para todo x

Uma introdução à Lógica Formal

P. D. Magnus

para todoχ

Uma introdução à Lógica Formal

P.D. Magnus

Universidade de Albany, Universidade do Estado de Nova Iorque

Traduzido por:

Lauro Iane de Morais

Universidade Estadual Vale do Acaraú e doutorando no PPGF-UFS

Alessandro Bandeira Duarte

Universidade Federal Rural do Rio de Janeiro

William de Siqueira Piauí

Universidade Federal de Sergipe

O autor gostaria de agradecer às pessoas que tornaram este projeto possível. Dentre essas temos Cristyn Magnus, que leu muitos dos primeiros esboços; Aaron Schiller, que adotou o livro ainda durante seu processo de produção e fez observações atenciosas e importantes; Bin Kang, Craig Erb, Nathan Carter, Wes McMichael, Selva Samuel, Dave Krueger, Brandon Lee, Toan Tran, Marcus Adams, Matthew Brown e os estudantes de Introdução à Lógica, que detectaram vários erros nas primeiras versões do livro.

Dados Internacionais de Catalogação na Publicação (CIP)

MAGNUS, P.D.

Para todo x: uma introdução à Lógica Formal [recurso eletrônico] / P.D. Magnus -- Porto Alegre, RS: Editora Fi, 2021.

191 p.

ISBN - 978-65-5917-330-3
DOI - 10.22350/9786559173303

Disponível em: http://www.editorafi.org

1. Lógica formal; 2. Teoria; 3. Exercício; 4. Provas; 5. P. D. Magnus.; I. Título.

CDD: 100

Índices para catálogo sistemático:
1. Filosofia 100

Tradução para o português em 2021 por: Lauro Iane de Morais, Alessandro Bandeira Duarte e William de Siqueira Piauí

A editoração foi executada inteiramente em LaTeX2ε e LyX. O estilo para editoração das provas é baseado em fitch.sty (v0.4) de Peter Selinger, Universidade de Ottawa.

Esta cópia de `para todo`*x* é datada de 18 de outubro de 2021.

Sumário

Capítulo 1

O que é Lógica?

A ocupação da lógica é avaliar argumentos, separando os bons dos ruins. Um argumento lógico é estruturado de modo a dar a alguém uma razão para acreditar na conclusão. Aqui está um exemplo:

1. Está chovendo muito.
2. Se você não levar um guarda-chuva, você ficará ensopado.

$\therefore$ Você deveria levar um guarda-chuva.

Os três pontos na última linha do argumento significam "Portanto" e eles indicam que a sentença final é a *conclusão* do argumento. As outras sentenças são as *premissas* do argumento.

Os três pontos na última linha do argumento significam "Portanto" e eles indicam que a sentença final é a *conclusão* do argumento. As outras sentenças são as *premissas* do argumento. Se você acredita nas premissas, então o argumento lhe dá uma razão para acreditar na conclusão.

Este capítulo discute algumas noções lógicas básicas que se aplicam a argumentos em uma linguagem natural como o português. É importante começar com uma compreensão clara do que são os argumentos e qual o significado de se dizer que um argumento é válido. Posteriormente, traduziremos argumentos do português para a linguagem formal. Queremos que a validade formal, como definida na linguagem formal, tenha ao menos algumas das características importantes de validade da linguagem natural.

1.1 Argumentos

Quando as pessoas desejam argumentar, costumeiramente utilizam palavras como 'portanto' e 'porque'. Ao analisarmos um argumento, a primeira coisa a se fazer é separar as premissas da conclusão. Palavras como essas são chaves para mostrar onde o argumento quer chegar, principalmente caso a conclusão venha no começo ou no meio.

Indicadores de premissa: desde que, já que, porque, dado que

Indicadores de conclusão: portanto, logo, assim, consequentemente.

Em linhas gerais, podemos dizer que um argumento é uma série de sentenças. A sentença a que se quer chegar é a conclusão, as demais são premissas. Se as premissas são verdadeiras e o argumento é bom, então você tem razões para aceitar a conclusão.

Considere o exemplo:

Há café no jarro de café.

Há um dragão tocando fagote no armário

$\therefore$ Salvador Dali é jogador de poker.

Pode parecer estranho chamar isso de argumento, mas isso é porque esse seria um péssimo argumento. As duas premissas nada têm a ver com a conclusão. Não obstante, isso ainda é um argumento — apesar de ser ruim.

1.2 Sentenças

Em lógica, estamos interessados só nos enunciados que são verdadeiros ou falsos e podem assumir papel de premissa ou conclusão do argumento. Tais enunciados chamaremos de SENTENÇAS.

Não se deve confundir sentença, que pode ser verdadeira ou falsa, com a diferença entre fato e opinião. Em lógica muitas vezes as sentenças expressam coisas consideradas fatos — como, por exemplo, 'Kierkegaard era corcunda' ou

'Kierkegaard gostava de amêndoas.' Mas elas também podem expressar coisas consideradas opiniões — do tipo 'amêndoas são gostosas.'

E também existem aquelas que são consideradas 'sentenças' num curso de linguística ou gramática, mas não iremos considerá-las como sentenças na lógica.

Perguntas: em uma aula de gramática, 'Você já está com sono?' seria considerada como uma sentença interrogativa. Embora você possa estar sonolento ou alerta, a pergunta em si não é nem falsa nem verdadeira. Por essa razão, perguntas não serão consideradas como sentenças na lógica. Suponha que você responda àquela pergunta: 'Eu não estou com sono.' Isso é ou verdadeiro ou falso, portanto, essa é uma sentença no sentido lógico. Geralmente, *perguntas* não serão consideradas sentenças, mas *respostas* sim.

Imperativos: comandos são muitas vezes expressos como imperativos do tipo, 'Acorde!', 'Sente corretamente!' etc. Em uma aula de gramática, essas seriam consideradas como sentenças imperativas. Embora seja bom para você sentar corretamente ou não, o comando em si não é nem verdadeiro, nem falso. Note, entretanto, que comandos nem sempre são expressos como imperativos. 'Você irá respeitar minha autoridade' é ou verdadeiro ou falso — ou você irá ou não irá respeitar minha autoridade — e, dessa maneira, será uma sentença no sentido lógico do termo.

Exclamações: 'Ai!' é algumas vezes chamada de sentença exclamatória, mas não é nem verdadeira, nem falsa. Nós trataremos 'Ai! Machuquei meu dedo!' como significando a mesma coisa que 'Machuquei meu dedo.' O 'ai' nada acrescenta que possa ser verdadeiro ou falso.

1.3 Duas formas nas quais argumentos podem dar errado

Considere o argumento concluindo que você deveria levar o guarda-chuva (na p. 5). Se a premissa (1) é falsa — ou seja, se está ensolarado — então, o argumento não te dá razão para levar um guarda-chuva. E mesmo se estiver chovendo do lado de fora, você pode não precisar de um guarda-chuva. Você poderia usar uma capa de chuva ou se manter debaixo de um abrigo. Nesses casos, a premissa (2) seria falsa, já que você poderia sair sem um guarda-chuva e, ainda assim, evitar ficar ensopado.

Suponha por um momento que ambas as premissas sejam verdadeiras. Você não tem uma capa de chuva e precisa ir para lugares onde não há qualquer abrigo. Agora, o argumento lhe mostra que você deveria levar um guarda-chuva? Não necessariamente. Talvez você goste de andar na chuva e gostaria de se molhar. Nesse caso, apesar das premissas serem verdadeiras, a conclusão seria falsa.

Para qualquer argumento, existem dois modos segundo os quais ele pode ser fraco. Em primeiro lugar, uma premissa ou mais podem ser falsas — um argumento lhe dá razão para acreditar em sua conclusão apenas se você acreditar em suas premissas. Em segundo lugar, as premissas podem não respaldar a conclusão. Mesmo se as premissas forem verdadeiras, a forma do argumento pode ser fraca. O exemplo que demos é considerado fraco em ambos os sentidos.

Quando um argumento é fraco no segundo sentido, há algo errado com a forma lógica do argumento: as premissas fornecidas não levam necessariamente à conclusão dada. Nós estamos interessados principalmente na forma lógica dos argumentos.

Considere outro exemplo:

Você está lendo este livro.

Este é um livro de lógica.

∴ Você é um estudante de lógica.

Esse não é um argumento horrível. A maioria das pessoas que leem este livro são estudantes de lógica, mas, ainda assim, é possível que alguém, além de um estudante de lógica, o esteja lendo. Se um colega seu pegasse esse livro e o folheasse, ele não se tornaria automaticamente um estudante de lógica. Então, as premissas desse argumento, apesar de verdadeiras, não garantem a verdade da conclusão. Sua forma lógica é imperfeita.

Um argumento que não fosse fraco no segundo sentido, teria uma forma lógica perfeita. Se suas premissas forem verdadeiras, então sua conclusão *necessariamente* será verdadeira. Chamamos argumentos desse tipo de ‘dedutivamente válidos’ ou simplesmente ‘válidos’.

Mesmo que possamos entender o argumento acima como bom, em algum sentido, ele não é válido; i.e., é 'inválido'. Uma tarefa importante da lógica é separar os argumentos válidos dos inválidos.

1.4 Dedutivamente válido

Um argumento é dedutivamente VÁLIDO se e somente se é impossível suas premissas serem verdadeiras e sua conclusão falsa.

Laranjas são frutas ou instrumentos musicais.

Laranjas não são frutas.

∴ Laranjas são instrumentos musicais.

A conclusão desse argumento é ridícula, entretanto, ela se segue validamente das premissas. Logo, esse argumento é válido. *Se* ambas as premissas são verdadeiras, *então* a conclusão é necessariamente verdadeira.

Isso mostra que argumentos dedutivamente válidos não precisam ter premissas ou conclusões verdadeiras. Ao contrário, ter premissas e conclusão verdadeiras não é suficiente para fazer um argumento válido. Considere o exemplo:

Londres está na Inglaterra.

Beijing está na China.

∴ Paris está na França.

As premissas e a conclusão deste argumento são verdadeiras. Entretanto, esse argumento é terrível porque as premissas nada têm a ver com a conclusão. Imagine o que aconteceria se Paris declarasse independência do resto da França. Então a conclusão seria falsa, ainda que ambas as premissas fossem verdadeiras. Por isso esse argumento é inválido. Portanto, é logicamente possível as premissas de um argumento serem verdadeiras e a conclusão falsa.

Deve-se lembrar que a validade não diz respeito a verdade ou falsidade factual das sentenças num argumento. Ao invés disso, a validade do argumento é baseada em sua forma: a verdade das premissas é incompatível com a falsidade da conclusão.

Argumentos indutivos

Porém, podem existir argumentos bons que falham em ser dedutivamente válidos. Considere este:

Em janeiro de 2010, choveu no Rio.

Em janeiro de 2011, choveu no Rio.

Em janeiro de 2012, choveu no Rio.

$\therefore$ Chove todo janeiro no Rio.

Esse é um argumento INDUTIVO, porque ele generaliza a partir de muitos casos para uma conclusão sobre todos os casos.

O argumento poderia se tornar forte com a adição de mais premissas: em janeiro de 2013, choveu no Rio. Em janeiro de ... etc. Entretanto, independentemente de quantas premissas adicionemos, o argumento não será dedutivamente válido. É possível, apesar de improvável, que não chova próximo janeiro no Rio. Não obstante, sabemos que o clima pode ser imprevisível. Nenhuma quantidade de evidências pode nos convencer de que choverá *todo* janeiro no Rio. Quem poderá afirmar que não haverá um ano bizarro no qual não choverá no Rio? Até mesmo um simples contraexemplo é suficiente para fazer a conclusão do argumento falsa.

Argumentos indutivos, por melhor que sejam, não são dedutivamente válidos. Não estamos interessados em argumentos desse tipo neste livro.

1.5 Outras noções lógicas

Além da validade dedutiva, temos interesse por outros conceitos lógicos.

Valor de verdade

Uma sentença pode ser verdadeira ou falsa e esse é seu VALOR DE VERDADE. Definimos sentenças como aquilo que pode ser verdadeiro ou falso; em vez disso, poderíamos ter dito que sentenças são coisas que podem ter valor de verdade.

Verdade lógica

Ao considerar argumentos formalmente, nós nos preocupamos com o que se segue se as premissas forem verdadeiras. Em geral, não nos preocupamos com o valor de verdade factual de quaisquer sentenças particulares — se elas são *realmente* verdadeiras ou falsas. Apesar disso, existem algumas sentenças que têm de ser verdadeiras somente por uma questão de lógica.

Considere as seguintes sentenças:

1. Está chovendo.
2. Ou está chovendo ou não está chovendo.
3. Está chovendo e não está chovendo.

Para saber ou crer que a sentença 1 é verdadeira, você precisa olhar lá fora ou checar a previsão do tempo. Logicamente falando, tal sentença pode ser verdadeira ou falsa. Sentenças desse tipo são chamadas CONTINGENTES.

A sentença 2 é diferente. Você não precisa olhar lá fora para saber que ela é verdadeira. Independentemente de como está o tempo, está chovendo ou não está. Essa sentença é *logicamente verdadeira*; i.e. ela é verdadeira meramente pela lógica, independentemente de como está realmente o mundo. Uma sentença logicamente verdadeira é chamada TAUTOLOGIA.

Você também não precisa checar o tempo para saber sobre a sentença 3. Ela tem de ser falsa, simplesmente em função de sua estrutura lógica. Pode estar chovendo aqui e não chovendo do outro lado da cidade, pode estar chovendo agora, mas parar enquanto você está lendo. Entretanto, é impossível estar chovendo e não chovendo aqui e ao mesmo tempo.

A terceira sentença é *logicamente* falsa; ela é falsa independentemente de como está realmente o mundo. Uma sentença logicamente falsa é chamada CONTRADIÇÃO.

Podemos definir uma SENTENÇA CONTINGENTE como uma sentença que não é nem uma tautologia, nem uma contradição.

Uma sentença pode ser *sempre* verdadeira e ainda ser contingente. Por exemplo, assumamos que nunca houve um tempo no qual o universo conteve menos que sete coisas, então a sentença ‘Pelo menos 7 coisas existem’ seria sempre verdadeira. Mesmo assim, a sentença é contingente; sua verdade não é logicamente conhecida. Não há contradição envolvida em conceber um mundo possível no qual haja menos que sete coisas. A questão realmente importante é se a sentença *deve* ser verdadeira, somente no que diz respeito à sua estrutura lógica.

Equivalência lógica

Podemos também considerar as relações lógicas entre duas sentenças. Por exemplo:

José lavou a louça depois de ir à loja .

José lavou a louça antes de ir à loja.

Ambas as sentenças são contingentes, dado que José pode não ter ido a loja nem lavado a louça. E ainda assim, elas devem ter o mesmo valor de verdade. Se qualquer uma das sentenças é verdadeira, então ambas têm de ser; se qualquer uma delas é falsa, então ambas têm de ser. Quando duas sentenças necessariamente tem o mesmo valor de verdade, nós dizemos que elas são LOGICAMENTE EQUIVALENTES.

Consistência

Considere estas duas sentenças:

B1 Meu único irmão é mais alto que eu.

B2 Meu único irmão é mais baixo que eu.

Apenas a lógica não pode nos dizer qual dessas sentenças é verdadeira. Ainda assim, podemos dizer que *se* a primeira sentença B1 é verdadeira, *então* a segunda sentença B2 tem de ser falsa. E se B2 é verdadeira, B1 tem de ser falsa. Não pode ser o caso de ambas as sentenças serem verdadeiras.

Se sentenças de um conjunto não puderem ser todas verdadeiras ao mesmo tempo, como B1-B2, então elas são chamadas INCONSISTENTES. Caso contrário, elas serão CONSISTENTES.

Podemos considerar a consistência de qualquer número de sentenças. Por exemplo:

G1 Existem no mínimo quatro girafas no parque de vida selvagem.

G2 Existem exatamente sete gorilas no parque de vida selvagem.

G3 Não existem mais do que dois marcianos no parque de vida selvagem.

G4 Toda girafa no parque de vida selvagem é marciana.

G1 e G4 juntas implicam que existem no mínimo quatro girafas marcianas no parque, o que entra em conflito com G3, a qual implica que não existe mais do que duas girafas marcianas lá. Então, o conjunto de sentenças G1-G4 é inconsistente. Note que a inconsistência nada tem a ver com G2. Apenas ocorre G2 ser parte de um conjunto inconsistente.

Algumas vezes as pessoas dizem que um conjunto inconsistente de sentenças 'contém uma contradição.' Isso quer dizer que seria logicamente impossível todas as sentenças serem verdadeiras ao mesmo tempo. Um conjunto pode ser inconsistente mesmo se cada uma das sentenças contidas nele é ou contingente ou tautológica. Quando uma única sentença é uma contradição, então aquela sentença isolada não pode ser verdadeira.

1.6 Linguagens Formais

Aqui está um famoso argumento válido:

Sócrates é homem.

Todo homem é mortal.

$\therefore$ Sócrates é mortal.

A única maneira pela qual você poderia duvidar da conclusão é negando uma das premissas — a forma lógica é impecável. Mas e este próximo argumento?

Sócrates é homem.

Todo o homem é cenoura.

∴ Sócrates é cenoura.

Esse argumento pode ser menos interessante que o primeiro, porque a segunda premissa é obviamente falsa. Mesmo assim, o argumento é válido. Para ver isso, note que ambos os argumentos têm essa forma:

S é M

Todo H é C.

∴ S é C.

Em ambos os argumentos S significa Sócrates e H, homem. No primeiro argumento, C significa mortal; no segundo, cenoura. Ambos os argumentos têm essa forma, e todo argumento dessa forma é válido. Portanto, ambos são válidos.

O que fizemos aqui foi trocar as palavras como 'homem' ou 'cenoura' por símbolos como 'H' ou 'C', de maneira a deixar sua forma lógica explícita. Essa é a ideia central da lógica formal. Queremos remover do argumento aspectos irrelevantes ou que possam nos distrair, para fazermos sua forma lógica mais clara.

Ao começar por um argumento em uma *linguagem natural* como no português, nós o traduzimos para a *linguagem formal*. Partes das sentenças do português são trocadas por letras e símbolos. O objetivo é revelar a estrutura formal do argumento, assim como fizemos com esses dois.

Existem linguagens formais que funcionam como a simbolização que demos para esses dois argumentos. Uma lógica desse tipo foi desenvolvida por Aristóteles, um filósofo que viveu na Grécia durante o século quarto a.C.. Aristóteles foi um aluno de Platão e tutor de Alexandre, O Grande. A lógica de Aristóteles, com algumas revisões, foi a lógica dominante no mundo ocidental por mais de dois milênios.

Na lógica aristotélica, categorias são trocadas por letras maiúsculas. Toda sentença de um argumento é então representada por uma das quatro formas que os medievais classificaram assim: (A) Todo A é B. (E) Nenhum A é B. (I) Algum A é B. (O) Algum A não é B.

Dessa maneira, é possível descrever *silogismos válidos* do tipo que consideramos acima. Lógicos medievais deram nomes mnemônicos para todos os modos válidos de argumento . Por exemplo, o modo dos nossos dois argumentos foi chamada *BARBARA*. As vogais nesse nome, todas As, representam o fato que as duas premissas e a conclusão são todas formas de sentença (A).

Existem muitas limitações para a lógica aristotélica. Uma delas é ela não fazer distinção entre as classes e os indivíduos. Então a primeira premissa pode ser escrita como 'Todo S é H': Todo Sócrates é Homem. Apesar de sua importância histórica, a lógica aristotélica foi suplantada.

O restante deste livro desenvolverá duas lógicas formais. A primeira é a LS, que significa *lógica sentencial.* Na LS, as menores unidades são as próprias sentenças. Sentenças simples são representadas por letras e conectadas com conectivos lógicos como 'e' e 'não' para fazer sentenças mais complexas.

