \{a\} \mid X \in \mathcal{P}(B)\} \;\; \text{em que } a \notin B \end{cases}$$

Observe que o cálculo do conjunto potência inclui o elemento a em cada um dos subconjuntos de B. Evidentemente, como $\mathcal{P}(B)$ e $\{X \cup \{a\} \mid X \in \mathcal{P}(B)\}$ são disjuntos e cada um destes possui 2^n elementos (pela hipótese de indução), temos que $|\mathcal{P}(A)| = 2^{n+1}$. A demonstração deste fato é apresentada a seguir.

Teorema 50. *Para todo A, se $|A| = n$ então $|\mathcal{P}(A)| = 2^n$.*

Demonstração.

Caso base ($n = 0$): Neste caso, temos que $A = \emptyset$. Logo, $|P(\emptyset)| = |\{\emptyset\}| = 1 = 2^0$, conforme requerido.

Passo indutivo: Suponha $n \in \mathbb{N}$ arbitrário e que $|B| = n$ e $|\mathcal{P}(B)| = 2^n$. Suponha a arbitrário tal que $a \notin B$ e que $A = B \cup a$. Seja $X = \{Y \cup \{a\} \mid Y \in \mathcal{P}(B)\}$. É óbvio que $|X| = 2^n$. Como $\mathcal{P}(A) = \mathcal{P}(B) \cup X$, temos que $|\mathcal{P}(A)| = |\mathcal{P}(B)| + |X| = 2^n + 2^n = 2^{n+1}$.

□

10.2.2 A Sequência de Fibonacci

A sequência de Fibonacci é uma sequência de números inteiros em que os dois primeiros números são 0 e 1 e os demais são obtidos somando os dois termos anteriores, isto é:

$$0, 1, 1, 2, 3, 5, 8, 13, 21, 34, 55, 89, 144, 233, 377, \ldots$$

Evidentemente podemos representar o n-ésimo termo desta sequência pela seguinte função recursiva:

$$\begin{cases} F(0) &= 0 \\ F(1) &= 1 \\ F(n) &= F(n-1) + F(n-2) \end{cases}$$

Como exemplo de uso desta função, considere o seguinte cálculo de $F(5)$:

$$\begin{aligned}
&F(5) && = \\
&\underbrace{F(4) + F(3)}_{F(5)} && = \\
&\underbrace{F(3) + F(2)}_{F(4)} + \underbrace{F(2) + F(1)}_{F(3)} && = \\
&\underbrace{F(2) + F(1)}_{F(3)} + \underbrace{F(1) + F(0)}_{F(2)} + \underbrace{F(1) + F(0)}_{F(2)} + 1 && = \\
&\underbrace{F(1) + F(0)}_{F(2)} + 1 + 1 + 0 + 1 + 0 + 1 && = \\
&1 + 0 + 1 + 1 + 0 + 1 + 0 + 1 && = \\
&5
\end{aligned}$$

Evidentemente, que um algoritmo baseado nesta definição para calcular $F(n)$ será extremamente ineficiente. Desta forma, devemos procurar uma fórmula fechada para a sequência de Fibonacci de maneira que possamos calcular um elemento desta sequência sem usar recursão ou algum tipo de repetição.

Apesar de existir uma técnica para encontrar fórmulas fechadas para funções recursivas como a que define a sequência de Fibonacci, o uso desta foge ao escopo deste texto. Ao invés disso, vamos apresentar a fórmula fechada para $F(n)$ e provar que esta realmente corresponde a definição recursiva apresentada.

Teorema 51. *Seja $F(n)$ o n-ésimo termo da sequência de Fibonacci. Então,*

$$F(n) = \frac{\left(\frac{1+\sqrt{5}}{2}\right)^n - \left(\frac{1-\sqrt{5}}{2}\right)^n}{\sqrt{5}}$$

Demonstração. Suponha $n \in \mathbb{N}$ arbitrário. Suponha que para todo $k \in \mathbb{N}$, $k < n$, temos que:

$$F(k) = \frac{\left(\frac{1+\sqrt{5}}{2}\right)^k - \left(\frac{1-\sqrt{5}}{2}\right)^k}{\sqrt{5}}$$

Considere os seguintes casos:

1. Caso $n = 0$: Temos que:

$$\begin{array}{ll} F(0) & = \\ 0 & = \\ \frac{0}{\sqrt{5}} & = \\ \frac{1-1}{\sqrt{5}} & = \\ \frac{\left(\frac{1+\sqrt{5}}{2}\right)^0 - \left(\frac{1-\sqrt{5}}{2}\right)^0}{\sqrt{5}} & \end{array}$$

   conforme requerido.

2. Caso $n = 1$: Temos que:

$$\begin{array}{ll} F(1) & = \\ 1 & = \\ \frac{\sqrt{5}}{\sqrt{5}} & = \\ \frac{\frac{2\sqrt{5}}{2}}{\sqrt{5}} & = \\ \frac{\frac{\sqrt{5}+\sqrt{5}+1-1}{2}}{\sqrt{5}} & = \\ \frac{\left(\frac{1+\sqrt{5}}{2}\right)^1 - \left(\frac{1-\sqrt{5}}{2}\right)^1}{\sqrt{5}} & \end{array}$$

   conforme requerido.

3. Caso $n \geq 2$: Temos que:

$$
\begin{array}{lll}
F(n) & = & \\
F(n-1) + F(n-2) & = & \{\text{pela def. de } F(n)\} \\
\dfrac{\left(\dfrac{1+\sqrt{5}}{2}\right)^{n-1} - \left(\dfrac{1-\sqrt{5}}{2}\right)^{n-1}}{\sqrt{5}} + \dfrac{\left(\dfrac{1+\sqrt{5}}{2}\right)^{n-2} - \left(\dfrac{1-\sqrt{5}}{2}\right)^{n-2}}{\sqrt{5}} & = & \\
\dfrac{\left(\dfrac{1+\sqrt{5}}{2}\right)^{n-2}\left(1+\dfrac{1+\sqrt{5}}{2}\right) - \left(\dfrac{1-\sqrt{5}}{2}\right)^{n-2}\left(1+\dfrac{1-\sqrt{5}}{2}\right)}{\sqrt{5}} & = & \\
\dfrac{\left(\dfrac{1+\sqrt{5}}{2}\right)^{n-2}\left(\dfrac{1+\sqrt{5}}{2}\right)^{2} - \left(\dfrac{1-\sqrt{5}}{2}\right)^{n-2}\left(\dfrac{1-\sqrt{5}}{2}\right)^{2}}{\sqrt{5}} & = & \\
\dfrac{\left(\dfrac{1+\sqrt{5}}{2}\right)^{n} - \left(\dfrac{1-\sqrt{5}}{2}\right)^{n}}{\sqrt{5}} &&
\end{array}
$$

conforme requerido. □

Note que nesta prova usamos o fato de que os números $\frac{1+\sqrt{5}}{2}$ e $\frac{1-\sqrt{5}}{2}$ são soluções da seguinte equação: $x^2 = x + 1$, isto é, são números que se somarmos um a eles, produziremos este número elevado a segunda potência. Além disso, perceba que esta prova só é possível utilizando indução forte, pois precisamos que a hipótese seja válida não apenas para o antecessor de n, $n-1$, mas também para $n-2$. É útil o leitor tentar provar este teorema usando indução convencional e perceber onde não é possível continuar com a demonstração.

10.3 Problemas Recursivos

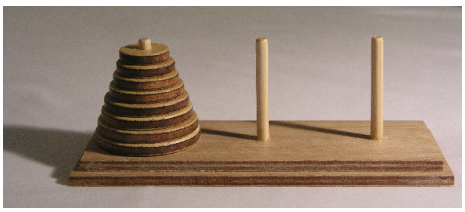
A seguir apresentamos alguns problemas clássicos e como estes podem ser modelados utilizando funções recursivas. Além dessa modelagem, apresentaremos como obter uma fórmula fechada equivalente a função apresentada.

