Notas de Aula de Matemática Discreta

Prof. Dr. Rodrigo Geraldo Ribeiro

2 de Março de 2020

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 [4].
2. How to Prove It: A Structured Approach, Velleman [6],
3. Matemática Concreta, Knuth et. al. [1].
4. Logic and Structure, Van Dalen [5].
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

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.

1.1 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.

1.2 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 1 (Proposição)**.** Definimos por proposição qualquer sentença passível de possuir um dos valores lógicos: verdadeiro ou falso. ■

Exemplo 1. 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 2 (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 2. 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 1.1: 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 1.1 apresenta algumas destas palavras e quais conectivos estas representam. Nesta tabela utilizamos as variáveis A e B para denotar proposições quaisquer.

Exemplo 3. 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 1.1 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.

1.2.1 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.

1.2.2 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 1.2: Notação para conectivos lógicos

Utilizando a notação presente na tabela 1.2, temos que as sentenças anteriores seriam representadas pelas seguintes fórmulas $A \wedge B$ e $C \wedge D$.

1.2.3 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 1.2.1.

1.3 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 3 (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 4.

Exemplo 4. 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 $A \rightarrow (B \rightarrow (C \rightarrow D))$.

1.3.1 Exercícios

1. Para cada uma dos termos a seguir, use a definição de fórmulas bem formadas (definição 4) 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) $\neg(A \land (B \lor C))$

1.4 Semântica da Lógica Proposicional

As fórmulas da lógica proposicional, descritas na seção 4, 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 [2].

Conforme apresentado no capítulo **??**, 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.

1.4.1 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

1.4.2 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.

1.4.3 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.

1.4.4 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

1.4.5 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.

1.4.6 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

1.4.7 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 4 (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 5. De acordo com a definição 4, 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 6. 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 6 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.

1.4.8 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 5 (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 7. 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 6 (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 8. 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.

1.4.9 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.

1.4.10 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 7 (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 9. 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 1.4.9. Na seção 1.5, apresentaremos o sistema de dedução natural para lógica proposicional que permite verificar consequências lógicas sem a construção de uma tabela verdade.

1.4.11 Exercícios

1. Obtenha o conjunto de subfórmulas de cada fórmula a seguir utilizando a definição 4.
   (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

1.5 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 8 (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.

1.5.1 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.

1.5.2 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 10. 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 11. 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 12. 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 13. 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:

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

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 $P \wedge (Q \wedge R)$. ■

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

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

■

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 15. 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:

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

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 16. 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 17. 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. ■

1.5.4 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 18. 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:

$$\begin{array}{l} \dfrac{\overline{P \wedge Q}\ \{\wedge_{ID}\}}{Q}\ \{\vee_{ID}\} \\ \hline P \vee Q \end{array}$$

■

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:

$$\begin{array}{l} \dfrac{\overline{P \wedge Q}\ \{\wedge_{ID}\}}{P}\ \{\vee_{IE}\} \\ \hline P \vee Q \end{array}$$

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 19. 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 20. 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 $B \vee C$.

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

■

1.5.5 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 21. 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 22. O sequente $\{A \vee B, \neg A\} \vdash B$ possui a seguinte demonstração:

$$\dfrac{\overline{A \vee B}\ \{_{ID}\} \qquad \dfrac{\dfrac{\overline{\neg A}\ \{_{ID}\} \quad \overline{A^1}\ \{_{ID}\}}{\bot}\ \{\rightarrow_{E}\}}{B}\ \{_{CTR}\} \qquad \overline{B^1}\ \{_{ID}\}}{B}\ \{\vee_{E}\}^1$$

■

1.5.6 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 23. 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. ■

1.5.7 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) $\vdash ((A \to B) \to A) \to A$

1.6 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 9 (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 24. 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.

1.6.1 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.

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

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

■

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.

1.6.3 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 26. 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.