A segunda é a LQ, que significa *lógica quantificada.* Na LQ, as unidades básicas são objetos, propriedades de objetos, e relações entre objetos.

Quando traduzimos um argumento para uma linguagem formal, esperamos deixar sua estrutura formal mais clara. Queremos incluir o suficiente da estrutura do argumento original, que está em português, para podermos julgar se o argumento é válido ou inválido. Se incluíssemos cada aspecto da língua portuguesa, com todas suas sutilezas e nuances, então não haveria vantagem em traduzi-lo para uma linguagem formal.

Ao mesmo tempo, gostaríamos de uma linguagem formal que nos permita representar vários tipos de argumentos da língua portuguesa. Essa é a razão de preferirmos a LQ à lógica aristotélica; a LQ pode representar todo argumento válido da lógica aristotélica e ainda mais.

Então ao se decidir sobre uma linguagem formal, há inevitavelmente uma tensão entre querer capturar tanto da estrutura o quanto possível e querer uma linguagem formal simples — tanto mais simples é a linguagem formal, quanto mais coisas se perdem. Isso quer dizer que não há linguagem formal perfeita. Algumas farão um trabalho melhor que as outras ao traduzir argumentos particulares da língua portuguesa.

Neste livro, assumimos que *verdadeiro* e *falso* são os únicos valores de verdade possíveis. Linguagens lógicas que assumem isso são chamadas de *bivalentes*, ou seja, que possuem dois valores. A lógica aristotélica, a LS e a LQ são todas bivalentes, mas existem limites na lógica bivalente. Por exemplo, filósofos afirmam que o futuro não está determinado ainda. Se eles estiverem certos, então as sentenças sobre *o que será o caso* ainda não são nem verdadeiras nem falsas. Algumas linguagens formais acomodam isso permitindo que haja sentenças que não são nem verdadeiras nem falsas, mas algo intermediário. Outras linguagens formais, chamadas de lógicas paraconsistentes, permitem que haja sentenças que são verdadeiras e falsas.

As linguagens apresentadas neste livro não são as únicas linguagens formais possíveis. Entretanto, a maioria das lógicas não-padrão são extensões da estrutura formal básica das lógicas bivalentes discutidas aqui. Portanto, esse é um bom lugar para começar.

Sumário de noções lógicas

- ▷ Um argumento é (dedutivamente) VÁLIDO se é impossível para suas premissas serem verdadeiras e sua conclusão falsa; do contrário, ele é inválido.
- ▷ Uma tautologia é uma sentença que tem de ser verdadeira, meramente por sua forma lógica.
- ▷ Uma contradição é uma sentença que tem de ser falsa, meramente sua forma lógica.
- ▷ Uma sentença contingente não é nem uma tautologia nem uma contradição.
- ▷ Duas sentenças são LOGICAMENTE EQUIVALENTES se elas necessariamente possuem o mesmo valor de verdade.
- ▷ Um conjunto de sentenças é consistente se é logicamente possível todos os membros do conjunto serem verdadeiros; do contrário, será inconsistente.

1.7 Exercícios práticos

Ao final de cada capítulo, você encontrará uma série de problemas práticos que revisarão e explorarão o material tratado ao longo de cada capítulo. Não há substituto para o trabalho real a partir de problemas, porque lógica trata mais de uma forma de pensar do que de memorização de fatos. As respostas para alguns desses problemas estão disponíveis no final do livro (Apêndice B); os problemas resolvidos no apêndice são marcados com um $\star$.

Parte A Quais das seguintes são 'sentenças' no sentido lógico?

1. A Inglaterra é menor que a China.
2. A Groenlândia está no sul de Jerusalém.
3. A Bahia fica ao norte de Minas Gerais?
4. O número atômico do hélio é 2.
5. O número atômico do hélio é π.
6. Odeio macarrão cozido demais.
7. Eca! Macarrão cozido demais!
8. Macarrão cozido demais é nojento.
9. Vá com calma.
10. Esta é a última questão.

Parte B Para cada uma das sentenças seguintes: ela é tautológica, contraditória ou contingente?

1. César atravessou o Rubicão.
2. Alguém atravessou o Rubicão.
3. Ninguem jamais atravessou o Rubicão.
4. Se César atravessou o Rubicão, então alguém já o fez.
5. Apesar de César ter atravessado o Rubicão, ninguém jamais atravessou o Rubicão.
6. Se alguém atravessou o Rubicão, então foi César.

$\star$ **Parte C** Volte às sentenças G1-G4 na p. 13 e considere cada um dos seguintes conjuntos de sentenças. Quais são consistentes? Quais são inconsistentes?

1. G2, G3 e G4.

2. G1,G3 e G4.
3. G1, G2 e G4.
4. G1, G2 e G3.

⋆ **Parte D** Quais das seguintes sentenças são possíveis? Se for possível, dê exemplos. Se não o for, explique o porquê.

1. Um argumento válido que tenha uma premissa falsa e uma verdadeira.
2. Um argumento válido que tenha uma conclusão falsa.
3. Um argumento válido no qual a conclusão é uma contradição.
4. Um argumento inválido no qual a conclusão é uma tautologia.
5. Uma tautologia que seja contingente.
6. Duas sentenças que sejam tautológicas e sejam logicamente equivalentes.
7. Duas sentenças em que uma seja tautológica e a outra seja contingente e ainda sejam logicamente equivalentes.
8. Duas sentenças equivalentes que juntas sejam um conjunto inconsistente.
9. Um conjunto consistente de sentenças que contenha uma contradição.
10. Um conjunto inconsistente de sentenças que contenha uma tautologia.

Capítulo 2

Lógica Sentencial[1]

Esse capítulo introduz uma linguagem lógica chamada LS. É uma versão de *lógica sentencial*, porque as unidades básicas da linguagem representarão uma sentença.

2.1 Letras sentenciais

Em LS, letras maiúsculas são usadas para representar sentenças básicas. Considerada apenas como um símbolo de LS, a letra A poderia significar qualquer sentença. Enquanto traduzirmos do português para LS, é importante fazermos uma chave de simbolização. Essa chave fornece uma sentença em português para cada letra sentencial utilizada na simbolização.

Por exemplo, considere o argumento:

Há uma maçã na mesa.

Se há uma maçã na mesa, então Joana veio para a aula.

$\therefore$ Joana veio para a aula.

Esse é obviamente um argumento válido em português. Quando simbolizarmos do português para a LS, precisamos preservar a estrutura do argumento, que o faz ser válido. O que acontece se trocarmos cada sentença por uma letra? Nossa chave de simbolização seria esta:

[1] Também chamada de Lógica Proposicional.

A: Há uma maçã na mesa.

B: Se há uma maçã na mesa, então Joana veio para a aula.

C: Joana veio para a aula.

E então simbolizaríamos o argumento deste modo:

A

B

$\therefore C$

Não existe conexão necessária entre alguma sentença A — que poderia ser qualquer sentença — e alguma outra sentença B e C — que poderiam ser quaisquer sentenças. A estrutura do argumento foi completamente perdida nesta tradução.

O mais importante sobre esse argumento é que a segunda premissa não é simplesmente *qualquer* sentença, separada logicamente das outras do argumento. Ela contém a primeira premissa e a conclusão *enquanto partes.* Nossa chave de simbolização para o argumento apenas precisa incluir os significados de A e C, desse modo, podemos construir a segunda premissa a partir deles. Então simbolizemos assim:

A

Se A, então C.

$\therefore C$

Dessa maneira, a estrutura do argumento, que o faz ser válido, é preservada. Entretanto, ele ainda faz uso da expressão portuguesa 'Se... então ... .' Apesar de querermos substituir todas as expressões do português por um notação lógica, esse é um bom começo.

As sentenças que podem ser simbolizadas por letras sentenciais são chamadas de *sentenças atômicas*, porque elas são os blocos de construção básicos a partir dos quais sentenças mais complexas são construídas. Seja qual a for a estrutura lógica que uma sentença possa ter, ela é perdida quando traduzida por uma sentença atômica. Do ponto de vista da LS, uma sentença é apenas uma letra. Ela pode ser utilizada para construir sentenças mais complexas, mas não pode ser decomposto.

Apesar de existirem apenas 26 letras no alfabeto, não há limite lógico quanto ao número de sentenças atômicas. Podemos utilizar a mesma letra para simbolizar sentenças atômicas diferentes adicionando um subscrito, um pequeno número escrito após a letra. Poderíamos ter uma chave de simbolização como esta:

$\mathbf{A_1}$: A maçã está embaixo do armário.

$\mathbf{A_2}$: Argumentos em LS sempre contêm sentenças atômicas.

$\mathbf{A_3}$: Ronaldo está num avião de São Paulo para Paris.

$\vdots$

$\mathbf{A_{294}}$: A aliteração irrita astronautas que, de outro modo, seriam afáveis.

Lembre-se que cada uma delas é uma letra sentencial diferente. Quando há subscritos na chave de simbolização, é importante se manter atentos a eles.

2.2 Conectivos

Conectivos lógicos são utilizados para construir sentenças complexas a partir de componentes atômicos. Existem 5 deles na LS. A seguinte tabela apresenta um resumo, e eles são explicados abaixo.

Símbolo	Como é chamado	O que significa
$\neg$	Negação	'Não é o caso que ...'
&	Conjunção	'... e ...'
$\vee$	Disjunção	'Ou... ou ...'
$\rightarrow$	Condicional	'Se ..., então ...'
$\leftrightarrow$	Bicondicional	'... se e somente se ...'

Negação

Pense sobre como podemos simbolizar estas sentenças:

1. Maria está em São Paulo.
2. Maria não está em São Paulo.
3. Maria está em outro lugar fora São Paulo.

Para simbolizarmos a sentença 1, precisaremos de uma letra sentencial. Desse modo, teremos:

B: Maria está em São Paulo.

Note que aqui estamos dando uma interpretação diferente a B da que a demos na seção anterior. Essa chave de simbolização especifica o que B significa em um *contexto específico*. É vital que continuemos usando esse significado de B enquanto falarmos que Maria está em São Paulo. Posteriormente, quando estivermos simbolizando sentenças diferentes, podemos escrever uma nova chave de simbolização e nesta utilizar B para significar outra coisa. Sendo assim, a sentença 1 é simplesmente B.

E já que a sentença 2 é obviamente ligada a sentença 1, não queremos introduzir uma nova letra sentencial. Se a disséssemos em português, teríamos 'Não B'. Para conseguirmos simbolizá-la, precisaremos de um símbolo para negação lógica, então usaremos '$\neg$.' Agora temos que 'Não B' pode ser traduzido por $\neg B$.

A sentença 3 diz respeito se Maria está ou não em São Paulo, mas não contém a palavra 'não.' Apesar disso, ela é obviamente equivalente à 2. Ambas significam que: não é o caso que Maria esteja em São Paulo. Sendo assim, tanto 2, quanto 3 podem ser traduzidas por $\neg B$.

Uma sentença pode ser simbolizada como $\neg\mathcal{A}$, se ela puder ser parafraseada em português por 'Não é o caso que $\mathcal{A}$.'

Pense sobre as próximas sentenças:

4. A peça pode ser substituído se quebrar.
5. A peça é insubstituível.
6. A peça não é insubstituível.

Se R significar 'A peça é substituível,' então a sentença 4 pode ser traduzida por R. E quanto à sentença 5? Dizer que a peça é insubstituível significa dizer que não é o caso que a peça seja substituível. Então, apesar da sentença 5 não ser negativa no português, nós a simbolizaremos usando uma negação, como $\neg R$.

A sentença 6 pode ser parafraseada por 'Não é o caso que a peça seja insubstituível.' Usando uma negação duas vezes, nós traduzimos isso por $\neg\neg R$. As duas negações usadas em sequência funcionam como negações, então a sentença significa 'Não é o caso que .... não é o caso que ... R.' Se você pensar sobre a sentença em português, concluirá que ela é logicamente equivalente a sentença 4. Então, quando definirmos equivalência lógica na LS, nos certificaremos que R e $\neg\neg R$ são logicamente equivalentes.

Mais exemplos:

7. Júlia está feliz.
8. Júlia está infeliz.

Se H significar 'Júlia está feliz', então simbolizaremos 7 por H. Entretanto, seria um erro simbolizar a sentença 8 como $\neg H$. Se Júlia está infeliz, então ela não está feliz — mas a sentença 8 não significa a mesma coisa que "Não é o caso que Júlia esteja feliz". Pode ser o caso que ela não esteja feliz, nem infeliz. Talvez ela esteja em algum ponto entre os dois estados. Para podermos simbolizar a possibilidade de ela estar indiferente, precisaremos de uma outra letra sentencial para a sentença 8.

Para qualquer sentença $\mathcal{A}$: Se $\mathcal{A}$ é verdadeira, então $\neg\mathcal{A}$ é falsa. Se $\neg\mathcal{A}$ é verdadeira, então $\mathcal{A}$ é falsa. Usando 'V' para verdadeiro e 'F' para falso, podemos resumir isso em uma *tabela de verdade específica* para a negação:

$\mathcal{A}$	$\neg\mathcal{A}$
V	F
F	V

Discutiremos a tabela de verdade posteriormente.

Conjunção

Reflita sobre as sentenças:

9. Roberto é atlético.
10. Bárbara é atlética.
11. Roberto é atlético e Bárbara também.

Precisaremos de letras separadas para 9 e 10, então definiremos essa chave de simbolização:

A: Roberto é atlético

B: Bárbara é atlética.

A sentença 9 pode ser simbolizada por A.

A sentença 10 pode ser simbolizada por B.

A sentença 11 pode ser parafraseada por 'A *e* B.' Para conseguirmos simbolizar essa sentença, precisaremos de outro símbolo. Nós usaremos '&.' Nós traduziremos 'A e B' por $A\&B$. O conectivo lógico '&' é chamado CONJUNÇÃO, ademais, A e B são chamados de CONJUNTOS.

Note que não tentamos simbolizar o 'também' da sentença 11. Palavras como 'ambos' e 'também' servem para chamar a atenção para o fato de que duas coisas estão sendo ligadas. Elas não tem nenhuma função lógica para além dessa, então não precisamos representá-las na LS.

Veja mais exemplos:

12. Bárbara é atlética e energética.
13. Bárbara e Roberto são ambos atléticos.
14. Apesar de Bárbara ser energética, ela não é atlética.
15. Bárbara é atlética, mas Roberto é mais atlético que ela.

A sentença 12 é obviamente uma conjunção. A sentença diz duas coisas sobre Bárbara, então em português é possível se referir a Bárbara apenas uma vez. Pode ser tentador fazer isso ao traduzir o argumento: Já que B significa 'Bárbara é atlética', alguém pode parafrasear a sentença por 'B e energética.' Isso seria um erro. Uma vez que traduzimos uma parte por 'B', qualquer estrutura posterior é perdida. B é uma sentença atômica, ela não é nada além de verdadeira ou falsa. Inversamente, 'energético' não é uma sentença; por si só não é nem verdadeiro, nem falso. Em vez disso, devemos parafrasear a sentença por 'B e Bárbara é energética.' Agora precisamos adicionar uma letra na chave de simbolização. Sendo E 'Bárbara é energética', então a sentença pode ser traduzida por $B\&E$.

> Uma sentença pode ser simbolizada por $\mathcal{A}\&\mathcal{B}$ se ela puder ser parafraseada no português por 'Ambos $\mathcal{A}$ e $\mathcal{B}$.' Cada um dos conjuntos deve ser uma sentença.

A sentença 13 diz uma coisa sobre dois sujeitos diferentes. Ela fala sobre ambos Bárbara e Roberto serem atléticos, e em português usamos a palavra 'atlético' apenas uma vez. Ao traduzir para a LS, é importante notar que a sentença pode ser parafraseada por 'Bárbara é atlética, e Roberto é atlético.' Isso se traduz por $B\&A$.

A sentença 14 é um pouco mais complicada. A expressão "Apesar de" cria um contraste entre a primeira e a segunda parte da sentença. Não obstante, a sentença diz, ao mesmo tempo, que Bárbara é energética e ela não é atlética. Para fazermos de cada um dos conjuntos uma sentença atômica, precisamos substituir ela por 'Bárbara.' Portanto, podemos parafrasear a sentença 14 por 'Bárbara é energética e Bárbara não é atlética.' O segundo conjunto contém uma negação, então a parafrasearemos por: 'Ambos: Bárbara é energética e não é o caso que Bárbara seja atlética.' Isso se traduz por $E\&\neg B$.

A sentença 15 contém uma estrutura contrastante similar. É irrelevante para o objetivo de se traduzir para a LS, então podemos parafrasear a sentença por 'Ambos Bárbara é atlética, e Roberto é mais atlético que Bárbara.' (Note que mais uma vez, substituímos o pronome 'ela' por seu nome próprio.) Como deveríamos traduzir o segundo conjunto? Nós já temos a letra sentencial A, que é sobre Roberto ser atlético, e a B, que é sobre Bárbara ser atlética, mas nenhuma delas significa um ser mais atlético que o outro. Precisamos de uma nova letra sentencial. Sendo R 'Roberto é mais atlético que Bárbara', assim, a sentença é traduzida por $B\&R$.

> Sentenças que podem ser parafrasedas por '$\mathcal{A}$, mas $\mathcal{B}$' ou por 'Embora $\mathcal{A}$, $\mathcal{B}$' são melhor simbolizadas, usando-se a conjunção $\mathcal{A}\&\mathcal{B}$.

É importante ter em mente que as letras sentenciais A, B e R são sentenças atômicas. Entendidas como símbolos da LS, elas não têm outro significado, além de verdadeiras ou falsas. Nós as utilizamos para simbolizar diferentes sentenças da língua portuguesa que são todas sobre pessoas serem atléticas, mas essa similaridade é completamente extinta quando as traduzimos para a LS. Nenhuma linguagem formal consegue capturar todas as estruturas da língua portuguesa, mas não há nada perdido ao não incluí-las todas, desde que essas estruturas não sejam importantes para o argumento.

Para quaisquer sentenças $\mathcal{A}$ e $\mathcal{B}$, $\mathcal{A}\&\mathcal{B}$ é verdadeiro se e somente se, ambos $\mathcal{A}$ e $\mathcal{B}$ forem verdadeiros. Podemos resumir isso em uma tabela de verdade específica a para a conjunção:

$\mathcal{A}$	$\mathcal{B}$	$\mathcal{A}\&\mathcal{B}$
V	V	V
V	F	F
F	V	F
F	F	F

A conjunção é simétrica, porque podemos permutar os conjuntos sem mudar o valor de verdade da sentença. Indiferentemente do que $\mathcal{A}$ e $\mathcal{B}$ sejam, $\mathcal{A}\&\mathcal{B}$ é logicamente equivalente a $\mathcal{B}\&\mathcal{A}$.

Disjunção

Considere estas sentenças:

16. Ou Denisson jogará golfe comigo ou irá assistir filmes.
17. Ou Denisson ou Helena jogará golfe comigo.

Para essas sentenças podemos usar a seguinte chave de simbolização:

D: Denisson jogará golfe comigo.

E: Helena jogará golfe comigo.

M: Denisson irá assistir filmes.

A sentença 16 é 'Ou D ou M.' Para simbolizá-la completamente, introduziremos um novo símbolo. A sentença se torna $D \vee M$. O conectivo '$\vee$' é chamado de DISJUNÇÃO, D e M são chamados de DISJUNTOS.

A sentença 17 é ligeiramente mais complicada. Existem dois sujeitos, mas a sentença portuguesa apresenta o verbo apenas uma vez. Ao traduzirmos, podemos parafraseá-la como 'Ou Denisson jogará golfe comigo ou Helena jogará golfe comigo.' Sendo assim, ela obviamente traduz $D \vee E$.