10.3.1 As Torres de Hanói

As torres de Hanói é um quebra-cabeça inventado por um matemático francês, Édouard Lucas em 1833. Este quebra-cabeça consiste de uma torre contendo uma quantidade $n \in \mathbb{N}$ de discos, inicialmente empilhados em ordem decrescente de tamanho. A figura abaixo, apresenta a configuração inicial deste quebra-cabeças:

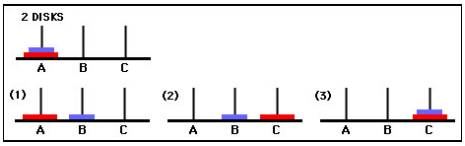
O objetivo deste jogo é transferir todos os discos de um pino para outro movendo apenas um disco de cada vez e nunca colocando um disco maior em cima de um menor.

Apesar de simples, não é óbvio que este quebra-cabeças possui solução. Após pensar um pouco, podemos perceber que este de fato, sempre possui solução. Porém, qual será a melhor? Isto é, é possível solucionar este problema fazendo o menor número de movimentos?

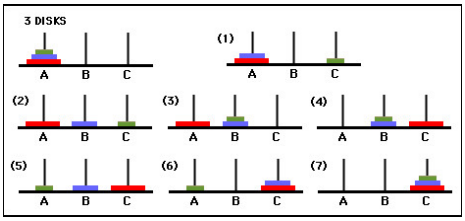


Para chegar a resposta para esta pergunta, devemos primeiro introduzir algumas notações. Chamaremos de $T(n)$ o número de movimentos necessários para solucionar o quebra cabeças contendo n discos.

É bastante fácil ver que $T(0) = 0$ e que $T(1) = 1$. A figura seguinte, mostra passo a passo, a solução para $n = 2$.



Para $n = 3$, temos:



Com isso, temos a seguinte tabela de valores iniciais de $T(n)$:

n	$T(n)$
0	0
1	1
2	3
3	7

Agora, que fizemos alguns experimentos com este problema, vamos mudar nossa perspectiva: ao invés de tentar pensar em como resolver este problema para casos específicos, vamos tentar generalizá-lo. Observando a figura para a solução com 3 discos, podemos perceber que o problema para $n = 3$ é resolvido da seguinte maneira:

- Mova $n - 1$ discos do pino A para o pino B.
- Mova o disco n do pino A para o pino C.
- Mova $n - 1$ discos do pino C para o pino C.

Como, para mover $n-1$ discos de um pino para outro, precisamos de $T(n-1)$ movimentos, no total precisamos de

$$T(n-1) + T(n-1) + 1 = 2T(n-1) + 1$$

para solucionar um quebra-cabeças de tamanho n. Assim, temos que o número mínimo de movimentos para a solução deste problema é dado pela seguinte função recursiva:

$$\begin{cases} T(0) &= 0 \\ T(n) &= 2T(n-1)+1 \end{cases}$$

Mas será que esta função reflete os resultados que obtivemos solucionando o problema? Vamos fazer os cálculos para $n = 3$:

$$\begin{array}{ll} T(3) & = \\ 2T(2)+1 & = \\ 2(2T(1)+1)+1 & = \\ 2(2(2T(0)+1)+1)+1 & = \\ 2(2(2.0+1)+1)+1 & = \\ 7 & \end{array}$$

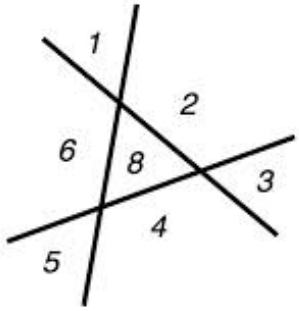
conforme requerido. Como vimos anteriormente, funções recursivas são usualmente ineficientes para o cálculo manual. Logo, é uma boa prática encontrarmos uma fórmula fechada para a função em questão. Porém, já encontramos esta fórmula no teorema 49.

10.3.2 O Problema da Pizzaria

Suponha que em um fim de semana você tenha ido a uma pizzaria que possuía a seguinte promoção:

> "O cliente que conseguir descobrir o número máximo de pedaços que pode ser obtido ao se fazer $n \in \mathbb{N}$ cortes em uma pizza, não a pagará."

Então, como pode-se comer uma pizza de graça? Novamente, vamos seguir a estratégia utilizada no exemplo anterior. Primeiro, vamos chamar de $T(n)$ o número de fatias obtidas após fazermos o n-ésimo corte. É bem fácil perceber que $T(0) = 1$, visto que se não fizermos nenhum corte, temos uma fatia (a pizza inteira). Usando um raciocínio parecido, temos que $T(1) = 2$, visto que ao fazermos um corte, iremos dividir a pizza em dois pedaços. Porém, quantos pedaços obtemos ao fazer o 3^o corte? A intuição nos diz que devemos obter $T(3) = 6$, porém, conforme mostrado na próxima figura, isso não é bem verdade...



Note que obtemos um número maior de pedaços fazendo com que o n-ésimo corte intercepte todos os cortes anteriores. Com isso, aumentamos o número total de fatias em n pedaços, isto é, $T(3) = 4 + 3 = 7$, em que $4 = T(2)$. Desta forma, podemos conjecturar que $T(n)$ é a seguinte função recursiva:

$$\begin{cases} T(0) &= 1 \\ T(n) &= T(n-1) + n \end{cases}$$

Note que ao calcularmos alguns valores de $T(n)$, podemos notar que este nada mais é que a soma dos n primeiros números naturais somados com 1, conforme expandido abaixo:

$$\begin{aligned} T(n) &= T(n-1) + n \\ &= (T(n-2) + (n-1)) + n \\ &= ((T(n-3) + (n-2)) + (n-1)) + n \\ &\vdots \\ &= T(0) + 1 + 2 + ... + (n-2) + (n-1) + n \\ &= 1 + 1 + 2 + ... + (n-2) + (n-1) + n \\ &= 1 + \sum_{k=1}^{n} k \end{aligned}$$

Pode-se mostrar por indução que $\sum_{k=1}^{n} = \frac{n(n+1)}{2}$. Logo, temos que $T(n)$ é dado por:

$$T(n) = \frac{n(n+1)}{2} + 1$$

Realmente esta fórmula corresponde a função $T(n)$, conforme provamos no teorema a seguir.

Teorema 52. *Seja $T(n)$ a função definida como:*

$$\begin{cases} T(0) &= 1 \\ T(n) &= T(n-1) + n \end{cases}$$

então, $T(n) = \frac{n(n+1)}{2} + 1$.

Demonstração.

Caso base ($n = 0$): Temos que $T(0) = \frac{0(0+1)}{2} + 1$, conforme requerido.

Passo indutivo: Suponha $n \in \mathbb{N}$ arbitrário e que $T(n) = \frac{n(n+1)}{2} + 1$. Temos que:

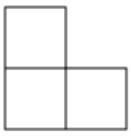
$$\begin{array}{lcl} T(n+1) & = & \\ T(n) + (n+1) & = & \{\text{pela def. de } T(n)\} \\ \dfrac{n(n+1)}{2} + 1 + (n+1) & = & \{\text{pela hipótese de indução}\} \\ \dfrac{n(n+1) + 2(n+1)}{2} + 1 & = & \\ \dfrac{(n+1)(n+2)}{2} + 1 & & \end{array}$$

conforme requerido.