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

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

■

1.6.5 Leis Envolvendo a Implicação e Bicondicional

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

$\alpha \to \beta$	$\equiv$	$\neg\alpha \vee \beta$	{implicação}
$\alpha \leftrightarrow \beta$	$\equiv$	$(\alpha \to \beta) \wedge (\beta \to \alpha)$	{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 28. A fórmula $A \to B$ é equivalente a $\neg B \to \neg A$, conforme demonstrado a seguir:

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

Note que no último passo, utilizamos a regra da implicação, que especifica que $\alpha \to \beta \equiv \neg\alpha \vee \beta$, 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 $\neg(\neg B) \vee \neg A = \neg B \to \neg A$. ■

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 10 (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 29. 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 $\vee$.

■

1.6.6 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 $\{\top, \neg, \vee, \wedge, \rightarrow, \leftrightarrow\}$.

1.7 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.

1.7.1 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 11 (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 30. 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 11, o conectivo $\vee$ ocorre apenas em cláusulas e não no nível mais externo da fórmula.

■

A definição 11 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 31. 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 32. 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. ■

1.7.2 Forma Normal Disjuntiva

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

Definição 12 (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 33. 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 34. 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.

■

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

1.8 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.

1.8.1 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.

1.8.2 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 13 (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 14 (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 **??**).

1.9 Notas Bibliográficas

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

2

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.

2.1 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 35. 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 1.8 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.

2.2 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.

2.2.1 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 36. 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)*. ■

2.2.2 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 37. 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.

2.2.3 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 38. 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 39. 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.

2.2.4 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 40. 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 $\forall x.C(x, \text{Clementino}) \rightarrow \neg G(x, \mathit{mãe}(\text{Clementino}))$.

■

2.3 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. ¬∃x.[E(x

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

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

2.4 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.

2.4.1 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 15 (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 15.

Exemplo 41. 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: $f(a, c)$, $g(f(a))$. ■

2.4.2 Fórmulas

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

Definição 16 (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 42. 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 $\neg$.

2.4.3 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 17 (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 43. 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 18 (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 2.5).

2.4.4 Substituição

Conforme discutido no exemplo 43, 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 19 (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 44. 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. ■

2.5 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 16) 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 18. 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) $\exists y.p(y) \wedge \forall x.q(f(a), g(x,b))$.

2.6 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 2.2, 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 20 (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 21 (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 45. 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 22 (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 46. 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 22, é mostrada passo-a-passo a seguir.

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

Logo, de acordo com a definição 22, 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 23 (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. ■

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

2.8 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.

2.8.1 Regras para o quantificador universal

Conforme apresentado na definição 22, 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 47. 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 48. 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 49. 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 50. 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.

2.8.2 Regras para o quantificador existencial

Conforme apresentado na definição 22, 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 51. 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 52. 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 $\{\exists x.P(x)\} \vdash \forall x.P(x)$.

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

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.

2.9 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) $\{\exists x.\neg P(x)\} \vdash \neg\forall x.P(x)$

2.10 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 53. 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:

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

■

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

2.12 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.

2.12.1 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.

2.12.2 Decidibilidade

Conforme apresentamos no capítulo 1, 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.

2.13 Notas Bibliográficas

Parte II

Demonstração de Teoremas e Teoria de Conjuntos

3

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.

3.1 Motivação

Nos capítulos 1 e 2 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.

3.2 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 54. 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.

3.3 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 [6] 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.

3.3.1 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 [6].

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 55. 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 [3].

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 56. 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.

3.3.2 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 57. 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 57.

Exemplo 58. 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 59. 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, $\neg R \to (P \to \neg Q)$.

■

3.3.3 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 $a \neq 0$.

3.3.4 Estratégias para Quantificadores ($\forall$), ($\exists$)

Estratégias de prova para conclusões envolvendo quantificadores são análogas às regras de introdução destes apresentadas nos capítulos 1 e 2.

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 60. 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 24 (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 61. 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 24, 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 $a \mid c$.

■

3.3.5 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 $a \mid b$.

3.3.6 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 62. 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.