> Uma sentença pode ser simbolizada por $\mathcal{A} \vee \mathcal{B}$ se ela pode ser parafraseada, em português, por 'Ou $\mathcal{A}$ ou $\mathcal{B}$.' Cada um dos disjuntos deve ser uma sentença.

Algumas vezes em português a palavra 'ou' exclui a possibilidade de ambos os disjuntos serem verdadeiros. Esse é chamado de um OU EXCLUSIVO[2]. Por exemplo, num menu de restaurante um *ou exclusivo* é claramente pressuposto quando se diz "Entradas vêm com sopa ou salada". Você pode pedir sopa; você pode pedir salada; mas, se você deseja ambos, então você precisa pagar uma taxa extra.

Outras vezes, a palavra 'ou' permite a possibilidade de ambos os disjuntos serem verdadeiros. Esse é provavelmente o caso com a sentença 17, acima. Eu posso jogar com Denisson ou Helena ou ambos. A sentença 17 meramente diz que jogarei com *ao menos* um deles. Este é chamado de OU INCLUSIVO.[3]

O símbolo '$\vee$' representa um ou inclusivo. Desse modo, $D \vee E$ é verdadeira ou se D é verdadeira, ou se E é verdadeira, ou se ambos D e E são verdadeiras. É falso apenas se ambos D e E são falsas. Podemos sumarizar isso com a tabela de verdade específica para a disjunção:

$\mathcal{A}$	$\mathcal{B}$	$\mathcal{A} \vee \mathcal{B}$
V	V	V
V	F	V
F	V	V
F	F	F

[2]Também chamado de DISJUNÇÃO EXCLUSIVA (NT).
[3]Também chamado de DISJUNÇÃO INCLUSIVA (NT).

Assim como a conjunção, a disjunção é simétrica. $\mathcal{A} \vee \mathcal{B}$ é logicamente equivalente à $\mathcal{B} \vee \mathcal{A}$.

Estas sentenças são mais complicadas:

18. Você não terá sopa ou não terá salada.
19. Você não terá nem sopa nem salada.
20. Você terá ou sopa ou salada, mas não ambos.

Sendo 'você terá sopa' traduzido por S_1 e 'você terá salada' traduzido por S_2.

A sentença 18 pode ser parafraseada da seguinte maneira: '*Ou não é o caso que* você terá sopa ou *não é o caso que* você terá salada.' Traduzir isso requer tanto a disjunção como a negação, e isso se torna: $\neg S_1 \vee \neg S_2$.

A sentença 19 também requer uma negação. Ela pode ser parafraseada como '*Não é o caso que* ou você terá sopa ou você terá salada.' Precisamos indicar de alguma forma que a negação não nega apenas o disjunto da direita ou da esquerda, mas, pelo contrário, nega a disjunção inteira. Para fazermos isso, colocaremos parênteses ao redor da disjunção: 'Não é o caso que $(S_1 \vee S_2)$, que por sua vez se torna, $\neg(S_1 \vee S_2)$.

Note que os parênteses estão cumprindo um papel importante aqui. A sentença $\neg S_1 \vee S_2$ significaria ' Você não terá sopa ou terá salada.'

A sentença 20 se refere a um *ou exclusivo*. Podemos dividir a sentença em duas partes. A primeira diz que você terá uma ou a outra. Traduziremos por $(S_1 \vee S_2)$. A segunda parte diz que você não terá ambos. Podemos parafraseá-la por 'não é o caso que você tenha sopa e salada.' Usando tanto a negação quanto a conjunção, traduziremos por $\neg(S_1 \& S_2)$. Agora, só precisamos ligar as duas partes e, como vimos acima, 'mas' pode, geralmente, ser traduzido por uma conjunção. A sentença 20 pode então ser traduzida por $(S_1 \vee S_2) \& \neg(S_1 \& S_2)$. Na língua portuguesa, *costumamos* diferenciar as disjunções exclusivas das inclusivas através do uso dos 'ous.' Quando começamos uma sentença com 'X ou Y', na maioria das vezes, *pretendemos* enunciar uma disjunção inclusiva. Já ao iniciarmos outra sentença com 'Ou X ou Y' geralmente *pretendemos* enunciar uma disjunção exclusiva.

Apesar de '$\vee$' ser um *ou inclusivo*, podemos simbolizar um *ou exclusivo* na LS. Nós só precisamos de mais um conectivo para fazê-lo.[4]

Condicional

Para as seguintes sentenças, traduziremos 'Você cortará o fio vermelho' por C e 'a bomba irá explodir' por B.

21. Se você cortar o fio vermelho, a bomba irá explodir.

22. A bomba irá explodir apenas se você cortar o fio vermelho.

A sentença 21 pode ser traduzida parcialmente como 'Se C, então B.' Nós usaremos o símbolo '$\rightarrow$' para representar implicação lógica. Assim, a sentença se torna, $C \rightarrow B$. Esse conectivo é chamado de CONDICIONAL. A sentença do lado esquerda da condicional, C, nesse exemplo, é a ANTECEDENTE. A lado direito, B, é a CONSEQUENTE.

A sentença 22 é também uma condicional. Já que a palavra 'se' aparece na segunda metade da sentença, pode ser tentador simbolizá-la do mesmo modo que a sentença 21. Isso seria um erro.

A condicional $C \rightarrow B$ diz que se C *fosse verdade*, então B também o seria. Ela não diz que a bomba explodiria *apenas* se você cortasse o fio vermelho. Outra pessoa poderia cortar o fio ou essa poderia ser uma bomba relógio. A sentença $C \rightarrow B$ não diz nada sobre o que esperar se C for falso. A sentença 22 é diferente. Ela diz que as únicas condições sob as quais a bomba explodiria envolvem você cortando o fio vermelho, i.e., se a bomba explodiu, então você cortou o fio vermelho. Assim sendo, a sentença 22 deve ser simbolizada como $B \rightarrow C$.

É importante se lembrar que o conectivo '$\rightarrow$' diz apenas que, se a antecedente é verdadeira, então a consequente também é. Ela não diz nada sobre a conexão *causal* entre os dois eventos. Ao se traduzir a sentença 22 como $B \rightarrow C$, não se quer dizer que a explosão da bomba haveria de alguma maneira causado que você cortasse o fio. Ambas as sentenças 21 e 22 sugerem que, se você cortar o fio vermelho, seu corte seria a causa da bomba explodindo. Elas diferem quanto à conexão *lógica*. Se a sentença 22 fosse verdade, então uma

[4] Alguns lógicos optam por simbolizar a disjunção exclusiva pelo símbolo $\veebar$. Desse modo, a sentença 20 pode ser traduzida simplesmente por $S_1 \veebar S_2$. Cf., Irving M. Copi, *Introdução a Lógica*. Na lógica computacional, a mesma operação é simbolizada por $\oplus$(NT).

explosão nos diria — àqueles que estão a uma distância segura da bomba — que você tinha cortado o fio vermelho. Sem uma explosão, a sentença 22 não nos diz nada.

> A sentença parafraseada '$\mathcal{A}$ apenas se $\mathcal{B}$' é logicamente equivalente à 'Se $\mathcal{A}$, então $\mathcal{B}$.'

'Se $\mathcal{A}$, então $\mathcal{B}$' significa que se $\mathcal{A}$ fosse verdade, $\mathcal{B}$ também seria. Assim, sabemos que se a antecedente $\mathcal{A}$ é verdadeira e a consequente $\mathcal{B}$ é falsa, então a condicional 'Se $\mathcal{A}$, então $\mathcal{B}$'' é falsa. Mas qual o valor de verdade de 'Se $\mathcal{A}$, então $\mathcal{B}$'' sob outras circunstâncias? Suponha, por exemplo, que a antecedente $\mathcal{A}$ fosse falsa. Nesse caso, 'Se $\mathcal{A}$, então $\mathcal{B}$'' não nos diria nada sobre o valor de verdade da consequente $\mathcal{B}$ e não é claro qual o valor de verdade que 'Se $\mathcal{A}$, então $\mathcal{B}$'' teria.

Em português, a verdade das condicionais depende de qual seria o caso se a antecedente fosse verdade — mesmo se, factualmente, a antecedente é falsa. Isso levanta um problema ao se traduzir condicionais para LS. Consideradas como sentenças da LS, C e B não têm nada a ver uma com a outra. Para podermos pensar como o mundo seria se C fosse verdade, nós precisaríamos analisar o que C diz sobre o mundo. Entretanto, já que C é um símbolo atômico de LS, não há mais nada o que ser analisado. Quando substituímos uma sentença por uma letra sentencial, consideramo-la meramente como uma sentença atômica que pode ser verdadeira ou falsa.

Para traduzirmos condicionais para LS, não tentaremos capturar todas as sutilezas do português 'Se ..., então ...'. Ao invés disso, o símbolo '$\rightarrow$' será uma *condicional material*[5]. Isso significa que quando $\mathcal{A}$ é falsa, a condicional $\mathcal{A} \rightarrow \mathcal{B}$ será automaticamente verdadeira, independentemente do valor de $\mathcal{B}$. Se ambos $\mathcal{A}$ e $\mathcal{B}$ são verdadeiras, a condicional $\mathcal{A} \rightarrow \mathcal{B}$ será verdadeira. Ou seja, $\mathcal{A} \rightarrow \mathcal{B}$ é falsa se e somente se, $\mathcal{A}$ é verdadeira e $\mathcal{B}$ é falsa.

Podemos resumir isso com uma tabela de verdade específica para a condicional.

[5] Também chamada de implicação material (NT).

$\mathcal{A}$	$\mathcal{B}$	$\mathcal{A} \to \mathcal{B}$
V	V	V
V	F	F
F	V	V
F	F	V

A condicional é assimétrica. Você não pode permutar a antecedente com a consequente sem mudar o significado da sentença, porque $\mathcal{A} \to \mathcal{B}$ e $\mathcal{B} \to \mathcal{A}$ não são logicamente equivalentes.

Nem todas as sentenças da forma 'Se ..., então ...' são condicionais. Considere esta sentença:

23. Se alguém quiser me ver, estarei na varanda.

Se eu disser isso, quero comunicar que estarei na varanda, independentemente se alguém quiser me ver ou não — mas se alguém de fato quisesse me ver, então deveria me achar lá. Se traduzíssemos 'estarei na varanda' por P, então a sentença 23 seria simplesmente P.

Bicondicional

Consideremos as sentenças:

24. A figura no quadro é um triângulo apenas se tem exatamente três lados.
25. A figura no quadro é um triângulo se tem exatamente três lados.
26. A figura no quadro é um triângulo se e somente se, ela tem exatamente três lados.

Podemos traduzir 'A figura é um triângulo' por T e 'a figura tem três lados' por L.

A sentença 24, pelas razões discutidas acima, será traduzida por $T \to L$.

A sentença 25 é diferente. Ela pode ser parafraseada por, 'Se a figura tem três lados, então é um triângulo.' Ela pode ser traduzida por $L \to T$.

A sentença 26 diz que T é verdadeira se e somente se S é verdadeira; podemos inferir S de T e podemos inferir T de S. Isso é chamado de BICON-

DICIONAL, porque ela implica as duas condicionais $S \to T$ e $T \to S$. Nós utilizaremos '$\leftrightarrow$' para representar a bicondicional; assim, a sentença 26 pode ser traduzida por $S \leftrightarrow T$.

Poderíamos prosseguir sem um novo símbolo para a bicondicional. Já que a sentença 26 significa '$S \to T$ e $T \to S$', poderíamos traduzi-la como $(S \to T)\&(T \to S)$. Nós precisamos de parênteses para indicar que $(S \to T)$ e $(T \to S)$ são conjuntos separados; a expressão $S \to T\&T \to S$ seria ambígua.

Uma vez que poderíamos escrever $(\mathcal{A} \to \mathcal{B})\&(\mathcal{B} \to \mathcal{A})$, em vez de $\mathcal{A} \leftrightarrow \mathcal{B}$, estritamente falando, não precisaríamos introduzir um novo símbolo. Apesar disso, linguagens lógicas geralmente têm tal símbolo. Assim, a LS também o terá, uma vez que ele tornará a tradução de frases do tipo 'se e somente se' mais fácil.

$\mathcal{A} \leftrightarrow \mathcal{B}$ é verdadeira se e somente se $\mathcal{A}$ e $\mathcal{B}$ têm o mesmo valor. Esta é a tabela de verdade específica para a bicondicional:

$\mathcal{A}$	$\mathcal{B}$	$\mathcal{A} \leftrightarrow \mathcal{B}$
V	V	V
V	F	F
F	V	F
F	F	V

2.3 Outra simbolização

Já introduzimos todos os símbolos da LS. Podemos combiná-los para traduzir muitos tipos de sentenças. Considere esses exemplos de sentenças que usam o conectivo português 'a menos que':

27. A menos que você vista uma jaqueta, você pegará um resfriado.
28. Você pegará um resfriado a menos que vista uma jaqueta.

Traduzamos 'Você vestirá uma jaqueta' por J e 'você pegará um resfriado' por R.

Podemos parafrasear a sentença 27 como 'A menos que J, R.' Isso significa que se você não vestir a jaqueta, então você pegará um resfriado; com isso em mente, podemos traduzi-la por $\neg J \rightarrow R$. Mas isso também significa que se você não pegou um resfriado, então você vestiu uma jaqueta; e com isso em mente, podemos também traduzi-la por $\neg R \rightarrow J$.

Qual dessas traduções é a correta para a sentença 27? Ambas são, porque as duas traduções são equivalentes em LS.

A sentença 28, em português, é logicamente equivalente a 27. Ela pode ser traduzida como $\neg J \rightarrow R$ *ou* $\neg R \rightarrow J$.

Ao simbolizarmos sentenças como as 27 e 28, é fácil de nos confundirmos. Já que a condicional não é simétrica, seria errado traduzi-las por $J \rightarrow \neg R$. Felizmente, existem expressões lógicas equivalentes. Ambas as sentenças significam que: você vestirá uma jaqueta ou — se você não vesti-la —, você pegará um resfriado. Desse modo, podemos traduzi-la por $J \vee R$. (Você pode notar aqui que o 'ou' aqui deveria ser uma *disjunção exclusiva*. Entretanto, as sentenças não excluem a possiblidade que você pode vestir uma jaqueta e *ainda* assim pegar um resfriado; jaquetas não te protegem de todas as formas possíveis de você pegar um resfriado.)

Se uma sentença pode ser parafraseada por 'A menos que $\mathcal{A}$,$\mathcal{B}$,' então ela pode ser simbolizada por $\mathcal{A} \vee \mathcal{B}$.

A simbolização de tipos de sentenças padrões é resumida na p. 187.

2.4 Sentenças da LS

A sentença 'Maçãs são vermelhas ou jaboticabas são pretas' é uma sentença do português, e a sentença '$(A \vee B)$' é uma sentença da LS. Apesar de podermos identificar sentenças do português, quando as encontramos, não temos uma definição formal de 'sentença do português'. Na LS, é possível definir formalmente o que conta como sentença. Esse é um dos aspectos nos quais uma linguagem formal como a LS é mais precisa do que as línguas naturais, como o português.

É importante distinguir entre a linguagem lógica de LS, a qual estávamos desenvolvendo, e a linguagem que usamos para falar sobre a LS. Quando estamos falando sobre uma linguagem, a linguagem da qual estamos falando é chamada de LINGUAGEM OBJETO. A linguagem que usamos para falar sobre a linguagem objeto é chamada de METALINGUAGEM.

A linguagem objeto nesse capítulo é a LS. A metalinguagem é o português — não o português comum, mas o português complementado com algum vocabulário lógico e matemático. A sentença '$(A \vee B)$' é uma sentença na linguagem objeto, porque ela usa apenas os símbolos da LS. Entretanto, a palavra 'sentença' não é em si mesma parte da LS, então a sentença 'Essa expressão é uma sentença de LS' não é uma sentença de LS. Ela é uma sentença na metalinguagem, uma sentença que utilizamos para falar *sobre* a LS.

Nesta seção, daremos uma definição formal para uma 'sentença da LS.' A definição em si mesma será dada em português matemático, a metalinguagem.

Expressões

Existem três tipos de símbolos na LS:

Letras sentenciais com subscritos, quando necessário	$A, B, C \ldots, Z, A_1, B_1 Z_1, A_2, A_{25}, J_{375}, \ldots$
Conectivos	$\neg, \&, \vee, \rightarrow . \leftrightarrow$
Parênteses	(,)

Nós definiremos uma EXPRESSÃO DE LS como sendo qualquer sequência de símbolos de LS. Escolha quaisquer dos símbolos da LS e escreva-os, em qualquer ordem, e você terá uma expressão.

Fórmula bem formada

Já que qualquer sequência de símbolos é uma expressão, muitas expressões da LS serão confusas ou desprovidas de sentido. Uma expressão provida de sentido é chamada de *fórmula bem formada*. É comum usar o acrônimo *fbf*; o plural é fbfs.

Obviamente, letras sentenciais individuais como A e G_{13} serão fbfs. Podemos formar mais fbfs a partir dessas, utilizando os conectivos. Usando a negação, teremos $\neg A$ e $\neg G_{13}$. Utilizando a conjunção, podemos ter $A \& G_{13}$, $G_{13} \& A$, $A \& A$ e $G_{13} \& G_{13}$. Poderíamos também aplicar a negação repetidamente para termos fbfs como $\neg\neg A$ ou aplicar a negação junto a conjunções para ter fbfs como $\neg(A \& G_{13})$ e $\neg(G_{13} \& \neg G_{13})$. As combinações possíveis são intermináveis, até mesmo começando com apenas essas duas letras sentenciais e existem infinitas letras sentenciais. Então não há sentido em tentar listar todas as fbfs.

Ao invés disso, descreveremos o processo pelo qual todas as fbfs são formadas. Tomemos a negação: Dada qualquer fbf $\mathcal{A}$ da LS, $\neg\mathcal{A}$ é uma fbf. É digno de nota aqui que, $\mathcal{A}$ não é uma letra sentencial. Efetivamente, ela é uma variável que pode ser substituída por qualquer fbf. Note que essa variável $\mathcal{A}$ não é um símbolo da LS, então $\neg\mathcal{A}$ não é uma expressão da LS. Ao invés disso, ela é uma expressão da metalinguagem que nos permite falar sobre muitas expressões da LS: todas as que começarem com o símbolo de negação. Por $\mathcal{A}$ ser parte da metalinguagem, ela é chamada de metavariável.

Podemos dizer coisas similares para cada um dos outros conectivos. Por exemplo, se $\mathcal{A}$ e $\mathcal{B}$ são fbfs da LS, então $(\mathcal{A}\&\mathcal{B})$ é uma fbf da LS. Ao criar frases como essas para todos os conectivos, chegamos a essa definição para uma fórmula bem formada de LS:

1. Toda sentença atômica é uma fbf.
2. Se $\mathcal{A}$ é uma fbf, então $\neg\mathcal{A}$ é uma fbf da LS.
3. Se $\mathcal{A}$ e $\mathcal{B}$ são fbfs, então $(\mathcal{A}\&\mathcal{B})$ é uma fbf.
4. Se $\mathcal{A}$ e $\mathcal{B}$ são fbfs, então $(\mathcal{A} \vee \mathcal{B})$ é uma fbf.
5. Se $\mathcal{A}$ e $\mathcal{B}$ são fbfs, então $(\mathcal{A} \rightarrow \mathcal{B})$ é uma fbf.
6. Se $\mathcal{A}$ e $\mathcal{B}$ são fbfs, então $(\mathcal{A} \leftrightarrow \mathcal{B})$ é uma fbf.
7. Todas e apenas as fbfs de LS podem ser geradas através da aplicação dessas regras.