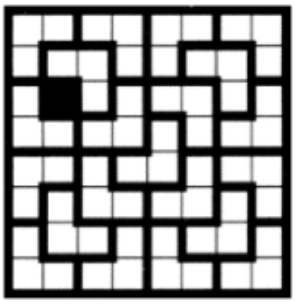
□

10.3.3 Preenchendo um Tabuleiro de Xadrez

Considere seguinte quebra-cabeça: preencher um tabuleiro $2^n \times 2^n$, $n \in \mathbb{N}$, $n \geq 1$, com peças em formato de "L" como a apresentada abaixo:

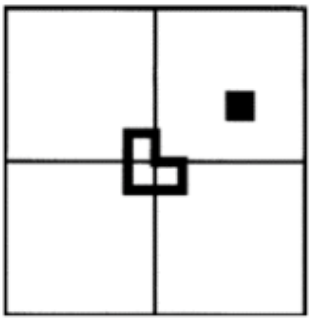


de maneira que apenas uma posição do tabuleiro não seja ocupada por estas peças. Abaixo apresentamos a solução deste quebra-cabeças para um tabuleiro de $2^3 \times 2^3$:



em que a posição não ocupada por peças é a que está em "preto". A questão é como resolver este quebra-cabeças para um valor qualquer de $n \geq 1$?

É fácil ver pela figura abaixo que o quebra-cabeça é obviamente solúvel para $n = 1$.



Mas, como solucionar este quebra-cabeças para um $n > 1$? O ponto principal para solucionar este problema para $n > 1$ é observar que um tabuleiro $2^{n+1} \times 2^{n+1}$ é formado por 4 tabuleiros de $2^n \times 2^n$. Logo, podemos resolver o quebra-cabeça para um tabuleiro de $2^{n+1} \times 2^{n+1}$ a partir das 4 soluções para tabuleiros de $2^n \times 2^n$. A chave para combinar as soluções de cada um dos "pedaços" dos quebra cabeças é deixar a posição não preenchida de cada um destes na extremidade em que esta faz junção com os outros tabuleiros de mesmo tamanho. Como são 4 tabuleiros de tamanho $2^n \times 2^n$, haverão 4 posições não preenchidas, permitindo assim o encaixe de mais uma peça em L, completando a solução do quebra-cabeça.

A descrição informal acima apresentada, mostra como resolver este quebra-cabeça, isto é, fornece um algoritmo recursivo para o problema em questão. Além disso, esta mesma descrição é exatamente a estrutura de uma prova por indução que mostra que este problema é solúvel para todo $n \geq 1$. Isto não é uma mera coincidência. Normalmente, provas por indução possuem a mesma estrutura de algoritmos recursivos.

Teorema 53. *Para todo $n \geq 1$, temos que todo tabuleiro de $2^n \times 2^n$ pode ser preenchido por peças em forma de "L" de maneira que somente uma posição do tabuleiro não seja ocupada por uma destas peças.*

Demonstração. A prova será por indução sobre n.

1. Caso base ($n = 1$). Imediato. Basta ocupar o tabuleiro com uma peça em forma de "L".

2. Passo indutivo. Suponha $n \in \mathbb{N}$ arbitrário e que todo tabuleiro de $2^n \times 2^n$ possa ser preenchido de forma que apenas uma posição não esteja ocupada. Como um tabuleiro de $2^{n+1} \times 2^{n+1}$ é formado por 4 tabuleiros de $2^n \times 2^n$, pela hipótese de indução, estes 4 tabuleiros podem ser preenchidos de maneira que uma posição destes não seja preenchida. Deixando a posição vazia destes tabuleiros de $2^n \times 2^n$ no ponto de junção destes tabuleiros, podemos formar a solução para o tabuleiro de $2^{n+1} \times 2^{n+1}$ acrescentando uma peça, deixando apenas uma posição livre, completando assim o quebra-cabeça.

□

10.4 Exercícios

1. Encontre uma fórmula fechada (sem recursividade) equivalente a cada uma das funções recursivas a seguir e prove que a fórmula encontrada é equivalente a função em questão.

   (a) $\begin{cases} T(0) = 0 \\ T(n) = 2T(n-1) + n \end{cases}$

   (b) $\begin{cases} T(0) = 2 \\ T(n) = (T(n-1))^2 \end{cases}$

   (c) $\begin{cases} T(0) = 2 \\ T(n) = 2T(n-1) + n \end{cases}$

   (d) $\begin{cases} T(1) = 1 \\ T(n) = \frac{T(n-1)}{1+T(n-1)} \end{cases}$

   (e) $\begin{cases} T(1) = \frac{1}{4} \\ T(2) = \frac{1}{8} \\ T(n) = \frac{T(n-1)T(n-2)}{2T(n-2)-T(n-1)} \end{cases}$

2. Seja $F(n)$ o n-ésimo termo da sequência de Fibonacci, definida como:

$$\begin{cases} F(0) &= 0 \\ F(1) &= 1 \\ F(n) &= F(n-1) + F(n-2) \end{cases}$$

   Prove os seguintes fatos sobre a sequência de Fibonacci.

   (a) $\sum_{i=0}^{n} F(i) = F(n+2) - 1$

   (b) $\sum_{i=0}^{n} F(2i+1) = F(2n+2)$

   (c) $\sum_{i=0}^{n} (F(i))^2 = F(n)F(n+1)$

   (d) Prove que para todo $n \in \mathbb{N}$, $F(n) < 2^n$.

3. Seja A um conjunto. Representamos por $\mathcal{P}_2(A)$ o conjunto de todos os subconjuntos de A que contém 2 elementos. Prove que para todo conjunto A, se $|A| = n$, então $|\mathcal{P}_2(A)| = \frac{n(n-1)}{2}$.

10.5 Notas Bibliográficas

Recursividade e sua relação com a indução matemática é um tema presente em todo texto de matemática discreta. O foco do capítulo atual foi o uso de indução matemática para demonstrar fórmulas fechadas equivalentes a funções recursivas. Alguns dos exemplos deste capítulo foram retirados de [2].

11

Indução Estrutural

> Correctness is clearly the prime quality. If a system does not do what it is supposed to do, then everything else about it matters little.
>
> Berthrand Meyer, Cientista da Computação

11.1 Motivação

Como vimos nos dois capítulos anteriores (capítulos 9 e 10), a indução matemática é uma técnica de demonstração aplicável em diversas situações. Nestes capítulos, apresentamos um enfoque sobre a indução matemática que essencialmente abordou problemas matemáticos, porém, esta técnica é aplicável também a provas de propriedades sobre estruturas de dados recursivas e algoritmos sobre estas. A este tipo de demonstração de indução, damos o nome de indução estrutural.

O objetivo deste capítulo é o estudo da indução estrutural para demonstração de correção sobre alguns algoritmos sobre estruturas de dados simples como listas. Para evitar problemas relativos à utilização de atribuição de variáveis, e aspectos específicos de linguagens de programação, representaremos estruturas de dados, como definições sintáticas e algoritmos como funções, de maneira similar ao que fizemos no capítulo 1.

11.2 Indução Estrutural

Conforme apresentado no capítulo 1, definições sintáticas de conjuntos de termos devem possuir elementos iniciais (casos base) e, opcionalmente, formas de se construir termos mais complexos a partir de termos existentes (passo(s) indutivo(s)). De maneira simples, a técnica de indução estrutural pode ser resumida da seguinte maneira: Seja P a propriedade a ser demonstrada para todo termo

t pertencente a um conjunto $\mathcal{T}$. Para constatar que $P(t)$ é verdade basta mostrar que esta propriedade é verdadeira para cada um dos casos base e passos indutivos da definição do conjunto $\mathcal{T}$.

As próximas seções apresentarão a indução estrutural em exemplos concretos: números naturais na notação de Peano ($\mathcal{N}$) e listas.