■

3.3.7 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 $5 \mid n$.

3.3.8 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 63. 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 64. 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 65. 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 $\neg\beta \rightarrow \alpha$. ■

3.3.9 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 $|x - 4| > 2$.

3.3.10 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.

3.3.11 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 66. 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 $\dfrac{2y}{y+1} = x$.

■

3.3.12 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.

3.3.13 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 67. 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.

■

3.4 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 [6]. Ao contrário de [6], 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.

4

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.

4.1 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.

4.2 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.

4.3 Descrevendo Conjuntos

Existem diversas maneiras para se descrever conjuntos. Apresentaremos, de maneira suscinta três maneiras: enumeração, *set comprehension*[1] e por recursão.

4.3.1 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 68. 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:

$$p \in V$$
$$\text{vaca} \in P$$
$$7 \in X$$
$$2 \in L$$
$$\{1, \{1\}\} \in \{1, \{1\}\}$$

■

4.3.2 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 69. 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 70. 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 71. 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.

4.3.3 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 72. 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 **??** 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 73. 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 74. 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 $-2 \in \mathbb{Z}$:

$$\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{}{\forall n.n \in \mathbb{Z} \rightarrow n + 1 \in \mathbb{Z} \wedge n - 1 \in \mathbb{Z}}\,\{I_D\}}{-1 \in \mathbb{Z} \rightarrow 0 \in \mathbb{Z} \wedge -2 \in \mathbb{Z}}\,\{\forall_E\} \quad \dfrac{\dfrac{}{0 \in \mathbb{Z}}\,\{I_D\} \quad \dfrac{\dfrac{\dfrac{}{\forall n.n \in \mathbb{Z} \rightarrow n + 1 \in \mathbb{Z} \wedge n - 1 \in \mathbb{Z}}\,\{I_D\}}{0 \in \mathbb{Z} \rightarrow 1 \in \mathbb{Z} \wedge -1 \in \mathbb{Z}}\,\{\forall_E\}}{1 \in \mathbb{Z} \wedge -1 \in \mathbb{Z}}\,\{\rightarrow_E\}}{-1 \in \mathbb{Z}}\,\{\wedge_{ED}\}}{0 \in \mathbb{Z} \wedge -2 \in \mathbb{Z}}\,\{\rightarrow_E\}}{-2 \in \mathbb{Z}}\,\{\wedge_{ED}\}$$

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:

$$\{x \in \mathbb{R} \mid 0 \leq x \leq 1\}$$

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.

4.3.4 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.

4.4 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.

4.4.1 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 25 (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 26 (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 27 (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 $A \neq B$. ■

4.4.2 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 28 (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 75. Sejam $A = \{1, 2, 3\}$, $B = \{3, 4, 5\}$, $C = \{4, 5, 6\}$ e $\mathcal{U} = \{1, 2, 3, 4, 5, 6, 7\}$. Então:

$$\begin{array}{lcl} A \cup B & = & \{1, 2, 3, 4, 5\} \\ A \cap B & = & \{3\} \\ A - B & = & \{1, 2\} \\ A \cup C & = & \{1, 2, 3, 4, 5, 6\} \\ A \cap C & = & \emptyset \\ A - C & = & \{1, 2, 3\} \\ \overline{A} & = & \{4, 5, 6, 7\} \end{array}$$

■

4.4.3 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 29 (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 76. 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 30 (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 77. Seja $\mathcal{F} = \{\{1, 2, 3\}, \{2, 3, 4\}, \{3, 4, 5\}\}$ uma família de conjuntos. Temos que:

$$\begin{array}{rcccl} \bigcap \mathcal{F} & = & \{1, 2, 3\} \cap \{2, 3, 4\} \cap \{3, 4, 5\} & = & \{3\} \\ \bigcup \mathcal{F} & = & \{1, 2, 3\} \cup \{2, 3, 4\} \cup \{3, 4, 5\} & = & \{1, 2, 3, 4, 5\} \end{array}$$

■

Finalmente, uma notação alternativa para famílias de conjuntos são as chamadas *famílias indexadas*, que são definidas em termos de um conjunto de índices.