Note que não podemos aplicar imediatamente essa definição para saber se uma expressão arbitrária é uma fbf. Suponha que queiramos saber se $\neg\neg\neg D$ é ou não uma fbf da LS. Olhando para a segunda cláusula da definição, sabemos que $\neg\neg\neg D$ é uma fbf se $\neg\neg D$ é uma fbf. Então, agora precisamos saber se $\neg\neg D$ é ou não uma fbf. E de novo, olhando na segunda cláusula da definição, $\neg\neg D$ será uma fbf se $\neg D$ for uma. Mais uma vez, $\neg D$ será uma fbf se D for uma fbf. Finalmente, D é uma letra sentencial, uma sentença atômica da LS, então sabemos que D é uma fbf por causa da primeira cláusula da definição. Dessa maneira, para uma fórmula composta, como $\neg\neg\neg D$, precisamos aplicar a definição repetidamente e eventualmente chegaremos nas sentenças atômicas nas quais a fbf foi construída.

Definições desse tipo são chamadas de *recursivas*. Definições recursivas começam com alguns elementos básicos especificados e definem formas de (re)combiná-los indefinidamente. Assim como a definição recursiva permite que sentenças complexas sejam construídas a partir de partes simples, você pode usá-la para decompor as sentenças em suas partes mais simples. Para determinar se algo atende ou não a definição, você pode precisar consultá-la de novo várias vezes.

Ao decompor uma sentença, o conectivo que você deve olhar primeiro é chamado de OPERADOR LÓGICO PRINCIPAL daquela sentença. Por exemplo: o operador lógico principal de $\neg(E \vee (F \rightarrow G))$ é a negação, $\neg$. O operador lógico principal de $(\neg E \vee (F \rightarrow G))$ é a disjunção, $\vee$.

Sentenças

Lembre-se que a sentença é uma expressão provida de significado que pode ser verdadeira ou falsa. Já que as expressões de LS que possuem significado são as *fbfs* e já que toda *fbf* é ou verdadeira ou falsa, a definição de LS para sentença é igual a definição de *fbf*. Todavia, nem toda linguagem formal terá esse belo recurso. Na linguagem da LQ, que será posteriormente desenvolvida neste livro, existem *fbfs* que não são sentenças.

A estrutura recursiva das sentenças na LS será importante quando considerarmos sob quais circunstâncias uma sentença em particular seria verdadeira ou falsa. A sentença $\neg\neg\neg D$ será verdadeira se e somente se, a sentença $\neg\neg D$ for falsa, e dessa maneira, prosseguiríamos até chegarmos aos em seus componentes atômicos: $\neg\neg\neg D$ será verdade se e somente se, a sentença atômica D é falsa. Retornaremos a esse ponto no próximo capítulo.

Convenções de notação

Uma *fbf* do tipo $(Q\&R)$ deve ser cercada por parênteses, porque podemos aplicar novamente a definição para usá-la como parte de uma sentença mais complicada. Se negarmos $(Q\&R)$, teremos $\neg(Q\&R)$. Se tivéssemos simplesmente $Q\&R$ sem parênteses e colocássemos uma negação em sua frente, teríamos $\neg Q\&R$. O mais natural nesse caso seria lê-la como $(\neg Q\&R)$, algo muito diferente de $\neg(Q\&R)$. A sentença $\neg(Q\&R)$ significa que não é o caso que ambos Q e R sejam verdadeiros; Q ou R pode ser falso, mas a sentença não nos diz qual. A sentença $(\neg Q\&R)$ significa especificamente que Q é falso e que R é verdadeiro. Portanto, os parênteses são cruciais para o significado da sentença.

Então, estritamente falando, $Q\&R$ sem parênteses *não* é uma sentença de LS. Porém, quando usarmos a LS, afrouxaremos um pouco essa definição precisa para facilitar nosso trabalho. Faremos isso de diversas maneiras.

Primeiramente, entenderemos que $Q\&R$ significará o mesmo que $(Q\&R)$. Por mera convenção, não colocaremos esses parênteses que ocorrem ao *redor da sentença inteira*.

Em segundo lugar, algumas vezes pode ser confuso olhar para longas sentenças com muitos parênteses próximos. Adotaremos a convenção de utilizar colchetes '[' e ']' no lugar dos parênteses. Por exemplo, não há diferença lógica entre $(P \vee Q)$ e $[P \vee Q]$. A sentença

$$(((H \to I) \vee (I \to H))\&(J \vee K))$$

pode ser escrita assim

$$[(H \to I) \vee (I \to H)]\&(J \vee K).$$

Em terceiro lugar, algumas vezes traduziremos a conjunção de três ou mais sentenças. Para a sentença, 'Alice, Bob e Cláudia foram à festa', suponhamos que A signifique 'Alice foi à festa', B 'Bob foi à festa' e C 'Cláudia foi à festa.' A definição só nos permite formar a conjunção a partir de duas sentenças, então a traduziremos por $(A\&B)\&C$ ou $A\&(B\&C)$. Não há motivo para distinguilas, já que ambas as traduções são logicamente equivalentes. Não há diferença lógica entre a primeira, na qual $(A\&B)$ é conjunta com C e a segunda, na qual A é conjunta com $(B\&C)$. Sendo assim, podemos escrever simplesmente $A\&B\&C$. Por mera convenção, podemos não colocar parênteses aos conjuntar três ou mais sentenças.

Em quarto lugar, uma situação semelhante ocorre com disjunções múltiplas. 'Alice, Bob ou Cláudia irão à festa' pode ser traduzida por $(A \vee B) \vee C$ ou por $A \vee (B \vee C)$, e já que essas traduções são logicamente equivalentes, podemos escrever $A \vee B \vee C$.

Essas últimas duas convenções só se aplicam a múltiplas conjunções ou disjunções. Se uma série de conectivos incluem ambas disjunções e conjunções, então os parênteses são essenciais; por exemplo, $(A\&B) \vee C$ e $A\&(B \vee C)$. Os parênteses também são necessários se houver uma série de condicionais ou de bicondicionais; por exemplo, $(A \rightarrow B) \rightarrow C$ e $(A \leftrightarrow B) \leftrightarrow C$.

Nós adotamos essas quatro regras como convenções de notação, não como mudanças nas definições de uma sentença. Estritamente falando, $A \vee B \vee C$ ainda não é uma sentença. Ao invés disso, ela é um tipo de simplificação. Podemos escrevê-la por conveniência, mas o que realmente queremos dizer é $((A \vee B) \vee C)$.

Se tivéssemos dado uma definição diferente de *fbf*, então essas seriam *fbfs*. Poderíamos ter escrito a regra 3 deste modo: "Se $\mathcal{A}, \mathcal{B}, \ldots, \mathcal{Z}$ são fbfs, então $(\mathcal{A} \& \mathcal{B} \& \ldots \& \mathcal{Z})$ é uma *fbf*." Isso tornaria a tradução de algumas sentenças do português mais fácil, todavia a custo de tornar nossa linguagem mais complicada. Nós precisaríamos manter essa definição complexa em mente ao desenvolver tabelas de verdade e um sistema de prova. Queremos uma linguagem lógica que seja *expressivamente* simples, nos permitindo traduzir facilmente do português, mas que também seja *formalmente* simples. Adotar convenções de notação é um compromisso entre esses dois desejos.

2.5 Exercícios práticos

$\star$ **Parte A** Utilizando a chave de simbolização dada, traduza cada sentenças do português para a LS.

M: Aquelas criaturas são homens fantasiados.

C: Aquelas criaturas são chimpanzés.

G: Aquelas criaturas são gorilas.

1. Aquelas criaturas não são homens fantasiados.
2. Aqueles criaturas são homens fantasiados ou não.
3. Aquelas criaturas são gorilas ou chimpanzés.
4. Aquelas criaturas nem são gorilas nem chimpanzés.
5. Se aquelas criaturas são chimpanzés, então elas nem são gorilas nem são homens fantasiados.

6. A menos que aquelas criaturas sejam homens fantasiados, elaes são chimpanzés ou elas são gorilas.

Parte B Utilizando a chave de simbolização dada, traduza cada sentença do português para a LS.

A: Sr. Gonzaga foi assassinado.

B: O mordomo fez isso.

C: o cozinheiro fez isso.

D: Karolina está mentindo.

E: Sr. Jackson foi assassinado.

F: A arma do assassinato foi uma frigideira.

1. Ou Sr. Gonzaga ou Sr. Jackson foram assassinados.
2. Se o Sr. Gonzaga foi assassinado, então o cozinheiro fez isso.
3. Se o Sr. Jackson foi assassinado, então o cozinheiro não fez isso.
4. Ou o mordomo fez isso ou Karolina está mentindo.
5. O cozinheiro fez isso somente se Karolina está mentindo.
6. Se a arma do assassinato foi uma frigideira, então o culpado deve ser o cozinheiro.
7. Se a arma do assassinato não foi uma frigideira, então o culpado foi o cozinheiro ou o mordomo.
8. Sr. Gonzaga foi assassinado se, e somente se, Sr. Jackson não foi assassinado.
9. Karolina está mentindo, a menos que Sr. Jackson tenha sido de fato assassinado.
10. Se Sr. Gonzaga foi assassinado, ele o foi com uma frigideira.
11. Já que o cozinheiro fez isso, o mordomo não o fez.
12. É claro que Karolina está mentindo!

⋆ **Parte C** Utilizando a chave de simbolização dada, traduza cada sentença do português para a LS.

E1: Maria é uma eletricista.

E2: José é um eletricista.

F1: Maria é uma bombeira.

F2: José é um bombeiro.

S1: Maria está satisfeita com sua carreira.

S2: José está satisfeito com sua carreira

1. Maria e José são eletricistas.
2. Se Maria é uma bombeira, então ela está satisfeita com sua carreira.
3. Maria é uma eletricista, a menos que seja uma bombeira.
4. José é um eletricista insatisfeito.
5. Nem Maria nem José são eletricistas.
6. Ambos Maria e José são eletricistas, mas nenhum deles acha isso satisfatório.
7. José está satisfeito apenas se ele é um bombeiro.
8. Se Maria não é uma eletricista, então nem José é, mas se ela for, ele também será.
9. Maria está satisfeita com sua carreira se, e somente se, José não está satisfeito com sua carreira.
10. Se José é um eletricista e um bombeiro, então ele deve estar satisfeito com seu trabalho.
11. Não pode ser o caso que José seja um eletricista e um bombeiro.
12. José e Maria são ambos bombeiros se, e somente se, nenhum dos dois é eletricista.

⋆ **Parte D** Crie uma chave de simbolização e simbolize as seguintes sentenças em LS.

1. Alice e a Duquesa são ambas espiãs.
2. Se Alice ou a Duquesa são espiãs, então o código foi decifrado.
3. Se nem Alice nem a Duquesa são espiãs, então o código permanece secreto.
4. A embaixada alemã estará uma confusão, a menos que alguém tenha decifrado o código.
5. Ou o código foi decifrado ou não, contudo, independentemente disso, a embaixada alemã estará uma confusão.
6. Ou Alice ou a Duquesa são espiãs, mas não ambas.

Parte E Crie uma chave de simbolização e simbolize as seguintes sentenças em LS.

1. Se César jogar no gol, o time perderá.
2. O time perderá a menos que haja um milagre.
3. O time perderá ou não, contudo, independetemente disso, César jogará no gol.
4. A mãe de César fará biscoitos se, e somente se, ele jogar no gol.
5. Se houver um milagre, então a mãe de César não fará biscoitos.

Parte F Para cada argumento, escreva uma chave de simbolização e traduza o argumento da melhor forma possível para LS.

1. Se Dora toca piano pela manhã, então Mario acorda irritado. Dora toca piano pela manhã a menos que esteja ocupada. Assim, se Mario não acordar irritado, então Dora deve ter se ocupado.
2. Ou choverá ou fará sol na terça-feira. Se chover, Alana ficará triste. Se fizer sol, Alana ficará com calor. Portanto, Alana ficará ou triste ou com calor na terça.
3. Se Daniel se lembrou de fazer as tarefas domésticas, então as coisas estarão limpas, mas não organizadas. Se ele não se lembrou, então as coisas estarão organizadas, mas não limpas. Por conseguinte, as coisas estão ou organizadas ou limpas – mas não ambas.

⋆ **Parte G** Para cada uma das sentenças seguintes: (a) ela é uma fbf de LS? (b) Ela é uma sentença de LS que permite as notações convencionais serem aplicadas?

1. (A)
2. $J_{374} \vee \neg J_{374}$
3. $\neg\neg\neg\neg F$
4. $\neg \& S$
5. $(G \& \neg G)$
6. $\mathcal{A} \rightarrow \mathcal{A}$
7. $(A \rightarrow (A \& \neg F)) \vee (D \leftrightarrow E)$
8. $[(Z \leftrightarrow S) \rightarrow W] \& [J \vee X]$
9. $(F \leftrightarrow \neg D \rightarrow J) \vee (C \& D)$

Parte H

1. Existe alguma fbf de LS que não contém letra sentencial alguma? Por que sim ou por que não?
2. Neste capítulo, simbolizamos um *ou exclusivo* utilizando $\vee, \&$ e $\neg$. Como você poderia traduzir um *ou exclusivo* utilizando apenas dois conectivos? Existe alguma forma de traduzi-lo usando apenas um conectivo?

Capítulo 3

Tabelas de verdade

Este capítulo introduz uma forma de avaliar sentenças e argumentos da LS. Apesar de ser trabalhoso, o método da tabela de verdade é um procedimento puramente mecânico e não requer nenhuma intuição ou percepção acurada.

3.1 Conectivos verofuncionais

Qualquer sentença não atômica na LS é composta de sentenças atômicas com conectivos sentenciais. O valor de verdade de sentenças compostas depende apenas dos valores de verdade das sentenças atômicas que as compõem. Por exemplo, para saber o valor de verdade de $(D \leftrightarrow E)$, você precisa saber apenas dos valores de verdade de D e E. Conectivos que funcionam dessa maneira são chamados de VEROFUNCIONAIS.

Neste capítulo, faremos uso do fato de que todos os operadores lógicos da LS são verofuncionais — isso torna possível construirmos tabelas de verdade para sabermos os atributos lógicos de uma sentença. Contudo, você deve notar que isso não é possível em todas as linguagens. No português, é possível formar uma nova sentença a partir de qualquer sentença $\mathcal{X}$ mais simples ao dizermos 'É possível que $\mathcal{X}$.' O valor de verdade dessa nova sentença não depende completamente no valor de verdade de $\mathcal{X}$. Até mesmo se $\mathcal{X}$ fosse falsa, talvez, em algum sentido, $\mathcal{X}$ poderia ter sido verdadeira — então a nova sentença seria verdadeira. Na lógica modal, poderíamos traduzir 'É possível que $\mathcal{X}$' por $\Diamond\mathcal{X}$. Todavia, a habilidade de traduzir sentenças como essas vem a um preço: O operador $\Diamond$ não é verofuncional, portanto, lógicas modais não estão sujeitas a tabelas de verdade.

$\mathcal{A}$	$\neg\mathcal{A}$
1	0
0	1

$\mathcal{A}$	$\mathcal{B}$	$\mathcal{A}\&\mathcal{B}$	$\mathcal{A}\vee\mathcal{B}$	$\mathcal{A}\to\mathcal{B}$	$\mathcal{A}\leftrightarrow\mathcal{B}$
1	1	1	1	1	1
1	0	0	1	0	0
0	1	0	1	1	0
0	0	0	0	1	1

Tabela 3.1: As tabelas de verdades específicas para os conectivos da LS.

3.2 Tabelas de verdade completas

O valor de verdade de sentenças que contém apenas um conectivo é dado pela tabela de verdade específica daquele conectivo. No capítulo anterior, escrevemos as tabelas de verdade com 'V' para verdadeiro e 'F' para falso. Todavia, é importante notar que isso não diz respeito a uma verdade em qualquer sentido profundo ou cósmico. Poetas e filósofos podem discutir acerca da natureza e significado da verdade, mas as verofuncões na LS são apenas regras que produzem resultados a partir dos valores introduzidos. Para enfatizar esse aspecto, neste capítulo escreveremos '1' e '0' em vez de 'V' e 'F'. Apesar de interpretarmos '1' como significando 'verdadeiro' e '0' como falso, computadores podem ser programados para completar tabelas de verdades de maneira totalmente mecânica. Em uma máquina '1' pode significar que um registro é acessado e '0' que o mesmo registro é fechado. Matematicamente, eles são os dois únicos valores possíveis que uma sentença da LS pode ter. As tabelas de verdade para os conectivos da LS, escrita em termos de 1s e 0s, são dadas na tabela 3.1.

Por exemplo, a tabela de verdade específica da conjunção dá as condições de verdade para qualquer sentença da forma $(\mathcal{A}\&\mathcal{B})$. Mesmo que os conjuntos $\mathcal{A}$ e $\mathcal{B}$ sejam formados por sentenças longas e complicadas, a conjunção será verdadeira, se e somente se, ambos $\mathcal{A}$ e $\mathcal{B}$ forem verdadeiros. Consideremos a sentença $(H\&I) \to H$. Faremos todas as combinações possíveis de verdadeiro e falso para H e I, o que nos dará quatro linhas. Então, copiaremos os valores de verdade para as letras sentenciais e escrevê-los-emos abaixo das letras sentenciais.

H	I	$(H$	$\&$	$I)$	$\to$	H
1	1	**1**		**1**		**1**
1	0	**1**		**0**		**1**
0	1	**0**		**1**		**0**
0	0	**0**		**0**		**0**

Agora, consideremos a subsentença $H\&I$. Essa é uma conjunção $\mathcal{A}\&\mathcal{B}$, com H como $\mathcal{A}$ e I como $\mathcal{B}$. H e I são verdadeiros na primeira linha, e já que uma conjunção é verdadeira quando ambos os conjuntos também o são, escreveremos 1 abaixo do símbolo da conjunção. Procederemos assim nas três outras linhas e obteremos essa tabela:

H	I	$(H$	$\&$	$I)$	$\to$	H
		$\mathcal{A}$	$\&$	$\mathcal{B}$		
1	1	1	**1**	1		1
1	0	1	**0**	0		1
0	1	0	**0**	1		0
0	0	0	**0**	0		0

A sentença inteira é uma condicional $\mathcal{A} \to \mathcal{B}$ com $(H\&I)$ como $\mathcal{A}$ e H como $\mathcal{B}$. Por exemplo, na segunda linha $(H\&I)$ é falso e H é verdadeiro e já que uma condicional é verdadeira quando a antecedente é falsa, escreveremos 1 na segunda linha abaixo do símbolo condicional. Procederemos assim nas três outras linhas e obteremos essa tabela:

H	I	$(H$	$\&$	$I)$	$\to$	H
			$\mathcal{A}$		$\to$	$\mathcal{B}$
1	1		1		**1**	1
1	0		0		**1**	1
0	1		0		**1**	0
0	0		0		**1**	0

A coluna de 1s abaixo da condicional diz-nos que a sentença $(H\&I) \to I$ é verdadeira independentemente dos valores de verdade de H e I. Elas podem ser verdadeiras ou falsas em qualquer combinação e ainda assim, a sentença composta será verdadeira. É crucial que consideramos todas as combinações possíveis, pois se tivéssemos uma tabela de verdade de apenas duas linhas, não poderíamos ter certeza de que a sentença não seria falsa em alguma outra combinação dos valores de verdade.

Nesse exemplo, não repetimos todos os registros da tabela de verdade. Todavia, quando estivermos escrevendo tabelas de verdade no papel, é impraticável apagar colunas inteiras ou reescrever tabelas inteiras em cada etapa. Apesar de mais cheia, a tabela de verdade pode ser escrita desta maneira:

H	I	$(H$	$\&$	$I)$	$\to$	H
1	1	1	1	1	1	1
1	0	1	0	0	1	1
0	1	0	0	1	1	0
0	0	0	0	0	1	0

A maioria das colunas abaixo das sentenças estão lá por mera questão estilística do livro. Quando você se tornar mais experiente com tabelas de verdade, provavelmente não precisará mais escrever colunas para cada uma das letras sentenciais. De qualquer forma, em cada linha, o valor de verdade da sentença é dado pela coluna abaixo do operador lógico principal da mesma; nesse caso, dado pela coluna abaixo da condicional.