11.2.1 Números Naturais na Notação de Peano

Conforme apresentado no capítulo 1, o conjunto $\mathcal{N}$, dos termos que representam números naturais na notação de Peano, pode ser definido pelas seguintes regras:

$$\begin{array}{l} zero \in \mathcal{N} \\ \text{se } n \in \mathcal{N} \text{ então } suc\, n \in \mathcal{N} \end{array}$$

Nesta notação, o número natural 3 é representado pelo termo $suc\,(suc\,(suc\,zero))$, isto é todo número natural ou é representado pelo termo *zero* ou por uma sequência de n *suc*'s que terminam com a constante *zero*.

Usando esta notação, podemos definir como funções recursivas operações sobre números naturais, como por exemplo, a adição:

$$\begin{array}{lcll} plus(zero, m) & = & m & (1) \\ plus(suc\, n, m) & = & suc(plus(n, m)) & (2) \end{array}$$

Numeramos as equações da definição de *plus* para referenciar uma equação específica quando necessário. Como um exemplo da utilização da função *plus*, considere a soma: 2+3, que é representada como $plus(suc(suc\,zero), suc(suc(suc\,zero)))$:

$$\begin{array}{lcl} plus(suc(suc\,zero), suc(suc(suc\,zero))) & \equiv & \\ suc(plus(suc\,zero, suc(suc(suc\,zero)))) & \equiv & \{\text{pela equação 2 de } plus\} \\ suc(suc(plus(zero, suc(suc(suc\,zero))))) & \equiv & \{\text{pela equação 2 de } plus\} \\ suc(suc(suc(suc(suc\,zero)))) & & \{\text{pela equação 1 de } plus\} \end{array}$$

Note que o processo de execução da função *plus* é completamente determinado por sua definição: se o primeiro parâmetro desta função é igual a *zero*, o seu resultado será o segundo parâmetro (m, na definição de *plus*). Porém, se o primeiro parâmetro não for igual a *zero*, necessariamente este deverá ser $suc\, n$, para algum $n \in \mathcal{N}$, e o resultado será o sucessor da chamada recursiva $plus(n, m)$. É útil que você faça mais algumas execuções da função *plus* até que você tenha compreendido completamente seu funcionamento.

De acordo com a definição da função *plus*, note que $\forall m.plus(zero, m) \equiv m$ (pela equação 1 de *plus*), porém não é imediato que $\forall n.plus(n, zero)$. Isto se deve que o termo $plus(zero, m)$ pode ser reduzido imediatamente a m, de acordo com a equação 1 de *plus*, enquanto $plus(n, zero)$ não, uma vez que não é possível determinar se n é ou não igual a *zero*.

Em lógica, dizemos que a expressão $plus(zero, m)$ é *igual por definição*[1] a m, uma vez que esta igualdade pode ser deduzida diretamente pela definição de *plus*, executando-a. Note que apesar de evidentemente verdadeira, a igualdade $plus(n, zero)$ não pode ser considerada igual por definição a n, visto que não existe uma única possibilidade de execução para esta expressão pois, n pode ser ou não igual a *zero*. Neste caso, se desejamos demonstrar tal igualdade,

[1] Tradução livre do termo: "definitionally equal to".

devemos prová-la usando indução. Antes disso, vamos apresentar a definição do princípio de indução estrutural para o conjunto $\mathcal{N}$.

Definição 78 (Indução sobre $\mathcal{N}$)**.** Seja P uma propriedade qualquer sobre elementos de $\mathcal{N}$. Podemos demonstrar que $\forall n.n \in \mathcal{N} \rightarrow P(n)$ usando a seguinte fórmula:

$$P(zero) \wedge \forall n.n \in \mathcal{N} \wedge P(n) \rightarrow P(suc\, n)$$

■

Note que esta definição é exatamente igual ao princípio de indução matemática que vimos no capítulo 9, a menos do uso do conjunto $\mathcal{N}$ ao invés de $\mathbb{N}$ e das constantes *zero* e *suc*.

A seguir, apresentamos a prova da propriedade $\forall n.n \in \mathcal{N} \rightarrow plus(n, zero) \equiv n$, usando indução estrutural.

Teorema 54. *Para todo $n \in \mathcal{N}$, $plus(n, zero) \equiv n$.*

Demonstração. Esta demonstração será por indução sobre n.

1. Caso base ($n = zero$). Neste caso, temos que

   $$\begin{array}{lll} plus(zero, zero) & \equiv & \\ zero & & \{\text{pela equação 1 de } plus\} \end{array}$$

   conforme requerido.

2. Passo indutivo ($n = suc\, n'$). Suponha $n' \in \mathcal{N}$ arbitrário e que $plus(n', zero) \equiv n'$. Temos que:

   $$\begin{array}{lll} plus(suc\, n', zero) & \equiv & \\ suc(plus(n', zero)) & \equiv & \{\text{pela equação 2 de } plus\} \\ suc\, n' & & \{\text{pela hipótese de indução}\} \end{array}$$

   conforme requerido.

□

Observe que no passo indutivo desta demonstração, consideramos que o primeiro parâmetro n é tal que $n = suc\, n'$. A hipótese de indução é obviamente definida para n', o antecessor de n.

Usualmente, provas por indução estrutural sobre funções devem realizar a indução sobre o parâmetro recursivo da definição da função. Como a função *plus* é definida recursivamente sobre seu 1^o parâmetro, provas sobre esta devem ser feitas utilizando indução sobre este. Como um segundo exemplo de demonstração por indução estrutural, considere demonstrar que a adição é uma operação associativa, isto é:

$$plus(n, plus(m, p)) \equiv plus(plus(n, m), p)$$

Essa propriedade é demonstrada no teorema seguinte.

Teorema 55 (*plus* é uma operação associativa)**.** *Para todo $n, m, p \in \mathcal{N}$, temos que $plus(n, plus(m, p)) \equiv plus(plus(n, m), p)$.*

Demonstração. Esta prova será por indução sobre n. Suponha $m, p \in \mathcal{N}$ arbitrários.

1. Caso base ($n = zero$). Temos que:

$$\begin{array}{ll} plus(zero, plus(m,p)) & \equiv \\ plus(m,p) & \{\text{pela equação 1 de } plus\} \end{array}$$

   conforme requerido.

2. Passo indutivo ($n = suc\, n'$): Suponha $n' \in \mathcal{N}$ arbitrário e que $plus(n', plus(m,p)) \equiv plus(plus(n', m), p)$. Temos que:

$$\begin{array}{lll} plus(suc\, n', plus(m,p)) & \equiv & \\ suc(plus(n', plus(m,p))) & \equiv & \{\text{pela equação 2 de } plus\} \\ suc(plus(plus(n', m), p)) & \equiv & \{\text{pela hipótese de indução}\} \\ plus(suc(plus(n', m)), p) & \equiv & \{\text{pela equação 2 de } plus\} \\ plus(plus(suc\, n', m), p) & & \{\text{pela equação 2 de } plus\} \end{array}$$

   conforme requerido.

□

11.2.2 Exercícios

1. Prove que a soma de números na notação de Peano é uma operação comutativa, isto é, prove que:

$$\forall n.n \in \mathcal{N} \rightarrow \forall m.m \in \mathcal{N} \rightarrow plus(n,m) \equiv plus(m,n)$$

2. Considere a seguinte definição alternativa da soma na notação de Peano.

$$\begin{array}{lcll} plus_{alt}(n, zero) & = & n & (1) \\ plus_{alt}(n, suc\, m) & = & suc(plus_{alt}(n,m)) & (2) \end{array}$$

   Prove que para quaisquer valores $n, m \in \mathcal{N}$, $plus_{alt}(n,m) \equiv plus(n,m)$.