Definição 31 (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 78. 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:

$$\begin{array}{lcll} \mathcal{F} & = & \{A_1, A_2, A_3\} & = \quad \{\{1, 2, 3\}, \{2, 3, 4\}, \{3, 4, 5\}\} \\ \bigcup_{i \in \{1,2,3\}} & = & \{1, 2, 3, 4, 5\} & \\ \bigcap_{i \in \{1,2,3\}} & = & \{3\} & \end{array}$$

■

4.4.4 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).

4.5 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 79. 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 80. 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:

$$\begin{array}{ll} [C \cap (A \cup B)] \cup [(A \cup B) \cap \overline{C}] & \equiv \\ [(A \cup B) \cap C] \cup [(A \cup B) \cap \overline{C}] & \equiv \\ (A \cup B) \cap (C \cup \overline{C}) & \equiv \\ (A \cup B) \cap \mathcal{U} & \equiv \\ A \cup B & \end{array}$$

■

4.5.1 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) $\overline{[(\overline{A} \cup \overline{B}) \cap \overline{A}]} \equiv A$

4.6 Teoremas Envolvendo Conjuntos

As técnicas de demonstração apresentadas no capítulo 3 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 81. 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 82. 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 83. 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 84. 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 85. 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.

4.6.1 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 $\mathcal{F}$, $\bigcup!\mathcal{F} \subseteq \bigcup\mathcal{F}$.

4.7 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 [6].

Todos os exemplos de teoremas envolvendo conjuntos podem ser encontrados em [6].

Parte III

Indução e Recursividade

5

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.

5.1 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.

5.2 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 6. *Para todo $n \in \mathbb{N}$, temos que $\sum_{k=0}^{n} 2^k = 2^{n+1} - 1$*

Comentário 1. 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 7. *Para todo $n \in \mathbb{N}$, $3 \,|\, (n^3 - n)$.*

Comentário 2. Novamente, utilizaremos indução matemática. Note que o enunciado do teorema é dado por:

$$\forall n.n \in \mathbb{N} \rightarrow 3 \,|\, (n^3 - n)$$

em que a propriedade a ser demonstrada é

$$3 \,|\, (n^3 - n)$$

Note que $3 \,|\, (n^3 - n)$ é na verdade uma fórmula envolvendo um quantificador existencial.

$$3 \,|\, (n^3 - n) \equiv \exists k.k \in \mathbb{N} \wedge n^3 - n = 3k$$

Logo, temos que mostrar a veracidade das seguintes fórmulas, para demonstrar o teorema usando indução:

- $3 \,|\, (0^3 - 0)$
- $\forall n.n \in \mathbb{N} \wedge 3 \,|\, (n^3 - n) \rightarrow 3 \,|\, ((n+1)^3 - (n+1))$

A demonstração de $3 \,|\, (0^3 - 0)$ 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

$$\forall n.n \in \mathbb{N} \wedge 3 \,|\, (n^3 - n) \rightarrow 3 \,|\, ((n+1)^3 - (n+1))$$

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 8. *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 3. 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).

■

5.2.1 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 $n \geq 1$, $5 \,|\, (n^5 - n)$
   (e) Para todo $n \geq 1$, $6 \,|\, (7^n - 1)$
   (f) Para todo $n \geq 1$, $6 \,|\, (n^3 + 5n)$
   (g) Para todo $n \in \mathbb{N}$, $2 \mid (n^2 + n)$

5.3 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 9. *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:

$$\forall n.n \in \mathbb{N} \to n > 1 \to n \text{ é primo} \vee n \text{ é produto de primos}$$

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 $\forall n.n \in \mathbb{N} \to P(n)$)**.** Prove que a seguinte fórmula é verdadeira:

$$\forall n.n \in \mathbb{N} \to (\forall k.k \in \mathbb{N} \wedge k < n \to P(k)) \to P(n)$$

■

O ponto chave de uma demonstração usando indução está em sua hipótese de indução:

$$\forall k.k \in \mathbb{N} \wedge k < n \to P(k)$$

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 $\forall n.n \in \mathbb{N} \to P(n)$? Para a indução convencional, mostramos que esta é equivalente a $\forall n.n \in \mathbb{N} \to P(n)$ 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 9.