UMA TABELA DE VERDADE COMPLETA tem uma linha para todas as combinações possíveis de 1 e 0 para todas as letras sentenciais. O tamanho de uma tabela de verdade completa depende do número de letras sentenciais diferentes na mesma. Uma sentença que contém apenas uma sentença requer apenas duas linhas, como na tabela de verdade específica da negação. Isso será verdade mesmo se a mesma letra for repetida várias vezes, como na sentença $[(C \leftrightarrow C) \to C]\&\neg(C \to C)$. A tabela de verdade completa necessita de apenas duas linhas, porque existem apenas duas possibilidades: C pode ser verdadeira, ou pode ser falsa. Uma única letra sentencial não pode ser marcada com 1 e 0 na mesma linha. A tabela de verdade dessa sentença é:

C	$[(C$	$\leftrightarrow$	$C)$	$\to$	$C]$	$\&$	$\neg$	$(C$	$\to$	$C)$
1	1	1	1	1	1	**0**	0	1	1	1
0	0	1	0	0	0	**0**	0	0	1	0

Olhando a coluna abaixo do conectivo principal, vemos que a sentença é falsa nas duas linhas da tabela; i.e., ela é falsa independe se C é verdadeira ou falsa.

Uma sentença que contém duas letras sentenciais necessita de quatro linhas para uma tabela de verdade completa, como na tabela de verdade específica de

$(H\&I) \rightarrow H$. Já uma sentença que contém três letras sentenciais necessita de oito linhas. Por exemplo:

M	N	P	M	$\&$	$(N$	$\vee$	$P)$
1	1	1	1	**1**	1	1	1
1	1	0	1	**1**	1	1	0
1	0	1	1	**1**	0	1	1
1	0	0	1	**0**	0	0	0
0	1	1	0	**0**	1	1	1
0	1	0	0	**0**	1	1	0
0	0	1	0	**0**	0	1	1
0	0	0	0	**0**	0	0	0

A partir dessa tabela, sabemos que a sentença $M\&(N \vee P)$ pode ser verdadeira ou falsa, dependendo dos valores de verdade de M, N e P.

Uma tabela de verdade completa para uma sentença que possui quatro letras sentenciais diferentes necessita de 16 linhas. Cinco letras, 32 linhas, seis letras, 64 linhas etc. Em linhas gerais: se uma tabela de verdade completa tem n letras sentenciais diferentes, então ela deve ter 2^n linhas.

Para completar as colunas de uma tabela de verdade, comece com a letra sentencial mais à direita e alterne 1s e 0s. Na próxima coluna à esquerda, escreva dois 1s e dois 0s, e repita o processo. E para a terceira letra sentencial, escreva quatro 1s seguidos de quatro 0s. Assim, você fará uma tabela de verdade de oito linhas como aquela acima.

Para uma tabela de verdade de 16 linhas, a próxima coluna de letras sentenciais deve ter oito 1s seguidos de oito 0s. Por uma tabela de 32 linhas, a próxima linha precisaria ter 16 1s seguidos de 16 0s, e etc.

3.3 Utilizando tabelas de verdade

Tautologias, contradições e sentenças contingentes

Lembre-se que uma sentença em português é uma tautologia se ela tem de ser verdade somente em função de sua forma lógica. Com uma tabela de verdade completa, consideramos todas as possibilidades as quais o mundo pode ser. Se a sentença é verdadeira em todas as linhas de uma tabela de verdade completa, então ela é logicamente verdadeira, independentemente de como o mundo é.

Logo, uma sentença é uma TAUTOLOGIA NA LS se a coluna abaixo de seu operador principal é 1 em toda linha de uma tabela de verdade completa.

Pelo contrário, uma sentença é uma CONTRADIÇÃO NA LS se a coluna abaixo de seu principal é 0 em toda linha de uma tabela de verdade completa.

Finalmente, uma sentença é CONTINGENTE NA LS se não é nem uma tautologia, nem uma contradição; i.e., se é 1 em ao menos uma linha e 0 em ao menos uma outra linha.

Das tabelas de verdade na seção anterior, sabemos que $(H\&I) \rightarrow H$ é uma tautologia, que $[(C \leftrightarrow C) \rightarrow C]\&\neg(C \rightarrow C)$ é uma contradição e que $M\&(N \vee P)$ é contingente.

Equivalência lógica

Em português, duas sentenças são logicamente equivalentes se elas possuem o mesmo valor de verdade no que diz respeito à lógica. Mais uma vez, tabelas de verdade nos permitem definir um conceito análogo para a LS: duas sentenças são LOGICAMENTE EQUIVALENTES NA LS se elas tiverem o mesmo valor de verdade, em todas as linhas de uma tabela de verdade completa, em seu conectivo lógico principal.

Considere as sentenças $\neg(A \vee B)$ e $\neg A\&\neg B$. Elas são logicamente equivalentes? Para descobrir, construiremos uma tabela de verdade:

A	B	$\neg$	$(A$	$\vee$	$B)$	$\neg$	A	$\&$	$\neg$	B
1	1	**0**	1	1	1	0	1	**0**	0	1
1	0	**0**	1	1	0	0	1	**0**	1	0
0	1	**0**	0	1	1	1	0	**0**	0	1
0	0	**1**	0	0	0	1	0	**1**	1	0

Olhe para as colunas dos operadores principais dos conectivos principais; a negação para a primeira sentença, e a conjunção para a segunda. Nas três primeiras linhas, ambas são 0, já na linha final, ambas são 1. E dado que elas se igualam em todas as linhas, as duas sentenças são logicamente equivalentes.

Consistência

Em português, um conjunto de sentenças é consistente se é logicamente possível que elas sejam todas verdadeiras ao mesmo tempo. Um conjunto de sentenças é LOGICAMENTE CONSISTENTE NA LS se existe ao menos uma linha de uma tabela de verdade completa na qual todas as sentenças sejam verdadeiras. De outra forma, ele será INCONSISTENTE.

Validade

Em português, um argumento é válido se é logicamente impossível que suas premissas sejam, ao mesmo tempo, verdadeiras e sua conclusão falsa. Por sua vez, um argumento é VÁLIDO NA LS se não houver uma linha, de uma tabela de verdade completa, na qual as premissas são todas 1 e a conclusão é 0; um argumento é INVÁLIDO NA LS se houver tal linha.

Considere o argumento:

$$\neg L \rightarrow (J \vee L)$$

$$\neg L$$

$$\therefore\ J$$

Ele é válido? Para descobrirmos, construiremos a tabela de verdade.

J	L	$\neg$	L	$\rightarrow$	$(J$	$\vee$	$L)$	$\neg$	L	J
1	1	0	1	**1**	1	1	1	**0**	1	**1**
1	0	1	0	**1**	1	1	0	**1**	0	**1**
0	1	0	1	**1**	0	1	1	**0**	1	**0**
0	0	1	0	**0**	0	0	0	**1**	0	**0**

Sim, o argumento é válido. A única linha na qual ambas as premissas são 1 é a segunda, e na mesma, a conclusão também o é.

3.4 Tabelas de verdade parciais

Para mostrarmos que uma sentença é uma tautologia, precisamos mostrar que 1 está em todas as linhas e, para isso, precisaremos de uma tabela de verdade completa. Todavia, para mostrar que uma sentença *não* é uma tautologia, nós precisamos de apenas uma linha: uma na qual a sentença é 0. Portanto, para mostrarmos que algo não é uma tautologia, é suficiente fazermos uma *tabela de verdade parcial* de apenas uma linha — independentemente de quantas letras sentenciais uma sentença possa ter.

Considere, por exemplo, a sentença $(U\&T) \to (S\&W)$. Queremos mostrar que ela não é uma tautologia através de uma de uma tabela de verdade parcial. Completaremos 0 para a sentença inteira. O conectivo principal da sentença é o condicional. Para que essa condicional seja falsa, a antecedente deve ser verdadeira (1) e a consequente deve ser falsa (0). Portanto, as colocaremos na tabela:

S	T	U	W	$(U$	$\&$	$T)$	$\to$	$(S$	$\&$	$W)$
					1		**0**		0	

Para que $(U \ \& \ T)$ seja verdadeira, ambos U e T devem ser.

S	T	U	W	$(U$	$\&$	$T)$	$\to$	$(S$	$\&$	$W)$
	1	1		1	1	1	**0**		0	

Agora, só precisamos fazer $(S\&W)$ falsa. Para isso, precisamos fazer pelo menos um de S ou W falso. Podemos fazer ambos falsos se quisermos. O que interessa aqui é que toda a sentença se torne falsa nesta linha. Tomando uma decisão arbitrária, terminamos a tabela de verdade assim:

S	T	U	W	$(U$	$\&$	$T)$	$\to$	$(S$	$\&$	$W)$
0	1	1	0	1	1	1	**0**	0	0	0

Para mostrar que algo é uma contradição, é necessária uma tabela de verdade completa. Para mostrar que algo *não* é uma contradição, é necessária apenas uma tabela de verdade parcial de uma linha, na qual a sentença é verdadeira.

Uma sentença é contingente se ela não é nem uma tautologia, nem uma contradição. Então, para mostrar que uma sentença é contingente, é necessária uma tabela de verdade parcial de duas linhas: a sentença deve ser verdadeira em uma linha e falsa na outra. Por exemplo, podemos mostrar que a sentença acima é contingente com a seguinte tabela de verdade:

S	T	U	W	$(U$	$\&$	$T)$	$\rightarrow$	$(S$	$\&$	$W)$
0	1	1	0	1	1	1	**0**	0	0	0
0	1	0	0	0	0	1	**1**	0	0	0

	SIM	NÃO
tautologia?	tabela de verdade completa	tabela de verdade parcial de duas linhas
contradição?	tabela de verdade completa	tabela de verdade parcial de duas linhas
contingente?	tabela de verdade parcial de duas linhas	tabela de verdade completa
equivalente?	tabela de verdade completa	tabela de verdade parcial de duas linhas
consistente?	tabela de verdade parcial de uma linha	tabela de verdade completa
válido?	tabela de verdade completa	tabela de verdade parcial de duas linhas

Tabela 3.2: Você precisa de uma tabela de verdade completa ou parcial? Depende do que você quer provar.

Note que existem muitas combinações de valores de verdade que fariam a sentença verdadeira, portanto, existem muitas maneiras possíveis de escrever a segunda linha.

Para provar que uma sentença *não* é contingente, é necessário fazer uma tabela de verdade completa, porque deve-se mostrar que ela é uma tautologia ou uma contradição. Se você não sabe se uma sentença particular é contingente ou não, então você não sabe se precisará de uma tabela de verdade completa ou parcial. Você pode sempre começar a análise por uma tabela de verdade completa. Se as linhas já feitas mostrarem que a sentença é contingente, você não precisa terminá-la, caso contrário, complete a tabela. Apesar de duas linhas cuidadosamente selecionadas mostrarem que uma sentença contingente é, de fato, contingente, não há nada errado em completar as outras linhas

Para provar que duas sentenças são logicamente equivalentes, é necessário fazer uma tabela de verdade completa. Para provar que elas não são logicamente equivalentes, é necessário apenas uma tabela de verdade parcial de uma linha: faça-a de modo que uma sentença seja falsa e a outra verdadeira.

Para provar que um conjunto de sentenças é consistente, é necessário que se tenha uma linha na tabela de verdade na qual todas as sentenças são verdadeiras. O resto da tabela é irrelevante, então uma tabela de verdade parcial de uma linha é suficiente. Para provar que um conjunto de sentenças é inconsistente, ao contrário, é necessário uma tabela de verdade completa: você deve mostrar que em todas as linhas pelo menos uma das sentenças é falsa.

Para provar que um argumento é válido, é necessária uma tabela de verdade completa. Para provar que um argumento é *inválido*, é necessária apenas uma tabela de verdade de uma linha: se você conseguir fazer uma linha na qual todas as premissas são verdadeiras e a conclusão é falsa, o argumento é inválido.

A tabela 3.2 resume as situações nas quais uma tabela de verdade completa é necessária e quando uma parcial é suficiente.

3.5 Exercícios práticos

Se você quiser exercícios extras, você pode construir tabelas de verdades para quaisquer das sentenças e argumentos dos exercícios do capítulo anterior.

$\star$ **Parte A** Determine se cada sentença é tautológica, contraditória ou contingente. Justifique sua resposta com uma tabela de verdade parcial ou completa.

1. $A \to A$
2. $\neg B \& B$
3. $\neg A \to A$
4. $\neg D \vee D$
5. $(A \leftrightarrow B) \leftrightarrow \neg(A \leftrightarrow \neg B)$
6. $(A \& B) \vee (B \& A)$
7. $(A \to B) \vee (B \to A)$
8. $\neg[A \to (B \to A)]$
9. $(A \& B) \to (B \vee A)$
10. $A \leftrightarrow [A \to (B \& \neg B)]$

11. $\neg(A \vee B) \leftrightarrow (\neg A \& \neg B)$
12. $\neg(A \& B) \leftrightarrow A$
13. $[(A \& B) \& \neg(A \& B)] \& C$
14. $A \rightarrow (B \vee C)$
15. $[(A \& B) \& C] \rightarrow B]$
16. $(A \& \neg A) \rightarrow (B \vee C)$
17. $\neg[(C \vee A) \vee B]$
18. $(B \& D) \leftrightarrow [A \leftrightarrow (A \vee C)]$

⋆ **Parte B** Determine se cada par de sentenças é logicamente equivalente. Justifique sua resposta com uma tabela de verdade parcial ou completa.

1. $A, \neg A$
2. $A, A \vee A$
3. $A \rightarrow A, A \leftrightarrow A$
4. $A \vee \neg B, A \rightarrow B$
5. $A \& \neg A, \neg B \leftrightarrow B$
6. $\neg(A \& B), \neg A \vee \neg B$
7. $\neg(A \rightarrow B), \neg A \rightarrow \neg B$
8. $(A \rightarrow B), (\neg B \rightarrow \neg A)$
9. $[(A \vee B) \vee C], [A \vee (B \vee C)]$
10. $[(A \vee B) \& C], [A \vee (B \& C)]$

⋆ **Parte C** Determine se cada conjunto de sentenças é consistente ou inconsistente. Justifique sua resposta com uma tabela de verdade parcial ou completa.

1. $A \rightarrow A, \neg A \rightarrow \neg A, A \& A, A \vee A$
2. $A \& B, C \rightarrow \neg B, C$
3. $A \vee B, A \rightarrow C, B \rightarrow C$

4. $A \to B, B \to C, A, \neg C$
5. $B\&(C \vee A), A \to B, \neg(B \vee C)$
6. $A \vee B, B \vee C, C \to \neg A$
7. $A \leftrightarrow (B \vee C), C \to \neg A, A \to \neg B$
8. $A, B, C, \neg D, \neg E, F$

$\star$ **Parte D** Determine se cada argumento é válido ou inválido. Justifique sua resposta com uma tabela de verdade parcial ou completa.

1. $A \to A, \therefore A$
2. $A \vee [A \to (A \leftrightarrow A)], \therefore A$
3. $A \to (A\&\neg A), \therefore \neg A$
4. $A \leftrightarrow \neg(B \leftrightarrow A), \therefore A$
5. $A \vee (B \to A), \therefore \neg A \to \neg B$
6. $A \to B, B, \therefore A$
7. $A \vee B, B \vee C, \neg A, \therefore B\&C$
8. $A \vee B, B \vee C, \neg B, \therefore A\&C$
9. $(B\&A) \to C, (C\&A) \to B, \therefore (C\&B) \to A$
10. $A \leftrightarrow B, B \leftrightarrow C, \therefore A \leftrightarrow C$

$\star$ **Parte E** Responda cada uma das questões abaixo e justifique sua resposta.

1. Suponha que $\mathcal{A}$ e $\mathcal{B}$ são logicamente equivalentes. O que você pode dizer sobre $\mathcal{A} \leftrightarrow \mathcal{B}$?
2. Suponha que $(\mathcal{A}\&\mathcal{B}) \to \mathcal{C}$ é contingente. O que você pode dizer sobre o argumento"$\mathcal{A}$,$\mathcal{B}$.$\therefore$ $\mathcal{C}$" ?
3. Suponha que $\{\mathcal{A}, \mathcal{B}, \mathcal{C}\}$ é inconsistente. O que você pode dizer sobre $(\mathcal{A}\&\mathcal{B}\&\mathcal{C})$?
4. Suponha que $\mathcal{A}$ é uma contradição. O que você pode dizer sobre o argumento "$\mathcal{A}$,$\mathcal{B}$.$\therefore$ $\mathcal{C}$"?

5. Suponha que $\mathcal{C}$ é uma tautologia. O que você pode dizer sobre "$\mathcal{A}$,$\mathcal{B}$.$\therefore$ $\mathcal{C}$" ?

6. Suponha que $\mathcal{A}$ e $\mathcal{B}$ são logicamente equivalentes. O que você pode dizer sobre $(\mathcal{A} \vee \mathcal{B})$?

7. Suponha que $\mathcal{A}$ e $\mathcal{B}$ *não* são logicamente equivalentes. O que você pode dizer sobre $(\mathcal{A} \vee \mathcal{B})$?

Parte F Poderíamos deixar o bicondicional ($\leftrightarrow$) fora de nossa linguagem. Se fizéssemos isso, ainda poderíamos escrever $'A \leftrightarrow B'$ de forma a tornar as sentenças mais fáceis de ler, mas isso seria uma abreviação para $(A \rightarrow B)\&(B \rightarrow A)$.A língua resultante seria formalmente equivalente à LS, já que $'A \leftrightarrow B$ e $(A \rightarrow B)\&(B \rightarrow A)$ são logicamente equivalentes em LS. Se valorizássemos a simplicidade formal em vez da riqueza expressiva, poderíamos substituir mais conectivos por convenções notacionais e ainda teríamos uma língua equivalente à LS.

Há muitas linguagens formais equivalentes que utilizam apenas dois conectivos. Basta possuirmos apenas a negação e o condicional material. Demonstre isso escrevendo sentenças que sejam logicamente equivalentes a cada uma das seguintes utilizando apenas parênteses, letras sentenciais, negação ($\neg$) e o condicional material ($\rightarrow$).

⋆1. $A \vee B$

⋆2. $A\&B$

⋆3. $A \leftrightarrow B$

Poderíamos ter uma língua que fosse equivalente à LS com apenas a negação e a disjunção como conectivos. Demonstre isso: utilizando apenas parênteses, letras sentenciais negação ($\neg$) e disjunção ($\vee$), escreva sentenças que sejam logicamente equivalentes a cada uma das seguintes.

4. $A\&B$

5. $A \rightarrow B$

6. $A \leftrightarrow B$

A *barra de Sheffer* é um conectivo lógico com a seguinte tabela de verdade:

$\mathcal{A}$	$\mathcal{B}$	$\mathcal{A} \mid \mathcal{B}$
1	1	0
1	0	1
0	1	1
0	0	1

7. Escreva uma sentença usando os conectivos da LS que seja logicamente equivalente a $(A \mid B)$.

Toda sentença escrita utilizando um conectivo da LS pode ser reescrita por uma sentença logicamente equivalente utilizando uma ou mais barras de Sheffer. Utilizando apenas a barra de Sheffer, escreva sentenças que sejam logicamente equivalentes a cada uma das seguintes.