3. Defina a função $mult(n,m)$ que realiza a multiplicação de números naturais na notação de Peano.

4. Prove que a função de multiplicação definida por você é uma operação comutativa.

11.2.3 Listas

Nesta seção, consideraremos algumas funções sobre listas e provas de propriedades sobre estas utilizando indução estrutural. No capítulo 1, apresentamos o conjunto de listas cujos elementos são de $\mathcal{T}$, $List\,\mathcal{T}$, como sendo os termos definidos recursivamente como:

$[\,] \in List\,\mathcal{T}$
se $t \in \mathcal{T}$ e $ts \in List\,\mathcal{T}$ então $t :: ts \in List\,\mathcal{T}$

Por questão de simplicidade, vamos considerar que os elementos de listas são valores booleanos, cuja definição apresentamos a seguir:

$$T \in \mathcal{B}$$
$$F \in \mathcal{B}$$

É importante notar que esta simplificação será feita apenas para fins de facilitar o entendimento e a escrita de exemplos. Todas as funções e suas respectivas propriedades são válidas para listas cujos elementos pertencem a um conjunto $\mathcal{T}$ qualquer. Desta forma, representaremos a lista que contém os elementos T e F, nesta ordem, como: $T :: F :: [\,]$. Note que o valor que representa uma lista vazia ($[\,]$) possui funcionalidade similar ao um ponteiro "nulo" em implementações de listas encadeadas em linguagens de programação como C/C++, a de indicar o final da lista em questão.

Como exemplos de funções sobre listas, considere, as funções para determinar o número de elementos ($length$) e concatenação de duas listas ($++$) apresentadas a seguir:

$$\begin{array}{lcll} length\ [\,] & = & 0 & (1) \\ length\ (t :: ts) & = & 1 + length\ ts & (2) \end{array}$$

$$\begin{array}{lcll} [\,] ++ ys & = & ys & (1) \\ (x :: xs) ++ ys & = & x :: (xs ++ ys) & (2) \end{array}$$

Novamente, numeramos as equações para futura referência. Antes de apresentarmos um primeiro exemplo de propriedade a ser demonstrada para listas, vamos definir o princípio de indução para listas.

Definição 79 (Indução Estrutural para $List\ \mathcal{T}$)**.** Seja $\mathcal{T}$ um conjunto qualquer. Seja P uma propriedade sobre o conjunto de listas finitas de elementos do conjunto $\mathcal{T}$, $List\ \mathcal{T}$. Então, podemos provar que $\forall t.t \in List\ \mathcal{T} \rightarrow P(t)$ usando a seguinte fórmula:

$$P([\,]) \wedge \forall x.x \in \mathcal{T} \rightarrow \forall xs.xs \in List\ \mathcal{T} \wedge P(xs) \rightarrow P(x :: xs)$$

■

Intuitivamente, podemos provar que uma propriedade é verdadeira para todas as listas finitas se formos capazes de provar que esta vale para a lista vazia e, além disso, provarmos que a propriedade continua sendo verdadeira se inserirmos um novo elemento em uma lista qualquer para a qual a propriedade em questão era válida.

Como um exemplo de demonstração por indução sobre listas, considere a seguinte propriedade que pode ser usada para caracterizar a correção de um algoritmo de concatenação de duas listas: o tamanho da concatenação de duas listas xs e ys é igual a soma dos tamanhos de cada uma destas listas. Mais formalmente, a propriedade em questão é:

$$\forall xs.xs \in List\,\mathcal{T} \rightarrow \forall ys.ys \in List\,\mathcal{T} \rightarrow length(xs\ ++\ ys) = length\,xs + length\,ys$$

Essa propriedade é facilmente demonstrada por indução sobre a primeira lista (xs). A indução será feita sobre a primeira lista devido ao fato de que a concatenação é definida recursivamente sobre a primeira lista fornecida como parâmetro.

Teorema 56. *Seja $\mathcal{T}$ um conjunto qualquer de termos. Então para todo $xs, ys \in List\,\mathcal{T}$, temos que $length(xs \texttt{ ++ } ys) = length\,xs + length\,ys$.*

Demonstração. A prova será por indução sobre xs. Suponha $ys \in List\,\mathcal{T}$ arbitrário.

1. Caso base ($xs = [\,]$). Temos que:

$$\begin{array}{lll} length([\,] \texttt{ ++ } ys) & \equiv & \\ length\,ys & \equiv & \{\text{pela equação 1 de ++}\} \\ 0 + length\,ys & \equiv & \\ length\,[\,] + length\,ys & & \{\text{pela equação 1 de ++}\} \end{array}$$

   conforme requerido.

2. Passo indutivo. Suponha $x \in \mathcal{T}$ arbitrário e que $length(xs \texttt{ ++ } ys) = length\,xs + length\,ys$. Temos que:

$$\begin{array}{lll} length((x :: xs) \texttt{ ++ } ys) & \equiv & \\ length(x :: (xs \texttt{ ++ } ys)) & \equiv & \{\text{pela equação 2 de ++}\} \\ 1 + length(xs \texttt{ ++ } ys) & \equiv & \{\text{pela equação 2 de } length\} \\ 1 + length\,xs + length\,ys & \equiv & \{\text{pela hipótese de indução}\} \\ length(x :: xs) + length\,ys & & \{\text{pela equação 2 de } length\} \end{array}$$

   conforme requerido.

□

Para o nosso próximo exemplo de uma propriedade sobre listas, considere a função *reverse*, que inverte uma lista fornecida como parâmetro.

$$\begin{array}{lcll} reverse\,[\,] & = & [\,] & (1) \\ reverse\,(x :: xs) & = & reverse\,xs \texttt{ ++ } (x :: [\,]) & (2) \end{array}$$

De maneira simples, *reverse* move o primeiro elemento da lista fornecida como parâmetro para o final do resultado de se inverter o restante desta lista. Note que só é possível inserir um elemento na primeira posição de uma lista. Se desejamos inserir um elemento ao final de uma lista, devemos concatená-lo ao final e não simplesmente inserí-lo. Por isso, definimos a função *reverse* em termos da operação de concatenação de duas listas.

Como exemplo do funcionamento da função *reverse*, considere a seguinte execução desta para a lista $T :: F :: T :: [\,]$, apresentada a seguir:

$$\begin{array}{lll} reverse(T :: F :: T :: [\,]) & \equiv & \\ reverse(F :: T :: [\,]) \texttt{ ++ } (T :: [\,]) & \equiv & \{\text{pela equação 2 de } reverse\} \\ (reverse(T :: [\,]) \texttt{ ++ } (F :: [\,])) \texttt{ ++ } (T :: [\,]) & \equiv & \{\text{pela equação 2 de } reverse\} \\ ((reverse\,[\,] \texttt{ ++ } (T :: [\,])) \texttt{ ++ } (F :: [\,])) \texttt{ ++ } (T :: [\,]) & \equiv & \{\text{pela equação 2 de } reverse\} \\ (([\,] \texttt{ ++ } (T :: [\,])) \texttt{ ++ } (F :: [\,])) \texttt{ ++ } (T :: [\,]) & \equiv & \{\text{pela equação 1 de } reverse\} \\ (((T :: [\,])) \texttt{ ++ } (F :: [\,])) \texttt{ ++ } (T :: [\,]) & \equiv & \{\text{pela equação 1 de ++}\} \\ (T :: ([\,] \texttt{ ++ } (F :: [\,]))) \texttt{ ++ } (T :: [\,]) & \equiv & \{\text{pela equação 2 de ++}\} \\ (T :: (F :: [\,])) \texttt{ ++ } (T :: [\,]) & \equiv & \{\text{pela equação 1 de ++}\} \\ T :: ((F :: [\,]) \texttt{ ++ } (T :: [\,])) & \equiv & \{\text{pela equação 2 de ++}\} \\ T :: (F :: ([\,] \texttt{ ++ } (T :: [\,]))) & \equiv & \{\text{pela equação 2 de ++}\} \\ T :: (F :: (T :: [\,])) & & \{\text{pela equação 1 de ++}\} \end{array}$$

Como exemplo de propriedade sobre a função *reverse*, apresentaremos como esta se relaciona com a operação de concatenação de listas.