[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 10. *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 11 (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 12. $\sqrt{2}$ *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 11), 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. □

5.3.1 Exercícios

1. Prove que $\sqrt{3}$ é irracional.

5.4 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 13. *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 14. *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.

5.5 Notas Bibliográficas

Indução matemática é tema de todo livro de matemática discreta. Os exemplos e definições deste capítulo foram obtidos de [6].

6

Recursividade

Recursive. adj. See
RECURSIVE.

Stan Kelly-Bootie — The Devil's
DP Dictionary

6.1 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.

6.2 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 15. *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 16. *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, $f(n+1) = 2^{n+1} - 1$.

□

6.2.1 Conjunto Potência, Recursivamente

No capítulo 4, 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 **??**).

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 17. *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 $|\mathcal{P}(A)| = |\mathcal{P}(B)| + |X| = 2^n + 2^n = 2^{n+1}$.

□

6.2.2 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 18. *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.

6.3 Problemas Recursivos

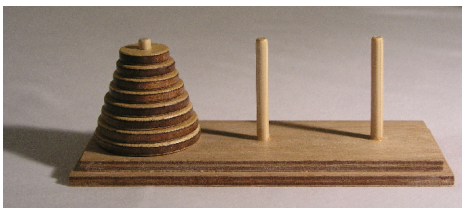
A seguir apresentamos alguns problemas clássicos e como estes podem ser modelados utilizando funções recursivas. Além dessa modelagem, apresentaremos como obter uma fórmula fechada equivalente a função apresentada.

6.3.1 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:

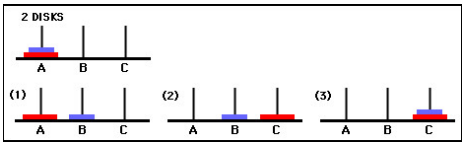
O objetivo deste jogo é transferir todos os discos de um pino para outro movendo apenas um disco de cada vez e nunca colocando um disco maior em cima de um menor.

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?

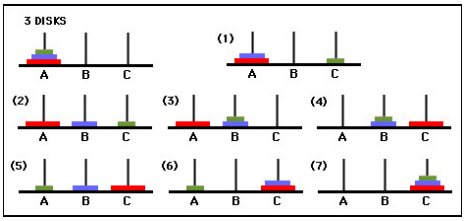


Para chegar a resposta para esta pergunta, devemos primeiro introduzir algumas notações. Chamaremos de $T(n)$ o número de movimentos necessários para solucionar o quebra cabeças contendo n discos.

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 $n = 3$:

$$\begin{array}{ll} T(3) & = \\ 2T(2)+1 & = \\ 2(2T(1)+1)+1 & = \\ 2(2(2T(0)+1)+1)+1 & = \\ 2(2(2.0+1)+1)+1 & = \\ 7 & \end{array}$$

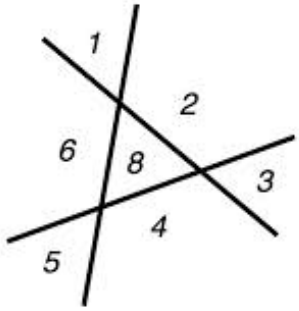
conforme requerido. Como vimos anteriormente, funções recursivas são usualmente ineficientes para o cálculo manual. Logo, é uma boa prática encontrarmos uma fórmula fechada para a função em questão. Porém, já encontramos esta fórmula no teorema 16.