8. $\neg A$
9. $(A \& B)$
10. $(A \vee B)$
11 $(A \rightarrow B)$
12. $(A \leftrightarrow B)$

Capítulo 4

Lógica Quantificada

Este capítulo introduz uma linguagem lógica chamada LQ. Ela é uma versão de lógica quantificada, pois permite quantificadores como *todo* e *algum*. A lógica quantificada também é chamada de *lógica de predicados*, porque as unidades básicas da linguagem são predicados e termos.

4.1 Das sentenças aos predicados

Considere o seguinte argumento que obviamente é válido em português:

> Se todo mundo sabe lógica, então ninguém ficará confuso ou todos ficarão. Todos ficarão confusos somente se tentarmos acreditar em uma contradição. Essa é uma aula de lógica, então todos sabem lógica.
> $\therefore$ Se não tentarmos acreditar em uma contradição, ninguém ficará confuso.

Para simbolizarmos isso em LS, precisaremos de uma simbolização.

L: Todo mundo sabe lógica.

N: Ninguém ficará confuso.

T: Todos ficarão confusos.

A: Tentarmos acreditar em uma contradição.

Note que N e T são ambos sobre pessoas ficando confusas, mas são duas letras sentenciais diferentes. Não podemos substituir T por $\neg N$. Por que não? $\neg N$ significa 'não é o caso que ninguém ficará confuso.' Esse seria o caso mesmo se uma pessoa ficasse confusa, portanto, é muito diferente de *todos* ficarão confusos.

Uma vez que temos letras sentenciais diferentes para N e T, entretanto, impedimos qualquer conexão entre as duas. Elas são apenas duas sentenças atômicas que podem ser individualmente verdadeira ou falsa. Em português, nunca poderia ser o caso de ninguém e todos estarem confusos. Porém, enquanto sentenças de LS existe um valor de verdade para o qual N e T são ambas verdadeiras.

Expressões do tipo 'ninguém', 'todos' e 'qualquer um' são chamadas de quantificadores. Ao traduzir N e T por sentenças atômicas, nós perdemos a *estrutura de quantificação* da sentença. Felizmente, essa estrutura não é o que faz esse argumento ser válido. Dessa forma, podemos ignorá-la. Para ver isso, traduziremos o argumento para LS:

$L \to (N \vee T)$

$T \to B$

L

$\therefore\ \neg B \to N$

Esse argumento é válido. (Você pode fazer uma tabela de verdade para conferir.)

Agora, considere outro argumento. Esse também é válido em português.

William é um lógico. Todos lógicos vestem chapéus engraçados.

1. $\therefore$ William veste chapéus engraçados.

Para simbolizar isso, definiremos a simbolização:

L: William é um lógico.

T: Todos lógicos vestem chapéus engraçados.

E: William veste chapéus engraçados.

Agora, simbolizaremos o argumento:

L

T

$\therefore E$

Esse é inválido na LS. (Mais uma vez, você pode conferir em uma tabela de verdade.) Há algo errado aqui, porque obviamente esse é um argumento válido em português. A simbolização em LS perde toda estrutura importante. Mais uma vez, a tradução em LS ignora a estrutura de quantificação: a sentença 'Todos lógicos vestem chapéus engraçados' é sobre lógicos e vestir chapéus. Ao não traduzir essa estrutura, perdemos a ligação entre William ser um lógico e William vestir chapéu.

Alguns argumentos com estruturas de quantificação podem ser traduzidos na LS, como no primeiro exemplo, mesmo que a LS ignore essa estrutura. Outros argumentos são completamente perdidos na LS, como no segundo exemplo. Note que o problema não é que erramos enquanto simbolizávamos o segundo argumento. Essas são as melhores simbolizações que podemos dar a esses argumentos em LS.

Geralmente, se um argumento contendo quantificadores é *válido* na LS, então o argumento em português também será. Se ele for *inválido* na LS, então não podemos dizer que o argumento em português será inválido. O argumento pode ser válido por causa da estrutura de quantificação que ele tem em português e que lhe falta na LS.

Similarmente, se uma sentença com quantificadores for uma *tautologia* em LS, então a sentença em português é logicamente verdadeira. Se ela for *contingente* em LS, pode ser por causa da estrutura de quantificação que é removida quando a traduzimos para a linguagem formal.

Para simbolizarmos argumentos que possuem estrutura de quantificação, precisamos desenvolver uma linguagem lógica diferente. Chamaremos de lógica quantificada, LQ.

4.2 Blocos de construção da LQ

Assim como sentenças foram as unidades básicas da lógica sentencial, predicados serão as unidades básicas da lógica quantificada. Um predicado é uma expressão do tipo 'é um cachorro.' Isso não é uma sentença por si. Não é verdadeira nem falsa. Para ser verdadeiro ou falso, precisamos especificar algo: quem ou o que é um cachorro?

Os detalhes serão explicados no resto do capítulo, mas aqui está a ideia básica: na LQ, representaremos predicados com letras maiúsculas. Por exemplo, podemos traduzir '______ é um cachorro' por D. Usaremos as letras minúsculas para nomes de coisas específicas. Por exemplo, podemos traduzir 'Bertie' por b. A expressão Db será uma sentença em LQ. Essa é a tradução de 'Bertie é um cachorro.'

Para podermos representar a estrutura quantificada, também teremos símbolos que representarão quantificadores. Por exemplo, '$\exists$' significará 'Há algum ______.' Então para dizermos há um cachorro, escreveremos $\exists x Dx$; ou seja, há um x tal que x é cachorro.

Comecemos por definir os termos singulares e predicados.

Termos singulares

Em português, um TERMO SINGULAR é uma palavra ou frase que se refere a uma pessoa, lugar ou coisa *específicos*. A palavra 'cachorro' não é um termo singular, porque existem muitos cachorros. A frase 'o cahorro de Felipe, Bertie' é um termo singular, porque se refere a um terrier específico.

Um NOME PRÓPRIO é um termo singular que se refere a um indivíduo sem descrevê-lo. O nome 'Emerson' é um nome próprio, mas apenas o nome não lhe diz nada sobre Emerson. Claro, alguns nomes são tradicionalmente dados a homens e outros a mulheres. Se 'Dani' é usado como um termo singular, você pode pensar que ele se refere a um homem. Entretanto, o nome não significa necessariamente que a pessoa referida é um homem — nem mesmo que a criatura referida é uma pessoa. Dani pode ser uma girafa que você poderia apenas chamá-la assim. Há muito debate filosófico acerca desse problema, mas o ponto importante aqui é que um nome é um termo singular, porque se refere a um único indivíduo específico.

Outros termos singulares indicam a informação do objeto ao qual se referem mais obviamente. Por exemplo, você pode dizer, sem outras informações adicionais, que 'o cachorro de Felipe, Bertie' é um termo singular que se refere a um cachorro. Uma DESCRIÇÃO DEFINIDA indica um indivíduo através de uma definição única. Em português, descrições definidas são comumente frases da forma 'que é tal e tal.' Elas se referem a coisa específica que se encaixa na descrição dada. Por exemplo, 'o maior integrante de Monty Python' e 'o primeiro imperador da China' são descrições definidas. Uma descrição que não se refere a um indivíduo específico não é uma descrição definida. 'Um integrante do Monty Python' e 'um imperador da China' não são descrições definidas.

Podemos utilizar nomes próprios e descrições definidas para indicar a mesma coisa. O nome próprio 'Nova Iorque' nomeia o local indicado pela definição de 'cidade mais populosa dos E.U.A.' As expressões se referem ao mesmo local de formas diferentes. Você não apreende nada se eu disser que vou para Nova Iorque, a menos que saiba um pouco de geografia. Você poderia pensar que é uma cidade nos E.U.A., mas não teria certeza.

Em português, a especificação de um termo singular pode depender do contexto; 'William' significa uma pessoa específica e não qualquer um chamado William; 'P.D. Magnus,' como um termo lógico singular, se refere a mim, não a outro P.D. Magnus. Nós temos esse tipo de ambiguidade no português, mas é importante ter em mente que termos singulares em LQ devem se referir apenas a uma coisa específica.

Na LQ, simbolizaremos termos singulares com letras minúsculas de a até w. Podemos adicionar subscritos se quisermos utilizar alguma letra mais que uma vez. Portanto, $a, b, c, \ldots, w, a_1, f_{32}, j_{390}$ e m_{12} são todos termos em LQ.

Termos singulares são chamados de CONSTANTES, porque se referem a indivíduos específicos. Note que x, y e z não são constantes em LQ. Elas serão variáveis, letras que nada indicam de específico. Precisaremos delas para introduzirmos os quantificadores.

Predicados

Os predicados mais simples são propriedades individuais. Existem coisas que você pode falar sobre um objeto. '______ é um cachorro' e '______ é um integrante do Monty Python' são ambos predicados. Ao traduzir sentenças do português, o termo nem sempre virá no início da sentença: 'o piano caiu no

______' é também predicado. Predicados como esses são chamados de unários, ou MONÁDICOS, porque há apenas um espaço em branco para ser completado. Um predicado unário e um termo singular combinam-se para fazer uma sentença.

Outros predicados são sobre *relação* entre duas coisas. Por exemplo, '______ é maior que ______', '______ está à esquerda de ______' e '______ deve dinheiro a ______'. Esses são predicados binários, ou DIÁDICOS, porque eles precisam ser completados com dois termos para fazer uma sentença.

Em geral, você pode pensar sobre predicados como sentenças esquemáticas que precisam ser completadas com algum número de termos. Inversamente, você pode começar com sentenças e fazer predicados a partir delas por se retirar termos. Considere a sentença, 'Vinícius pegou emprestado o carro familiar de César.' Ao retirarmos um termo singular, podemos reconhecer essa sentença como usando qualquer um de três predicados monádicos diferentes:

______ pegou emprestado o carro familiar de César.

Vinícius pegou emprestado ______ de César.

Vinícius pegou emprestado o carro familiar de ______.

Ao se remover dois termos singulares, podemos reconhecer três predicados diádicos diferentes:

Vinícius pegou emprestado ______ de ______.

______ pegou emprestado o carro familiar de ______.

______ pegou emprestado ______ de César.

Ao se remover três termos singulares, podemos reconhecer um predicado ternário, ou triádico:

______ pegou emprestado ______ de ______.

Se estivermos traduzindo essa sentença para a LQ, deveremos traduzi-la com um predicado unário, binário ou ternário? Depende do que pretendemos poder dizer. Se a única coisa que discutiremos que está sendo emprestada é o carro, então a generalidade do predicado ternário é desnecessária. Se a única substituição que precisamos simbolizar for pessoas diferentes pegando emprestado o carro familiar de César, então um predicado unário será suficiente.

Em geral, podemos ter predicados com tantos lugares quanto precisemos. Predicados com mais de um lugar vazio são chamados de POLIÁDICOS. Predicados com n lugares, para algum número n, são chamados de N-ÁRIOS ou N-ÁDICOS.

Na LQ, simbolizamos predicados com letras maiúsculas de A até Z, com ou sem subscritos. Quando damos uma chave de simbolização para predicados, não utilizaremos espaços; em vez deles, utilizaremos variáveis. Convencionalmente, as constantes são listadas ao final da chave. Então, podemos escrever uma chave que parece com isto:

Rx: x está com raiva.

Fx: x está feliz.

$\mathbf{A_1}$xy: x é tão alto quanto ou mais alto que y.

$\mathbf{A_2xy}$: x é tão ágil quanto ou mais ágil que y.

Exyz: y está entre x e z.

d: Donald

g: Geraldo

m: Maria

Podemos simbolizar sentenças que usam qualquer combinação desses predicados e termos. Por exemplo:

1. Donald está com raiva.
2. Se Donald está com raiva, assim estão Geraldo e Maria.
3. Maria é no mínimo tão alta e tão ágil quanto Geraldo.
4. Donald é mais baixo que Geraldo.

5. Geraldo está entre Donald e Maria.

A sentença 1 é direta: Rd. O 'x' no chave do termo 'Rx' está apenas marcando a posição; podemos substituí-lo por outros termos ao traduzirmos.

A sentença 2 pode ser parafraseada por: 'Se Rd, então Rg e Rm.' A LQ tem todos os conectivos verofuncionais da LS, então podemos traduzir isso por $Rd \to (Rg \& Rm)$.

A sentença 3 pode ser traduzida por A_1mg & A_2mg.

A sentença 4 poderia ser interpretada como se ela precisasse de um novo predicado. Se precisássemos simbolizar a sentença, poderíamos definir um predicado como Bxy singificar 'x é mais baixo que y.' Entretanto, isso ignoraria a conexão lógica entre 'mais baixo' e 'mais alto.' Considerados apenas como símbolos de LQ, não há conexão entre B e T_1. De fato, eles podem significar qualquer coisa. Ao invés de introduzirmos um novo predicado, parafrasearemos a sentença 4 usando predicados já presentes na nossa chave: 'não é o caso que Donald seja tão ou mais alto que Geraldo.' Podemos traduzi-la por A_1dg.

A sentença 5 requer que prestemos muita atenção na ordem dos termos na chave de simbolização. Ela se torna $Edgm$.

4.3 Quantificadores

Agora podemos introduzir os quantificadores. Considere estas sentenças:

6. Todos estão felizes
7. Todo mundo é no mínimo tão ágil quanto Donald.
8. Alguém está com raiva.

Pode ser tentador traduzir a sentença 6 por $Fd \& Fg \& Fm$. Todavia, isso diria apenas que Donald, Geraldo e Maria estão felizes. Queremos dizer que todos estão felizes, mesmo se não tivermos definido um nome constante a eles. Para fazermos isso, introduziremos o símbolo '$\forall$'. Ele é chamado de QUANTIFICADOR UNIVERSAL.

Um quantificador deve sempre vir seguido por uma variável e uma fórmula que contenha essa variável. Podemos traduzir a sentença 6 por $\forall xFx$. Parafraseando em português, significa 'Para todo x, x é feliz'. Chamamos $\forall x$ de um *quantificador-x*. A fórmula que se segue ao quantificador é chamada de *escopo* do quantificador. Posteriormente daremos uma definição formal do escopo, mas intuitivamente essa é a parte da sentença a qual o quantificador se aplica. Em $\forall xFx$, o escopo do quantificador universal é Fx.

A sentença 7 pode ser parafraseada por 'Para todo x, x é no mínimo tão ágil quanto Donald.' É traduzida por $\forall xA_2xd$.

Nessas sentenças quantificadas, a variável x está servindo como um tipo de marcador de posição. A expressão $\forall x$ significa que você pode selecionar qualquer indivíduo e colocá-los no lugar de x. Não há razão especial para usar x em detrimento de outra variável. A sentença $\forall xFx$ significa exatamente o mesmo que $\forall yFy$, $\forall zFz$ e $\forall x_5Fx_5$.

Para traduzirmos a sentença 8, introduziremos outro símbolo: o QUANTIFICADOR EXISTENCIAL, '$\exists$'. Como o quantificador universal, o quantificador existencial necessita de uma variável. A sentença 8 pode ser traduzida por $\exists xRx$. Isso significa que há um x que está com raiva. Mais precisamente, isso significa que há pelo menos uma pessoa com raiva. Mais uma vez, a variável é um tipo de marcador de posição; poderíamos também ter traduzido a sentença 8 por $\exists zRz$.

Considere as seguintes sentenças:

9. Ninguém está com raiva.

10 Há alguém que não está feliz.

11. Nem todos estão felizes.

A sentença 9 pode ser parafraseada por 'Não é o caso que alguém esteja com raiva.' Isso pode ser traduzido utilizando-se a negação e um quantificador existencial: $\neg\exists xRx$. Outra possibilidade é parafrasear a sentença 9 por 'Todos não estão com raiva.' Com isso em mente, podemos traduzi-la utilizando a negação e um quantificador universal: $\forall x\neg Rx$. Ambas essas traduções são aceitáveis, pois são logicamente equivalentes. A questão crítica é se a negação vem antes ou depois do quantificador.

Em geral, $\forall x\mathcal{A}$ é logicamente equivalente a $\neg\exists x\neg\mathcal{A}$. Isso significa que qualquer sentença que possa ser simbolizada com um quantificador universal pode ser simbolizada com um quantificador existencial e vice-versa. Uma tradução pode parecer mais natural que a outra, mas não há diferença lógica ao traduzir-se com um quantificador ao invés do outro. Para algumas sentenças, isso será simplesmente uma questão de gosto.

A sentença 10 é mais naturalmente parafraseada por 'Há algum x tal que x não é feliz.' Isso se torna $\exists x\neg Fx$. Equivalentemente poderíamos escrever $\neg\forall xFx$.

A sentença 11 é mais naturalmente traduzida por $\neg\forall xFx$. Isso é logicamente equivalente a sentença 10, então poderia ser traduzida por $\exists x\neg Fx$.

Apesar de termos dois quantificadores na LQ, poderíamos ter uma linguagem formal equivalente com um único quantificador. Por exemplo, poderíamos proceder apenas com o quantificador universal e tratar do quantificador existencial como uma mera convenção notacional. Utilizamos colchetes [] para fazer sentenças mais legíveis, mas sabemos que esses são apenas parênteses (). Igualmente poderíamos escrever '$\exists x$' sabendo que isso é apenas uma abreviação de '$\neg\forall x\neg$.' Há uma escolha entre fazer a lógica formalmente simples e fazê-la expressivamente simples. Com a LQ, optamos pela simplicidade expressiva. Ambos $\forall$ e $\exists$ serão símbolos da LQ.

Universo de Discurso

Dada a chave de simbolização que vínhamos utilizando, $\forall xFx$ significa que 'Todos estão felizes.' Quem está incluído nesse *todos*? Quando utilizamos sentenças como essas em português, geralmente não significamos todos aqueles que estão vivos agora na terra. Nós certamente não significamos todos aqueles que já viveram ou que viverão na terra. Queremos significar algo mais modesto: todos no prédio, todos na sala ou todos no quarto.

Para eliminarmos essa ambiguidade, precisaremos especificar um UNIVERSO DE DISCURSO — abreviado por UD. O UD é o conjunto de coisas sobre as quais estamos falando sobre. Então se quisermos falar sobre as pessoas em Belo Horizonte, definiremos o UD como sendo as pessoas em Belo Horizonte. Escrevemos isso no início da chave de simbolização, assim:

UD: pessoas em Belo Horizonte.

Os quantificadores *abrangem* o universo de discurso. Dado esse UD, $\forall x$ significa 'Todos em Belo Horizonte.' E $\exists x$ significa 'Alguém em Belo Horizonte.' Cada constante nomeia algum membro do UD, então só podemos utilizar esse UD com a chave de simbolização acima se Donald, Geraldo e Maria estiverem todos em Belo Horizonte. Se quisermos falar sobre pessoas em outros lugares além de Belo Horizonte, então precisaremos incluir essas pessoas no UD.

Na LQ, o UD deve ser *não-vazio*; i.e., ele deve incluir no mínimo uma coisa. É possível construir linguagens formais que permitem UDs vazios, mas isso introduz complicações.

Até mesmo se permitirmos um UD com apenas um membro pode-se produzir resultados estranhos. Suponhamos ter essa chave de simbolização:

UD: A Torre Eiffel.

Px: x está em Paris.

A sentença $\forall xPx$ pode ser parafraseada em português por 'Tudo está em Paris.' Todavia, isso seria um engano. Isso significa que tudo *no UD* está em Paris. Esse UD contém apenas a Torre Eiffel, então com a chave simbolização $\forall xPx$ só significaria que a Torre Eiffel está em Paris.

Termos não-referenciais

Na LQ, cada constante individual deve selecionar exatamente um membro do UD. Uma constante não pode se referir a mais do que uma coisa só — é um termo *singular*. Ademais, cada constante deve selecionar *algo*. Isso diz respeito a um problema filosófico clássico: o problema dos termos não-referenciais.