Teorema 57. *Seja $\mathcal{T}$ um conjunto qualquer de termos. Então para todo $xs, ys \in List\,\mathcal{T}$, temos que $reverse(xs \texttt{ ++ } ys) \equiv reverse\,ys \texttt{ ++ } reverse\,xs$.*

Demonstração. A prova será por indução sobre xs. Suponha $ys \in List\mathcal{T}$ arbitrário.

1. Caso base ($xs = [\,]$). Temos que:

$$\begin{array}{lll} reverse([\,]{+}{+}ys) & \equiv & \\ reverse\,ys & \equiv & \{\text{pela equação 1 de ++}\} \\ reverse\,ys + +[\,] & & \{\text{pela equação 1 de ++}\} \end{array}$$

   conforme requerido.

2. Passo indutivo. Suponha $x \in \mathcal{T}$ arbitrário e que $reverse(xs{+}{+}ys) \equiv reverse\,ys{+}{+}reverse\,xs$. Temos que:

$$\begin{array}{lll} reverse((x :: xs){+}{+}ys) & \equiv & \\ reverse(x :: (xs{+}{+}ys)) & \equiv & \{\text{pela equação 2 de ++}\} \\ reverse(xs{+}{+}ys){+}{+}(x :: [\,]) & \equiv & \{\text{pela equação 2 de } reverse\} \\ (reverse\,ys{+}{+}reverse\,xs){+}{+}(x :: [\,]) & \equiv & \{\text{pela hipótese de indução}\} \\ reverse\,ys{+}{+}(reverse\,xs{+}{+}(x :: [\,])) & \equiv & \{\texttt{++} \text{ é associativo}\} \\ reverse\,ys{+}{+}reverse(x :: xs) & & \{\text{pela equação 2 de } reverse\} \end{array}$$

□

Note que nesta demonstração usamos, sem demonstrar, o fato de que a operação de concatenação de listas é associativa, isto é:

$$\forall xs.\forall ys.\forall zs.xs \in List\,\mathcal{T} \wedge ys \in List\,\mathcal{T} \wedge zs \in List\,\mathcal{T} \rightarrow xs{+}{+}(ys{+}{+}zs) \equiv (xs{+}{+}ys){+}{+}zs$$

Esta demonstração simples é deixada como exercício para o leitor.

11.2.4 Exercícios

1. Prove que a concatenação de listas é uma operação associativa.
2. Prove o seguinte teorema envolvendo as funções *length* e *reverse*: Para toda lista $xs \in List\,\mathcal{T}$, temos que $length(reverse\,xs) \equiv length\,xs$.
3. Considere a seguinte definição alternativa de uma função que inverte uma dada lista:

$$\begin{array}{lcll} reverse_{alt}\,xs & = & rev\,xs\,[\,] & (1) \\ & & & \\ rev\,[\,]\,ys & = & ys & (1) \\ rev\,(x :: xs)\,ys & = & rev\,xs\;\;(x :: ys) & (2) \end{array}$$

   (a) Mostre, passo a passo, a execução de $reverse_{alt}(T :: F :: F :: [\,])$.

   (b) Prove que para toda lista $xs \in List\,\mathcal{T}$, $reverse_{alt}xs \equiv reverse\,xs$, em que *reverse* é a primeira definição apresentada de reverse neste texto.

4. Prove que para toda lista $xs \in List\,\mathcal{T}$, $reverse(reverse\,xs) \equiv xs$.

11.2.5 Árvores Binárias

Encerraremos este capítulo apresentando demonstrações simples de propriedades sobre árvores binárias, um tipo de estrutura de dados muito importante na ciência da computação.

De maneira simples, uma árvore binária é vazia ou consiste de um nó que armazena um elemento e possui duas sub árvores (também binárias), denominadas sub árvore esquerda e direita, respectivamente. A seguir, apresentamos a definição formal do conjunto de árvores binárias.

Definição 80 (Árvores binárias)**.** Seja $\mathcal{T}$ um tipo qualquer. O conjunto $\mathit{Tree}\,\mathcal{T}$ de árvores cujos elementos são do tipo $\mathcal{T}$ é definida recursivamente (indutivamente) como:

$$\begin{array}{l} \mathit{Leaf} \in \mathit{Tree}\,\mathcal{T} \\ \text{se } l, r \in \mathit{Tree}\,\mathcal{T} \text{ e } x \in \mathcal{T} \text{ então } \mathit{Node}\,x\,l\,r \in \mathit{Tree}\,\mathcal{T} \end{array}$$

■

11.3 Predicados Definidos Indutivamente

11.4 Notas Bibliográficas

Apêndice A

GNU Free Documentation License

Preamble

The purpose of this License is to make a manual, textbook, or other functional and useful document "free" in the sense of freedom: to assure everyone the effective freedom to copy and redistribute it, with or without modifying it, either commercially or noncommercially. Secondarily, this License preserves for the author and publisher a way to get credit for their work, while not being considered responsible for modifications made by others.

This License is a kind of "copyleft", which means that derivative works of the document must themselves be free in the same sense. It complements the GNU General Public License, which is a copyleft license designed for free software.

We have designed this License in order to use it for manuals for free software, because free software needs free documentation: a free program should come with manuals providing the same freedoms that the software does. But this License is not limited to software manuals; it can be used for any textual work, regardless of subject matter or whether it is published as a printed book. We recommend this License principally for works whose purpose is instruction or reference.

1. APPLICABILITY AND DEFINITIONS

This License applies to any manual or other work, in any medium, that contains a notice placed by the copyright holder saying it can be distributed under the terms of this License. Such a notice grants a world-wide, royalty-free license, unlimited in duration, to use that work under the conditions stated

herein. The "**Document**", below, refers to any such manual or work. Any member of the public is a licensee, and is addressed as "**you**". You accept the license if you copy, modify or distribute the work in a way requiring permission under copyright law.

A "**Modified Version**" of the Document means any work containing the Document or a portion of it, either copied verbatim, or with modifications and/or translated into another language.

A "**Secondary Section**" is a named appendix or a front-matter section of the Document that deals exclusively with the relationship of the publishers or authors of the Document to the Document's overall subject (or to related matters) and contains nothing that could fall directly within that overall subject. (Thus, if the Document is in part a textbook of mathematics, a Secondary Section may not explain any mathematics.) The relationship could be a matter of historical connection with the subject or with related matters, or of legal, commercial, philosophical, ethical or political position regarding them.

The "**Invariant Sections**" are certain Secondary Sections whose titles are designated, as being those of Invariant Sections, in the notice that says that the Document is released under this License. If a section does not fit the above definition of Secondary then it is not allowed to be designated as Invariant. The Document may contain zero Invariant Sections. If the Document does not identify any Invariant Sections then there are none.

The "**Cover Texts**" are certain short passages of text that are listed, as Front-Cover Texts or Back-Cover Texts, in the notice that says that the Document is released under this License. A Front-Cover Text may be at most 5 words, and a Back-Cover Text may be at most 25 words.