6.3.2 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 19. *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:

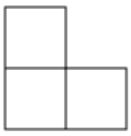
$$\begin{array}{lcl} T(n+1) & = & \\ T(n) + (n+1) & = & \{\text{pela def. de } T(n)\} \\ \dfrac{n(n+1)}{2} + 1 + (n+1) & = & \{\text{pela hipótese de indução}\} \\ \dfrac{n(n+1) + 2(n+1)}{2} + 1 & = & \\ \dfrac{(n+1)(n+2)}{2} + 1 & & \end{array}$$

conforme requerido.

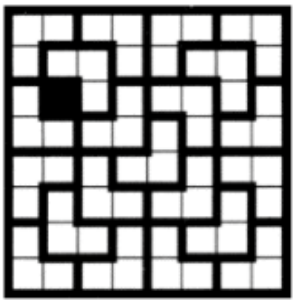
□

6.3.3 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:

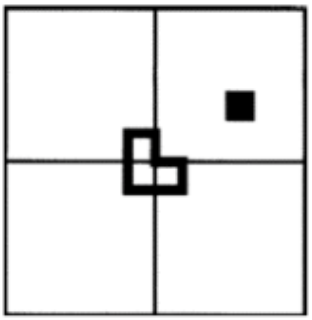


de maneira que apenas uma posição do tabuleiro não seja ocupada por estas peças. Abaixo apresentamos a solução deste quebra-cabeças para um tabuleiro de $2^3 \times 2^3$:



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 20. *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.

□

6.4 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 $|\mathcal{P}_2(A)| = \frac{n(n-1)}{2}$.

6.5 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 [1].

7

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

7.1 Motivação

Como vimos nos dois capítulos anteriores (capítulos 5 e 6), 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 **??**.

7.2 Indução Estrutural

Conforme apresentado no capítulo **??**, 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.

7.2.1 Números Naturais na Notação de Peano

Conforme apresentado no capítulo **??**, o conjunto $\mathcal{N}$, dos termos que representam números naturais na notação de Peano, pode ser definido pelas seguintes regras:

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

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 32 (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 5, 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 21. *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 22 (*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:

$$\begin{array}{lll} plus(zero, plus(m,p)) & \equiv & \\ plus(m,p) & & \{\text{pela equação 1 de } plus\} \end{array}$$

   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.

□

7.2.2 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.

7.2.3 Listas

Nesta seção, consideraremos algumas funções sobre listas e provas de propriedades sobre estas utilizando indução estrutural. No capítulo **??**, 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 33 (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 23. *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 24. *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.

7.2.4 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 $xs \in List\,\mathcal{T}$, $\forall ys.rev\,xs\,ys \equiv reverse\,xs + +ys$, em que *reverse* é a primeira definição apresentada de reverse neste texto.

4. Prove que para toda lista $xs \in List\,\mathcal{T}$, $reverse(reverse\,xs) \equiv xs$.

7.2.5 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 34 (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:

$$\begin{array}{l} \mathit{Leaf} \in \mathit{Tree}\,\mathcal{T} \\ \text{se } l, r \in \mathit{Tree}\,\mathcal{T} \text{ e } x \in \mathcal{T} \text{ então } \mathit{Node}\,x\,l\,r \in \mathit{Tree}\,\mathcal{T} \end{array}$$

■

7.3 Predicados Definidos Indutivamente

7.4 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] 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.

[2] Paul R. Halmos. Review: Alfred tarski, logic, semantics, metamathematics. *Bulletin of the American Mathematical Society*, 63(2):155–156, 03 1957.

[3] Leslie Lamport. How to write a 21st century proof. *Journal of Fixed Point Theory and Applications*, 11(1):43–63, 2012.

[4] Kenneth H. Rosen. *Discrete Mathematics and Its Applications*. McGraw-Hill Higher Education, 5th edition, 2002.

[5] Dirk van Dalen. *Logic and structure (3. ed.)*. Universitext. Springer, 1994.

[6] Daniel J. Velleman. *How to Prove It: A Structured Approach*. Cambridge University Press, Cambridge, England, 2nd edition edition, January 2006.