Filósofos medievais geralmente usavam sentenças sobre a quimera para exemplificar esse problema. A quimera é um monstro mitológico, ela não existe realmente. Considere as duas sentenças:

12. A quimera está com raiva.
13. A quimera não está com raiva.

É tentador definir uma constante para significar 'quimera.' A chave de simbolização se pareceria com isto:

UD: criaturas na Terra.

Rx: x está com raiva.

q: quimera

Poderíamos traduzir a sentença 12 por Rq e a 13 por $\neg Rq$. Todavia, problemas aparecerão ao perguntarmos se essas sentenças são verdadeiras ou falsas.

Uma opção é dizer que a sentença 12 não é verdadeira, porque não existe quimera alguma. Se a sentença 12 é falsa por falar de uma coisa não-existente, então a sentença 13 será falsa pela mesma razão. Porém, isso significaria que Rq e $\neg Rq$ seriam ambas falsas. Dada as condições de verdade para negação, esse não pode ser o caso.

Já que não podemos dizer que elas são ambas falsas, o que devemos fazer? Outra opção é dizer que a sentença 12 é *sem sentido*, pois fala sobre uma coisa não existente. Assim, Rq seria uma expressão com sentido na LQ para algumas interpretações, mas não para outras. Todavia, isso tornaria nossa linguagem formal sujeita a interpretações particulares. Já que estamos interessados na forma lógica, queremos considerar a força lógica de uma sentença como Rq independentemente de qualquer interpretação particular. Se Rq fosse algumas vezes com sentido e algumas vezes sem sentido, não poderíamos fazer isso.

Eis *o problema dos termos não referenciais*, retornaremos a ele depois (cf. p. 88). Por agora, *é necessário* que cada constante da LQ se refira a algo no UD, apesar do UD poder ser um conjunto de qualquer coisa que quisermos. Se quisermos simbolizar argumentos sobre criaturas mitológicas, então devemos criar um UD que as inclua. Essa opção é importante se quisermos considerar a lógica de histórias ficcionais. Podemos traduzir uma sentença como 'Sherlock Holmes morava na rua Baker, 221B' ao incluirmos personagens ficcionais como Sherlock Holmes em nosso UD.

4.4 Traduzindo para LQ

Agora temos todas as partes de uma LQ. A tradução de sentenças mais complicadas será apenas uma questão de saber a maneira correta de combinar quantificadores, predicados, constantes, conectivos e variáveis. Considere as sentenças:

14. Todas moedas do meu bolso são de 25 centavos.
15. Alguma moeda sobre a mesa é de 10 centavos.
16. Nem todas as moedas sobre a mesa são de 10 centavos.
17. Nenhuma das moedas do meu bolso é de 10 centavos.

Ao provermos uma chave de simbolização, precisaremos de especificar um UD. Já que estamos falando acerca de moedas no meu bolso e sobre a mesa, o UD deve, no mínimo, conter todas essas moedas. Já que não estamos falando sobre coisa alguma exceto moedas, faremos o UD ser todas as moedas. Já que não estamos falando sobre alguma moeda específica, também não precisaremos definir constante alguma. Definiremos essa chave desse modo:

UD: todas as moedas

Bx: x está no meu bolso.

Mx: x está na mesa.

Vx: x é 25 centavos.

Dx: x é 10 centavos.

A sentença 14 será mais naturalmente traduzida por um quantificador universal. O quantificador universal diz algo sobre tudo no UD, não apenas sobre algumas moedas no meu bolso. Essa sentença significa que (para qualquer moeda) se esta moeda está em meu bolso, então é de 25 centavos. Então, podemos traduzi-la por $\forall x(Bx \rightarrow Vx)$.

Já que a sentença 14 é sobre moedas que estão no meu bolso *e* são de 25 centavos, pode parecer tentador traduzi-la utilizando uma conjunção. Entretanto, a sentença $\forall x(Bx \& Vx)$ significaria que tudo no UD está no meu bolso e é de 25 centavos. Seria uma loucura dizer isso e, não obstante, tal sentença significa algo totalmente diferente da sentença 14.

A sentença 15 será mais naturalmente traduzida por um quantificador existencial. Ela diz que há uma moeda que está sobre a mesa e é de 10 centavos. Desse modo, podemos traduzi-la por $\exists x(Mx \& Dx)$.

Note que precisamos utilizar uma condicional com um quantificador universal, mas uma conjunção com um quantificador existencial. O que significaria escrever $\exists x(Mx \rightarrow Dx)$? Provavelmente não o que você pensa. Isso significa que há um membro do UD o qual satisfaria a subfórmula; grosseiramente falando que há algum a tal que $(Ma \rightarrow Da)$ é verdadeiro. Na LS, $\mathcal{A} \rightarrow \mathcal{B}$ é logicamente equivalente a $\neg\mathcal{A} \vee \mathcal{B}$ e isso também o será na LQ. Assim, se $\exists x(Mx \rightarrow Dx)$ é verdadeiro, se há algum a tal que $(\neg Ma \vee Da)$; i.e., é verdadeiro se alguma moeda não está sobre a mesa ou é de 10 centavos. Claro que há uma moeda que não está sobre a mesa — há moedas em muitos outros lugares. Desse modo, $\exists x(Mx \rightarrow Dx)$ é trivialmente verdadeira. Uma condicional será geralmente o conectivo natural para se utilizar com um quantificador universal, mas uma condicional dentro do escopo de um quantificador existencial pode implicar coisas muito estranhas. Via de regra, não coloque condicionais no escopo de quantificadores existenciais a menos que você tenha certeza de que precisa de um.

A sentença 16 pode ser parafraseada por 'Não é o caso que exista alguma moeda de 10 centavos no meu bolso.' Isso pode ser traduzido por $\neg\exists x(Bx \& Dx)$. Ela também pode ser parafraseada por 'Tudo no meu bolso não é uma moeda de 10 centavos' e então poderia ser traduzida por $\forall x(Bx \rightarrow \neg Dx)$. Mais uma vez, as sentenças são logicamente equivalentes. Ambas são traduções corretas da sentença 17.

Agora podemos traduzir o argumento da p. 58, o qual motivou a necessidade do uso de quantificadores:

William é um lógico. Todos lógicos vestem chapéus engraçados.

$\therefore$ William veste chapéus engraçados.

UD: pessoas

Lx: x é um lógico.

Ex: x veste chapéus engraçados.

w: William.

Traduzindo, temos:

Lw

$\forall x(Lx \rightarrow Ex)$.

$\therefore Fw$

Assim, recuperamos a estrutura que foi perdida pela tradução em LS desse argumento, tornando-o válido em LQ.

Predicados vazios

Um predicado não precisa ser aplicado a coisa alguma no UD. Um predicado que se aplica a nada no UD é chamado de predicado VAZIO.

Suponha que queremos simbolizar as seguintes:

18. Todo macaco sabe linguagem de sinais.
19. Algum macaco sabe linguagem de sinais.

É possível escrever a chave de simbolização para essas sentenças desta maneira:

UD : animais

Mx : x é macaco

Sx : x sabe linguagem de sinais.

A sentença 18 pode ser traduzida por $\forall x(Mx \rightarrow Sx)$.

A sentença 19 pode ser traduzida por $\exists x(Mx \& Sx)$.

É tentador dizer que a sentença 18 implica a sentença 19: se todo macaco sabe linguagem de sinais, então deve existir um macaco que saiba linguagem de sinais. Essa é uma inferência lógica válida na lógica aristotélica: Todos M são S, logo, algum M é S. Todavia, essa implicação não é válida na LQ. É possível para a sentença $\forall x(Mx \rightarrow Sx)$ ser verdadeira apesar da sentença $\exists x(Mx \& Sx)$ ser falsa.

Como isso pode se dar? A resposta vem da consideração se essas sentenças seriam verdadeiras ou falsas se não existisse nenhum macaco.

Definimos os quantificadores $\forall$ e $\exists$ de tal modo que $\forall \mathcal{A}$ é equivalente a $\neg\exists\neg\mathcal{A}$. Assim, o quantificador universal não envolve a existência de coisa alguma — apenas não existência. Se a sentença 18 é verdadeira, então não existem macacos que não sabem linguagem de sinais. Se não existissem macacos, então $\forall x(Mx \to Sx)$ seria verdadeira e $\exists x(Mx \& Sx)$ seria falsa.

Nós admitimos predicados vazios, porque queremos poder dizer coisas como 'Eu não sei se há qualquer macaco, mas se qualquer macaco existir, então ele saberá linguagem de sinais.' I.e., queremos poder ter predicados que ou não se refiram ou possam não se referir a coisa alguma.

O que acontece se adicionarmos um predicado vazio G à interpretação acima? Por exemplo, podemos definir Gx para significar 'x é uma geladeira.' Assim, a sentença $\forall x(Gx \to Mx)$ será verdadeira. Isso é contra intuitivo, já que não queremos dizer que há macacos geladeiras. Todavia, é importante lembrar-se que $\forall x(Gx \to Mx)$ significa que qualquer membro do UD que seja uma geladeira será um macaco. Já que o UD é animais, não há geladeiras no UD e a sentença é trivialmente verdadeira.

Se você estivesse realmente traduzindo a sentença 'Todas geladeiras são macacos', então você desejaria adicionar eletrodomésticos no UD. Então o predicado G não seria vazio e a sentença $\forall x(Gx \to Mx)$ seria falsa.

- ▷ Um UD deve ter *no mínimo* um membro.
- ▷ Um predicado pode ser aplicado a todos, algum ou nenhum membro do UD.
- ▷ Uma constante deve selecionar *exatamente* um membro do UD. Um membro do UD pode ser selecionado por uma constante, muitas constantes ou nenhuma.

Escolhendo um Universo de Discurso

A simbolização apropriada de uma sentença do português na LQ dependerá da chave de simbolização. De certo modo, isso é óbvio: é relevante se Fx significa 'x é uma fruta' ou 'x é uma flor.' O significado das sentenças na LQ dependerá do UD.

Deixemos Rx significar 'x é uma rosa' e Ex significar 'x tem espinhos' e consideremos a sentença:

20. Toda rosa tem espinhos.

É tentador dizer que a sentença 20 deve ser traduzida por $\forall x(Rx \to Ex)$. Se o UD contém todas as rosas, isso seria correto. Todavia, se o UD consistir meramente de coisas na mesa da minha cozinha, então $\forall x(Rx \to Ex)$ significaria apenas que toda rosa na mesa da minha cozinha tem espinhos. Se não houver rosas na mesa da minha cozinha, a sentença seria trivialmente verdadeira.

O quantificador universal se aplica apenas sobre os membros do UD, então precisamos incluir todas as rosas no UD para poder traduzir a sentença 20. Temos duas opções. Em primeiro lugar, podemos restringir o UD para incluir todas e somente todas as rosas. Então a sentença 20 se torna $\forall xEx$. Isso significa que tudo no UD tem espinhos. Essa opção pode nos evitar problemas se toda sentença que quisermos traduzir utilizando a chave de simbolização for sobre rosas.

Em segundo lugar, podemos deixar o UD conter coisas além de rosas: tulipas, ratos, moscas e qualquer outra coisa. Assim, a sentença 20 deve ser $\forall x(Rx \to Ex)$.

Se quiséssemos o quantificador universal para significar tudo, sem restrição, então podemos especificar um UD que contenha tudo. Isso criaria certos problemas. 'Tudo' inclui coisas que foram somente imaginadas, como personagens ficcionais? Se por um lado, desejamos poder simbolizar argumentos sobre Hamlet ou Sherlock Holmes, então precisaremos da opção de incluirmos personagens ficcionais no UD. Se por outro lado, nunca precisarmos falar sobre coisas que não existem, essa pergunta nem fará sentido. Há problemas filosóficos aqui que não desejamos adentrar. Podemos evitar essas dificuldades ao especificarmos sempre o UD. Por exemplo, se desejarmos falar sobre plantas, pessoas e cidades, então o UD pode ser 'coisas vivas e lugares.'

Suponha que desejamos traduzir a sentença 20 e, com a mesma chave de simbolização, traduzir estas sentenças:

21. Esmeralda tem uma rosa em seu cabelo.
22. Todos estão com raiva de Esmeralda.

Precisamos de um UD que inclua rosas (para podermos simbolizar a sentença 20) e um UD que inclua pessoas (para podermos traduzir as sentenças 21-22.) Aqui está uma chave adequada:

UD: pessoas e plantas

Px: x é uma pessoa.

Rx: x é uma rosa.

Ex: x tem espinhos.

Cxy: x está com raiva de y.

Hxy: x tem y em seu cabelo.

e: Esmeralda

Já que não temos um predicado que signifique '... tem uma rosa em seu cabelo', a tradução da sentença 21 precisará de uma paráfrase. A sentença diz que há uma rosa no cabelo de Esmeralda; i.e., há algo que é rosa e está no cabelo de Esmeralda. Assim, temos $\exists x(Rx \& Hex)$.

É tentador traduzir a sentença 22 por $\forall xCxe$. Infelizmente, isso significaria que todo membro do UD está com raiva de Esmeralda — tanto as pessoas quanto as plantas. Claramente a sentença 22 não significa isso.

'Todos' significa todas as pessoas, não todos os membros do UD. Assim, podemos parafrasear a sentença por 'Toda pessoa está com raiva de Esmeralda.' Sabemos como traduzir sentenças como essas: $\forall x(Px \rightarrow Cxe)$.

Em geral, o quantificador universal pode ser usado para significar 'todas as pessoas' se o UD conter apenas pessoas. Se houver pessoas e outras coisas no UD, então 'todos' deve ser entendido como 'todas pessoas.'

Traduzindo pronomes pessoais

Ao se traduzir para a LQ, é importante compreender a estrutura das sentenças que você deseja traduzir. Na LQ, o essencial é a tradução final e, algumas vezes, você poderá passar diretamente de uma sentença em português para uma sentença de LQ. Outras vezes, parafrasear a sentença uma ou mais vezes facilitará o trabalho de tradução. Cada paráfrase deve partir da sentença original para algo mais próximo àquilo que você pode traduzir diretamente para LQ.

Para os próximos exemplos, utilizaremos a seguinte chave de simbolização:

UD: pessoas

Gx: x pode tocar guitarra.

Rx: x é uma estrela do rock.

l: Lemmy

Agora considere as sentenças:

23. Se Lemmy pode tocar guitarra, então ele é uma estrela do rock.
24. Se uma pessoa pode tocar guitarra, então ela é uma estrela do rock.

As sentenças 23 e 24 tem a mesma consequente ('... ele é uma estrela do rock), mas não podem ser traduzidas da mesma maneira. Parafrasear as sentenças, substituindo os pronomes com referências explícitas ajudará nossa tradução.

A sentença 23 pode ser parafraseada por 'Se Lemmy pode tocar guitarra, então *Lemmy* é uma estrela do rock.' Isso pode ser obviamente traduzido por $Gl \to Rl$.

A sentença 24 deve ser parafraseada diferentemente: 'Se uma pessoa pode tocar guitarra, então *essa pessoa* é uma estrela rock.' Essa sentença não é sobre um indivíduo em particular, então precisaremos de uma variável. Podemos parafrasear essa sentença em uma tradução próxima da qual desejamos: 'Para qualquer pessoa x, se x pode tocar guitarra, então x é uma estrela do rock.' Assim, essa sentença pode ser traduzida por $\forall x(Gx \to Rx)$. Isso é o mesmo que 'Todos que podem tocar guitarra são estrelas do rock.'

Considere mais essas sentenças:

25. Se qualquer um pode tocar guitarra, então Lemmy pode.
26. Se qualquer um pode tocar guitarra, então ele ou ela é uma estrela do rock.

Essas duas sentenças sentenças têm a mesma antecedente ('qualquer um pode tocar guitarra...'), mas elas têm estruturas lógicas diferentes.

A sentença 25 pode ser parafraseada por 'Se alguém pode tocar guitarra, então Lemmy pode tocar guitarra.' A antecedente e a consequente são sentenças separadas, então essa sentença pode ser simbolizada com um condicional como o operador lógico principal: $\exists xGx \rightarrow Gl$.

A sentença 26 pode ser parafraseada por 'Para qualquer pessoa, se essa pessoa puder tocar guitarra, então essa pessoa é uma estrela do rock.' Seria um erro simbolizá-la com um quantificador existencial, porque ela fala de todas as pessoas. Essa sentença é equivalente a 'Todos guitarristas são estrelas do rock.' Ela é melhor traduzida por $\forall x(Gx \rightarrow Rx)$.

As palavras portuguesas 'qualquer' e 'qualquer um' normalmente devem ser traduzidas por quantificadores. Como esses dois exemplos mostram, algumas vezes elas exigirão um quantificador existencial (como na sentença 25) e alguma vezes, um quantificador universal (como na sentença 26). Se você estiver tendo dificuldades para determinar qual desses é necessário para a tradução, parafraseie a sentença para outra sentença do português que utilizes outras palavras além de 'qualquer' e 'qualquer um.'

Quantificadores e escopo

Na sentença $\exists xGx \rightarrow Gl$, o escopo do quantificador existencial é a expressão Gx. Faria diferença se o escopo do quantificador fosse a sentença inteira? Ou seja, a sentença $\exists x(Gx \rightarrow Gl)$ significa algo diferente?

Com a chave dada acima , $\exists xGx \rightarrow Gl$ significa que se há algum guitarrista, então Lemmy é um guitarrista. $\exists x(Gx \rightarrow Gl)$ significaria que há alguma pessoa tal que se essa pessoa fosse um guitarrista, então Lemmy seria um guitarrista. Lembre-se que o condicional aqui é material; o condicional será verdadeiro se o antecedente for falso. Façamos a constante p denotar o autor deste livro, alguém que certamente não é um guitarrista. A sentença $Gp \rightarrow Gl$ é verdadeira porque Gp é falsa. Já que alguém (isto é, p) satisfaz a sentença, então $\exists x(Gx \rightarrow Gl)$ é verdadeira. A sentença é verdadeira porque há um não-guitarrista, independentemente das habilidades de Lemmy com a guitarra.

Algo estranho ocorreu quando mudamos o escopo do quantificador, porque o condicional na LQ é um condicional material. Para mantermos o mesmo significado, seria necessário mudar o quantificador: $\exists xGx \rightarrow Gl$ significa o mesmo que $\forall x(Gx \rightarrow Gl)$ e $\exists x(Gx \rightarrow Gl)$ significa o mesmo que $\forall xGx \rightarrow Gl$.

Essa estranheza não surge com outros conectivos ou se a variável estiver na consequente do condicional. Por exemplo, $\exists xGx \& Gl$ significa a mesma coisa que $\exists x(Gx \& Gl)$ e $Gl \rightarrow \exists xGx$ significa o mesmo que $\exists x(Gx \rightarrow Gl)$.

Predicados ambíguos

Suponha que queiramos traduzir a sentença seguinte:

27. Adriano é um cirurgião habilidoso.

Deixemos o UD ser pessoas, Kx significar 'x é um cirurgião habilidoso' e a significar Adriano. A sentença 27 é simplesmente Ka.

Suponha agora que queiramos traduzir este argumento:

> O hospital contratará somente um cirurgião habilidoso. Todos cirurgiões são gananciosos. Billy é um cirurgião, mas não é habilidoso. Portanto, Billy é ganancioso, mas o hospital não irá contratá-lo.

Precisamos distinguir ser um *cirurgião habilidoso* de meramente ser um *cirurgião*. Definiremos a chave de simbolização como do seguinte modo:

UD: pessoas

Gx: x é ganancioso.

Hx: O hospital irá contratar x.

Rx: x é um cirurgião.