A "**Transparent**" copy of the Document means a machine-readable copy, represented in a format whose specification is available to the general public, that is suitable for revising the document straightforwardly with generic text editors or (for images composed of pixels) generic paint programs or (for drawings) some widely available drawing editor, and that is suitable for input to text formatters or for automatic translation to a variety of formats suitable for input to text formatters. A copy made in an otherwise Transparent file format whose markup, or absence of markup, has been arranged to thwart or discourage subsequent modification by readers is not Transparent. An image format is not Transparent if used for any substantial amount of text. A copy that is not "Transparent" is called "**Opaque**".

Examples of suitable formats for Transparent copies include plain ASCII without markup, Texinfo input format, LaTeX input format, SGML or XML using a publicly available DTD, and standard-conforming simple HTML, PostScript or PDF designed for human modification. Examples of transparent image formats include PNG, XCF and JPG. Opaque formats include proprietary formats that can be read and edited only by proprietary word processors, SGML or XML for which the DTD and/or processing tools are not generally available, and the machine-generated HTML, PostScript or PDF produced by some word processors for output purposes only.

The "**Title Page**" means, for a printed book, the title page itself, plus such following pages as are needed to hold, legibly, the material this License requires to appear in the title page. For works in formats which do not have any title page as such, "Title Page" means the text near the most prominent appearance of the work's title, preceding the beginning of the body of the text.

The "**publisher**" means any person or entity that distributes copies of the Document to the public.

A section "**Entitled XYZ**" means a named subunit of the Document whose title either is precisely XYZ or contains XYZ in parentheses following text that translates XYZ in another language. (Here XYZ stands for a specific section name mentioned below, such as "**Acknowledgements**", "**Dedications**", "**Endorsements**", or "**History**".) To "**Preserve the Title**" of such a section when you modify the Document means that it remains a section "Entitled XYZ" according to this definition.

The Document may include Warranty Disclaimers next to the notice which states that this License applies to the Document. These Warranty Disclaimers are considered to be included by reference in this License, but only as regards disclaiming warranties: any other implication that these Warranty Disclaimers may have is void and has no effect on the meaning of this License.

2. VERBATIM COPYING

You may copy and distribute the Document in any medium, either commercially or noncommercially, provided that this License, the copyright notices, and the license notice saying this License applies to the Document are reproduced in all copies, and that you add no other conditions whatsoever to those of this License. You may not use technical measures to obstruct or control the reading or further copying of the copies you make or distribute. However, you may accept compensation in exchange for copies. If you distribute a large enough number of copies you must also follow the conditions in section 3.

You may also lend copies, under the same conditions stated above, and you may publicly display copies.

3. COPYING IN QUANTITY

If you publish printed copies (or copies in media that commonly have printed covers) of the Document, numbering more than 100, and the Document's license notice requires Cover Texts, you must enclose the copies in covers that carry, clearly and legibly, all these Cover Texts: Front-Cover Texts on the front cover, and Back-Cover Texts on the back cover. Both covers must also clearly and legibly identify you as the publisher of these copies. The front cover must present the full title with all words of the title equally prominent and visible. You may add other material on the covers in addition. Copying with changes limited to the covers, as long as they preserve the title of the Document and satisfy these conditions, can be treated as verbatim copying in other respects.

If the required texts for either cover are too voluminous to fit legibly, you should put the first ones listed (as many as fit reasonably) on the actual cover, and continue the rest onto adjacent pages.

If you publish or distribute Opaque copies of the Document numbering more than 100, you must either include a machine-readable Transparent copy along with each Opaque copy, or state in or with each Opaque copy a computer-network location from which the general network-using public has access to download using public-standard network protocols a complete Transparent copy of the Document, free of added material. If you use the latter option, you must take reasonably prudent steps, when you begin distribution of Opaque copies

in quantity, to ensure that this Transparent copy will remain thus accessible at the stated location until at least one year after the last time you distribute an Opaque copy (directly or through your agents or retailers) of that edition to the public.

It is requested, but not required, that you contact the authors of the Document well before redistributing any large number of copies, to give them a chance to provide you with an updated version of the Document.

4. MODIFICATIONS

You may copy and distribute a Modified Version of the Document under the conditions of sections 2 and 3 above, provided that you release the Modified Version under precisely this License, with the Modified Version filling the role of the Document, thus licensing distribution and modification of the Modified Version to whoever possesses a copy of it. In addition, you must do these things in the Modified Version:

A. Use in the Title Page (and on the covers, if any) a title distinct from that of the Document, and from those of previous versions (which should, if there were any, be listed in the History section of the Document). You may use the same title as a previous version if the original publisher of that version gives permission.

B. List on the Title Page, as authors, one or more persons or entities responsible for authorship of the modifications in the Modified Version, together with at least five of the principal authors of the Document (all of its principal authors, if it has fewer than five), unless they release you from this requirement.

C. State on the Title page the name of the publisher of the Modified Version, as the publisher.

D. Preserve all the copyright notices of the Document.

E. Add an appropriate copyright notice for your modifications adjacent to the other copyright notices.

F. Include, immediately after the copyright notices, a license notice giving the public permission to use the Modified Version under the terms of this License, in the form shown in the Addendum below.

G. Preserve in that license notice the full lists of Invariant Sections and required Cover Texts given in the Document's license notice.

H. Include an unaltered copy of this License.

I. Preserve the section Entitled "History", Preserve its Title, and add to it an item stating at least the title, year, new authors, and publisher of the Modified Version as given on the Title Page. If there is no section Entitled "History" in the Document, create one stating the title, year, authors, and publisher of the Document as given on its Title Page, then add an item describing the Modified Version as stated in the previous sentence.

J. Preserve the network location, if any, given in the Document for public access to a Transparent copy of the Document, and likewise the network locations given in the Document for previous versions it was based on. These may be placed in the "History" section. You may omit a network location for a work that was published at least four years before the Document itself, or if the original publisher of the version it refers to gives permission.

K. For any section Entitled "Acknowledgements" or "Dedications", Preserve the Title of the section, and preserve in the section all the substance and tone of each of the contributor acknowledgements and/or dedications given therein.

L. Preserve all the Invariant Sections of the Document, unaltered in their text and in their titles. Section numbers or the equivalent are not considered part of the section titles.

M. Delete any section Entitled "Endorsements". Such a section may not be included in the Modified Version.

N. Do not retitle any existing section to be Entitled "Endorsements" or to conflict in title with any Invariant Section.

O. Preserve any Warranty Disclaimers.

If the Modified Version includes new front-matter sections or appendices that qualify as Secondary Sections and contain no material copied from the Document, you may at your option designate some or all of these sections as invariant. To do this, add their titles to the list of Invariant Sections in the Modified Version's license notice. These titles must be distinct from any other section titles.

You may add a section Entitled "Endorsements", provided it contains nothing but endorsements of your Modified Version by various parties—for example, statements of peer review or that the text has been approved by an organization as the authoritative definition of a standard.

You may add a passage of up to five words as a Front-Cover Text, and a passage of up to 25 words as a Back-Cover Text, to the end of the list of Cover Texts in the Modified Version. Only one passage of Front-Cover Text and one of Back-Cover Text may be added by (or through arrangements made by) any one entity. If the Document already includes a cover text for the same cover, previously added by you or by arrangement made by the same entity you are acting on behalf of, you may not add another; but you may replace the old one, on explicit permission from the previous publisher that added the old one.

The author(s) and publisher(s) of the Document do not by this License give permission to use their names for publicity for or to assert or imply endorsement of any Modified Version.

5. COMBINING DOCUMENTS

You may combine the Document with other documents released under this License, under the terms defined in section 4 above for modified versions, provided that you include in the combination all of the Invariant Sections of all of the

original documents, unmodified, and list them all as Invariant Sections of your combined work in its license notice, and that you preserve all their Warranty Disclaimers.

The combined work need only contain one copy of this License, and multiple identical Invariant Sections may be replaced with a single copy. If there are multiple Invariant Sections with the same name but different contents, make the title of each such section unique by adding at the end of it, in parentheses, the name of the original author or publisher of that section if known, or else a unique number. Make the same adjustment to the section titles in the list of Invariant Sections in the license notice of the combined work.

In the combination, you must combine any sections Entitled "History" in the various original documents, forming one section Entitled "History"; likewise combine any sections Entitled "Acknowledgements", and any sections Entitled "Dedications". You must delete all sections Entitled "Endorsements".

6. COLLECTIONS OF DOCUMENTS

You may make a collection consisting of the Document and other documents released under this License, and replace the individual copies of this License in the various documents with a single copy that is included in the collection, provided that you follow the rules of this License for verbatim copying of each of the documents in all other respects.

You may extract a single document from such a collection, and distribute it individually under this License, provided you insert a copy of this License into the extracted document, and follow this License in all other respects regarding verbatim copying of that document.

7. AGGREGATION WITH INDEPENDENT WORKS

A compilation of the Document or its derivatives with other separate and independent documents or works, in or on a volume of a storage or distribution medium, is called an "aggregate" if the copyright resulting from the compilation is not used to limit the legal rights of the compilation's users beyond what the individual works permit. When the Document is included in an aggregate, this License does not apply to the other works in the aggregate which are not themselves derivative works of the Document.

If the Cover Text requirement of section 3 is applicable to these copies of the Document, then if the Document is less than one half of the entire aggregate, the Document's Cover Texts may be placed on covers that bracket the Document within the aggregate, or the electronic equivalent of covers if the Document is in electronic form. Otherwise they must appear on printed covers that bracket the whole aggregate.

8. TRANSLATION

Translation is considered a kind of modification, so you may distribute translations of the Document under the terms of section 4. Replacing Invariant Sections with translations requires special permission from their copyright holders,

but you may include translations of some or all Invariant Sections in addition to the original versions of these Invariant Sections. You may include a translation of this License, and all the license notices in the Document, and any Warranty Disclaimers, provided that you also include the original English version of this License and the original versions of those notices and disclaimers. In case of a disagreement between the translation and the original version of this License or a notice or disclaimer, the original version will prevail.

If a section in the Document is Entitled "Acknowledgements", "Dedications", or "History", the requirement (section 4) to Preserve its Title (section 1) will typically require changing the actual title.

9. TERMINATION

You may not copy, modify, sublicense, or distribute the Document except as expressly provided under this License. Any attempt otherwise to copy, modify, sublicense, or distribute it is void, and will automatically terminate your rights under this License.

However, if you cease all violation of this License, then your license from a particular copyright holder is reinstated (a) provisionally, unless and until the copyright holder explicitly and finally terminates your license, and (b) permanently, if the copyright holder fails to notify you of the violation by some reasonable means prior to 60 days after the cessation.

Moreover, your license from a particular copyright holder is reinstated permanently if the copyright holder notifies you of the violation by some reasonable means, this is the first time you have received notice of violation of this License (for any work) from that copyright holder, and you cure the violation prior to 30 days after your receipt of the notice.

Termination of your rights under this section does not terminate the licenses of parties who have received copies or rights from you under this License. If your rights have been terminated and not permanently reinstated, receipt of a copy of some or all of the same material does not give you any rights to use it.

10. FUTURE REVISIONS OF THIS LICENSE

The Free Software Foundation may publish new, revised versions of the GNU Free Documentation License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns. See `http://www.gnu.org/copyleft/`.

Each version of the License is given a distinguishing version number. If the Document specifies that a particular numbered version of this License "or any later version" applies to it, you have the option of following the terms and conditions either of that specified version or of any later version that has been published (not as a draft) by the Free Software Foundation. If the Document does not specify a version number of this License, you may choose any version ever published (not as a draft) by the Free Software Foundation. If the Document specifies that a proxy can decide which future versions of this License can be used, that proxy's public statement of acceptance of a version permanently authorizes you to choose that version for the Document.

11. RELICENSING

"Massive Multiauthor Collaboration Site" (or "MMC Site") means any World Wide Web server that publishes copyrightable works and also provides prominent facilities for anybody to edit those works. A public wiki that anybody can edit is an example of such a server. A "Massive Multiauthor Collaboration" (or "MMC") contained in the site means any set of copyrightable works thus published on the MMC site.

"CC-BY-SA" means the Creative Commons Attribution-Share Alike 3.0 license published by Creative Commons Corporation, a not-for-profit corporation with a principal place of business in San Francisco, California, as well as future copyleft versions of that license published by that same organization.

"Incorporate" means to publish or republish a Document, in whole or in part, as part of another Document.

An MMC is "eligible for relicensing" if it is licensed under this License, and if all works that were first published under this License somewhere other than this MMC, and subsequently incorporated in whole or in part into the MMC, (1) had no cover texts or invariant sections, and (2) were thus incorporated prior to November 1, 2008.

The operator of an MMC Site may republish an MMC contained in the site under CC-BY-SA on the same site at any time before August 1, 2009, provided the MMC is eligible for relicensing.

ADDENDUM: How to use this License for your documents

To use this License in a document you have written, include a copy of the License in the document and put the following copyright and license notices just after the title page:

> Copyright © YEAR YOUR NAME. Permission is granted to copy, distribute and/or modify this document under the terms of the GNU Free Documentation License, Version 1.3 or any later version published by the Free Software Foundation; with no Invariant Sections, no Front-Cover Texts, and no Back-Cover Texts. A copy of the license is included in the section entitled "GNU Free Documentation License".

If you have Invariant Sections, Front-Cover Texts and Back-Cover Texts, replace the "with ... Texts." line with this:

> with the Invariant Sections being LIST THEIR TITLES, with the Front-Cover Texts being LIST, and with the Back-Cover Texts being LIST.

If you have Invariant Sections without Cover Texts, or some other combination of the three, merge those two alternatives to suit the situation.

If your document contains nontrivial examples of program code, we recommend releasing these examples in parallel under your choice of free software license, such as the GNU General Public License, to permit their use in free software.

Bibliografia

[1] Alfred V. Aho, Ravi Sethi, and Jeffrey D. Ullman. *Compilers: Princiles, Techniques, and Tools*. Addison-Wesley, 1986.

[2] Ronald L. Graham, Donald E. Knuth, and Oren Patashnik. *Concrete Mathematics: A Foundation for Computer Science*. Addison-Wesley Longman Publishing Co., Inc., Boston, MA, USA, 2nd edition, 1994.

[3] Paul R. Halmos. Review: Alfred tarski, logic, semantics, metamathematics. *Bulletin of the American Mathematical Society*, 63(2):155–156, 03 1957.

[4] John E. Hopcroft, Rajeev Motwani, and Jeffrey D. Ullman. *Introduction to Automata Theory, Languages, and Computation (3rd Edition)*. Addison-Wesley Longman Publishing Co., Inc., Boston, MA, USA, 2006.

[5] Leslie Lamport. How to write a 21st century proof. *Journal of Fixed Point Theory and Applications*, 11(1):43–63, 2012.

[6] Kenneth H. Rosen. *Discrete Mathematics and Its Applications*. McGraw-Hill Higher Education, 5th edition, 2002.

[7] Michael Sipser. *Introduction to the Theory of Computation (3rd Edition)*. Cengage Learning, Boston, MA, USA, 2012.

[8] Dirk van Dalen. *Logic and structure (3. ed.)*. Universitext. Springer, 1994.

[9] Daniel J. Velleman. *How to Prove It: A Structured Approach*. Cambridge University Press, Cambridge, England, 2nd edition edition, January 2006.

[10] Glynn Winskel. *The Formal Semantics of Programming Languages: An Introduction*. MIT Press, Cambridge, MA, USA, 1993.