Kx: x é habilidoso.

b: Billy

O argumento pode ser traduzido desse modo:

$\forall x[\neg(Rx \& Kx) \to \neg Hx]$

$\forall x(Rx \to Gx)$

$Rb \& \neg Kb$

$\therefore\ Gb \& \neg Hb$

A seguir, suponha que queiramos traduzir este argumento:

> Carol é uma cirurgiã habilidosa e uma jogadora de tênis. Logo,
> Carol é uma jogadora de tênis habilidosa.

Se começarmos com a chave de simbolização utilizada no argumento anterior, poderíamos adicionar um predicado (tal que Tx signifique 'x é uma jogadora de tênis) e uma constante (tal que c signifique 'Carol'). Então, o argumento se torna:

$$(Rc\&Kc)\&Tc$$
$$\therefore\ Tc\&Kc$$

Essa tradução é um desastre! Ela parte de um argumento horrível em português e o traduz como um argumento válido na LQ. O problema é que há uma diferença entre ser *habilidoso enquanto cirurgião* e ser *habilidoso enquanto jogador de tênis*. Para traduzirmos esse argumento corretamente, é necessário dois predicados separados, um para cada tipo de habilidade. Se K_1x significar 'x é habilidoso enquanto cirurgião' e K_2x significar 'x é habilidoso enquanto jogador de tênis', então podemos simbolizar o argumento deste modo:

$$(Rc\&K_1c)\&Tc$$
$$\therefore\ Tc\&K_2c$$

Assim como o argumento do português que essa simbolização traduz, ele é inválido.

A moral desses exemplos é evidenciar que você precisa ser cuidadoso ao simbolizar predicados que são ambíguos. Problemas similares surgem com predicados como *bom*, *mal*, *grande* e *pequeno*. Assim como cirurgiões e jogadores de tênis habilidosos tem habilidades diferentes, cachorros grandes, ratos grandes e problemas grandes são grandes de formas diferentes.

É suficiente ter um predicado que signifique 'x é um cirurgião habilidoso', em vez de dois predicados, 'x é habilidoso' e 'x é um cirurgião'? Algumas vezes sim. Como a sentença 27 evidencia, em alguns casos não precisamos distinguir entre cirurgiões habilidosos e outros cirurgiões.

Devemos sempre distinguir entre as formas diferentes de ser habilidoso, bom, ruim ou grande? Não. Como o argumento sobre Billy evidencia, algumas vezes precisamos falar apenas sobre um tipo de habilidade. Se você estiver traduzindo um argumento que é apenas sobre cachorros, está correto definir um predicado que signifique 'x é grande.' Todavia, se o UD incluir cachorros e ratos, provavelmente é melhor fazer o predicado significar 'x é um cachorro grande.'

Quantificadores múltiplos

Considere esta chave de simbolização e as sentenças que a seguem:

UD: pessoas e cachorros

Cx: x é um cachorro.

Axy: x é amigo de y.

Dxy: x é dono de y.

f: Fifi

g: Geraldo

28. Fifi é um cachorro.
29. Geraldo é dono de um cachorro.
30. Alguém é dono de um cachorro.
31. Todos os amigos de Geraldo são donos de cachorros.
32. Todo dono de cachorro é amigo de um dono de cachorro.

A sentença 28 é fácil: Cf.

A sentença 29 pode ser parafraseada por 'Há um cachorro cujo dono é Geraldo.' Isso pode ser traduzido por $\exists x(Cx \& Dgx)$.

A sentença 30 pode ser parafraseada por 'Há algum y tal que y é um dono de cachorro.' A subsentença 'y é um dono de cachorro' é semelhante à sentença 28, exceto que ela trata da variável y ao invés de Geraldo. Assim, podemos traduzir a sentença 30 por $\exists x \exists y(Cx \& Dyx)$.

A sentença 31 pode ser parafraseada por 'Todo amigo de Geraldo é um dono de cachorro.' Traduzindo parte dessa sentença, temos $\forall x(Axg \rightarrow x$ é um dono de cachorro). Mais uma vez, é importante reconhecer que 'x é um dono de cachorro' é estruturalmente igual à sentença 29. Uma vez que já temos um quantificador-x, é necessário utilizar outra variável para o quantificador existencial. Qualquer outra variável será suficiente. Utilizando z, a sentença 31 pode ser traduzida por $\forall x[Axg \rightarrow \exists z(Cz \& Dxz)]$.

A sentença 32 pode ser parafraseada por 'Para qualquer x que seja um dono de cachorro, há um dono de cachorro que é amigo de x.' Parcialmente traduzida, ela se torna:

$$\forall x[x \text{ é um dono de cachorro} \rightarrow \exists y(y \text{ é um dono de cachorro} \& Axy)].$$

Na tradução completa, a sentença 32 é:

$$\forall x[\exists z(Cz \& Dxz) \rightarrow \exists y(\exists z(Cz \& Dyz) \& Axy)].$$

Considere esta chave de simbolização e estas sentenças:

UD: pessoas

Gxy: x gosta de y.

j: João

c: Carlos

33. João gosta de todo mundo que Carlos gosta.
34. Existe alguém que gosta de todos que gostam de todos aqueles que ele gosta.

A sentença 33 pode ser parcialmente traduzida por $\forall x$(Carlos gosta de $x \rightarrow$João gosta de x). Isso se torna $\forall x(Gcx \rightarrow Gjx)$.

A sentença 34 é quase um trava-línguas. Não há muita esperança em traduzirmos a sentença por inteira, mas podemos fazê-lo passo a passo. A princípio, uma tradução poderia ser assim:

$$\exists x \text{ tal que todos que gostam de todos que } x \text{ goste, } x \text{ gosta deles}$$

A parte que permaneceu em português é uma sentença universal, então podemos traduzi-la:

$$\exists x \forall y(y \text{ gosta de todos que } x \text{ gosta} \rightarrow x \text{ gosta de } y).$$

A antecedente da condicional é estruturalmente igual à da sentença 33, y e x em vez de Carlos e João. Assim, a sentença 34 pode ser completamente traduzida por:

$$\exists x \forall y[\forall z(Gxz \rightarrow Gyz) \rightarrow Gxy].$$

Ao simbolizar sentenças com quantificadores múltiplos, é melhor proceder através de passos pequenos. Parafraseie a sentença em português de modo que a estrutura lógica esteja simbolizada na LQ, então traduza o restante pouco a pouco. Assim, ao invés da tarefa aterrorizante de se traduzir uma sentença longa, você terá a tarefa mais simples de traduzir fórmulas mais curtas.

4.5 Sentenças da LQ

Nesta seção, daremos uma definição formal para *fórmula bem formada* (fbf) e *sentença* da LQ.

Expressões

Há seis tipos de símbolos na LQ:

Predicados com subscritos, quando for necessário	$A, B, C, \ldots, Z, A_1, B_1, Z_1, A_2, A_{25}, J_{375}, \ldots$
Constantes com subscritos, quando for necessário	$a, b, c, \ldots, w, a_1, w_4, h_7, m_{32}, \ldots$
Variáveis com subscritos, quando for necessário	$x, y, z, x_1, y_1, z_1, x_2, \ldots$
Conectivos	$\neg, \&, \vee, \rightarrow, \leftrightarrow$
Parênteses	(,)
Quantificadores	$\forall, \exists$

Nós definimos uma EXPRESSÃO DE LQ como qualquer sequência de símbolos de LQ. Escolha quaisquer símbolos de LQ e escreva-os em qualquer ordem e você terá uma expressão.

Fórmulas bem formadas

Por definição, um TERMO DE LQ é ou uma constante ou uma variável.

Uma FÓRMULA ATÔMICA DE LQ é um predicado de lugar-n seguido por termos n.

Assim como fizemos para a LS, daremos uma definição *recursiva* para uma fbf de LQ. De fato, grande parte da definição se assemelhará à definição de fbf para LS: toda fórmula atômica é uma fbf e você pode formar novas fbfs aplicando conectivos sentenciais.

Poderíamos apenas adicionar uma regra para os quantificadores e satisfazer-nos com isso. Por exemplo: Se $\mathcal{A}$ é uma fbf, então $\forall x\mathcal{A}$ e $\exists x\mathcal{A}$ são fbfs. Entretanto, isso permitira que sentenças bizarras como $\forall x\exists xDx$ e $\forall xDw$ fossem fbfs. O que elas poderiam significar? Poderíamos adotar uma interpretação de tais sentenças, mas ao contrário disso daremos uma definição de uma fbf de tal modo que tais abominações não possam ser fbfs.

Para que $\forall x\mathcal{A}$ seja uma fbf, é necessário que $\mathcal{A}$ contenha a variável x e que já $\mathcal{A}$ não contenha um quantificador-x. Assim, $\forall xDw$ não será fbf porque 'x' não ocorre em Dw e $\forall x\exists xDx$ não será uma fbf porque $\exists xDx$ já contém um quantificador-x.

1. Toda fórmula atômica é uma fbf.
2. Se $\mathcal{A}$ é uma fbf, então $\neg\mathcal{A}$ é uma fbf.
3. Se $\mathcal{A}$ e $\mathcal{B}$ são fbfs, então $(\mathcal{A}\&\mathcal{B})$ é uma fbf.
4. Se $\mathcal{A}$ e $\mathcal{B}$ são fbfs, então $(\mathcal{A} \vee \mathcal{B})$ é uma fbf.
5. Se $\mathcal{A}$ e $\mathcal{B}$ são fbfs, então $(\mathcal{A} \rightarrow \mathcal{B})$ é uma fbf.
6. Se $\mathcal{A}$ e $\mathcal{B}$ são fbfs, então $(\mathcal{A} \leftrightarrow \mathcal{B})$ é uma fbf.
7. Se $\mathcal{A}$ é uma fbf, χ é uma variável, $\mathcal{A}$ contém pelo menos uma ocorrência de χ e não contém quantificador-χ, então $\forall\chi A$ é uma fbf.

8. Se $\mathcal{A}$ é uma fbf, χ é uma variável, $\mathcal{A}$ contém pelo menos uma ocorrência de χ e não contém quantificador-χ, então $\exists\chi A$ é uma fbf.

9. Todas e apenas as fbfs de LQ podem ser geradas através da aplicação dessas regras.

Note que a variável 'χ' que aparece na definição acima não é a variável x. Ela é uma metavariável que pode indicar qualquer variável da LQ. Assim como $\forall xAx$ é uma fbf, $\forall yAy$, $\forall zAz$, $\forall x_4 Ax_4$ e $\forall z_{29} Az_{29}$ também o são.

Agora podemos dar uma definição formal de escopo: o ESCOPO do quantificador é a subfórmula na qual o quantificador é o operador lógico principal.

Sentenças

Uma sentença é algo que pode verdadeiro ou falso. Na LS, toda fbf era uma sentença. Esse não será o caso na LQ. Considere esta chave de simbolização:

UD: pessoas

Axy: x ama a y.

b: Boris

Considere a expressão Lzz. Ela é uma fórmula atômica: um predicado binário seguido por dois termos. Todas fórmulas atômicas são fbfs, então Lzz é uma fbf, mas isso significa algo? Você poderia pensar que isso significa que z ama si mesmo, do mesmo modo que Lbb significa que Boris ama si mesmo. Todavia, z é uma variável; ela não nomeia uma pessoa do mesmo modo como uma constante o faria. A fbf Lzz não nos diz como interpretar z. Essa variável significa todos? Qualquer um? Alguém? Se tivéssemos um quantificador-z, ele nos diria como interpretar z. Por exemplo, $\exists zLzz$ significaria que alguém ama si mesmo.

Algumas linguagens formais tratam Lzz como possuindo um quantificador universal implícito antecedendo-o. Não adotaremos essa posição. Se você quiser dizer que todos se amam, então você precisará escrever o quantificador: $\forall zLzz$.

Para uma variável fazer sentido, precisamos de um quantificador que nos diga como interpretar essa variável. Por exemplo, o escopo de um quantificador-x é a parte da fórmula na qual o quantificador nos diz como interpretar x.

Para sermos precisos quanto a isso, definiremos uma VARIÁVEL LIGADA como uma ocorrência de uma variável x que está dentro do escopo de um quantificador-x. Uma VARIÁVEL LIVRE é uma ocorrência de uma variável que não está ligada.

Por exemplo, considere a fbf $\forall x(Ex \vee Dy) \rightarrow \exists z(Ex \rightarrow Lzx)$. O escopo do quantificador universal $\forall x$ é $(Ex \vee Dy)$, então o primeiro x é ligado ao quantificador universal, enquanto o segundo e o terceiro x são livres. Já que não há um quantificador-y, y está livre. O escopo do quantificador existencial $\exists z$ é $(Ex \rightarrow Lzx)$, então ambas as ocorrências de z estão ligadas a ele.

Definiremos uma SENTENÇA de LQ como uma fbf de LQ que não contenha variáveis livres.

Convenções notacionais

Adotaremos as mesmas convenções notacionais que fizemos para a LS. Em primeiro lugar, podemos suprimir os parênteses que compreendem sentença inteira. Em segundo lugar, utilizaremos colchetes '[' e ']' em vez de parênteses para facilitar a leitura das fórmulas. Em terceiro lugar, suprimiremos os parênteses entre cada par de conjuntos ao escrevermos uma longa série de conjunções. Por último, suprimiremos os parênteses entre cada par de disjuntos ao escrevermos uma longa série de disjunções.

4.6 Identidade

Considere esta sentença:

35. Pedro deve dinheiro a todas as outras pessoas.

Façamos o UD ser pessoas; isso nos permitirá traduzir 'todas pessoas' como um quantificador universal. Façamos Dxy significar 'x deve dinheiro a y' e p significar Pedro. Assim, podemos simbolizar a sentença 35 como $\forall xDpx$. Infelizmente, essa tradução traz consequências estranhas. Ela diz que Pedro deve dinheiro a todo membro do UD, incluindo Pedro; ela implica que Pedro deve dinheiro a si mesmo. Entretanto, a sentença 35 não diz que Pedro deve dinheiro a si mesmo, ele deve dinheiro a todas as *outras* pessoas. Esse é um problema, pois $\forall x Dpx$ é a melhor tradução que nós podemos dar para essa sentença em LQ.

A solução é adicionar outro símbolo à LQ. O símbolo '$=$' é um predicado binário. Já que ele tem um significado lógico especial, iremos escrevê-lo um pouco diferentemente: para dois termos t_1 e t_2, $t_1 = t_2$ será uma fórmula atômica.

O predicado $x = y$ significa 'x é idêntico a y.' Isso não significa somente que x e y são indistinguíveis ou que todos os predicados sejam verdadeiros para um também o serão para o outro. Mais precisamente, isso significa que x e y são a mesma coisa.

Quando escrevemos $x \neq y$, nós significamos que x e y não são idênticos. Não há razão para introduzir isso como um novo tipo de predicado. Ao invés disso, $x \neq y$ é uma abreviação de $\neg(x = y)$.

Assim, suponha que queiramos simbolizar esta sentença:

36. Pedro é o Senhor Checkov.

Façamos a constante c significar Senhor Checkov. A sentença 36 pode ser simbolizada por $p = c$. Isso significa que as constantes p e c referem-se ao mesmo homem.

Por enquanto, tudo está certo, mas como isso nos ajuda com a sentença 35? Aquela sentença pode ser parafraseada por 'Para todos aqueles que não são Pedro, Pedro é devedor.' Essa é uma sentença que já sabemos como simbolizar: 'Para todo x, se x não é Pedro, então Pedro deve dinheiro a x.' Na LQ com identidade, isso se torna $\forall x(x \neq p \rightarrow Dpx)$.

Em adição às sentenças que utilizam a palavra 'outro', a identidade será útil para simbolizarmos algumas sentenças que contenham as palavras 'além de' e 'apenas.' Considere estes exemplos:

37. Ninguém além de Pedro deve dinheiro a Hiran.
38. Apenas Pedro deve dinheiro a Hiran.

Adicionemos a constante h que significa Hiran.

A sentença 37 pode ser parafraseada por 'Ninguém que não seja Pedro deve dinheiro a Hiran.' Isso pode ser traduzido por $\neg\exists x\,(x \neq p \& Dxh)$.

A sentença 38 pode ser parafraseada por 'Pedro deve dinheiro a Hiran *e* ninguém além de Pedro deve dinheiro a Hiran.' Já traduzimos um dos conjuntos e o outro é bem claro. Essa sentença se torna $Dph \& \neg\exists x\,(x \neq p \& Dxh)$.

Expressões de quantidade

Também podemos utilizar a identidade para expressar quantas coisas existem de um tipo particular. Por exemplo, considere estas sentenças:

39. Há no mínimo uma maçã sobre a mesa.
40. Há no mínimo duas maçãs sobre a mesa.
41. Há no mínimo três maçãs sobre a mesa.

Façamos o UD ser coisas sobre a mesa e Mx significar 'x é uma maçã.'

A sentença 39 não requer identidade. Ela pode ser adequadamente traduzida por $\exists x Mx$: há alguma maçã na mesa — talvez muitas, mas no mínimo uma.

É tentador traduzir a sentença 40 também sem identidade. Assim, consideremos a sentença $\exists x\exists y(Mx \& My)$. Isso significa que há alguma maçã x e alguma maçã y no UD. Já que nada impede que x e y selecionem o mesmo membro do UD, isso seria verdadeiro mesmo se houvesse uma única maçã sobre a mesa. Para deixarmos claro que há duas maçãs *diferentes*, precisamos de um predicado de identidade. Essa sentença precisa dizer que há duas maçãs e que elas não são idênticas, então ela pode ser traduzida por $\exists x\exists y(Mx \& My \& x \neq y)$.

A sentença 41 requer que falemos sobre três maçãs diferentes. Ela pode ser traduzida por

$$\exists x\exists y\exists z(Mx \& My \& Mz \& x \neq y \& y \neq z \& x \neq z).$$

Continuando dessa forma, poderíamos traduzir a sentença 'Há no mínimo n maçãs sobre a mesa.' Existe um resumo de como simbolizar sentenças como essas na p. 187.

Agora considere as seguintes sentenças:

42. Há no máximo uma maçã sobre a mesa.

43. Há no máximo duas maçãs sobre a mesa.

A sentença 42 pode ser parafraseada por 'Não é o caso que haja no mínimo duas maçãs sobre a mesa.' Isso é apenas a negação da sentença 40:

$$\neg\exists x\exists y\,(Mx\&My\&x \neq y).$$

A sentença 42 pode também ser abordada de outro modo. Ela significa que quaisquer maçãs que houver sobre a mesa devem ser idênticas a si mesmas, então ela pode ser traduzida por $\forall x\forall y[(Mx\&My) \rightarrow x = y]$. Essas sentenças são logicamente equivalentes, logo, as duas estão corretas.

Semelhantemente, a sentença 43 pode ser traduzida de duas formas equivalentes. Ela pode ser parafraseada por 'Não é o caso que haja três ou mais maçãs distintas sobre a mesa' e assim ser traduzida como a negação da sentença 41. Utilizando quantificadores universais, ela também pode ser traduzida por:

$$\forall x\forall y\forall z[(Mx\&My\&Mz) \rightarrow (x = y \vee y = z \vee x = z)]$$

Veja a página xxx para o caso geral.

Os exemplos acima são sobre maçãs, mas a estrutura lógica dessas sentenças traduz desigualdades matemáticas como $a \geq 3$ e $a \leq 2$ e assim por diante. Também queremos poder traduzir sentenças de igualdade que dizem exatamente quantas coisas existem. Por exemplo:

44. Há exatamente uma maçã sobre a mesa.

45. Há exatamente duas maçãs sobre a mesa.

A sentença 44 pode ser parafraseada por 'Há *no mínimo* uma maçã sobre a mesa e há *no máximo* uma maçã sobre a mesa.' Isso é apenas a conjunção das sentenças 39 e 42: $\exists xMx\&\forall x\forall y[(Mx\&My) \rightarrow x = y]$. Todavia, essa é uma maneira complicada de traduzi-la. Talvez seja mais claro traduzi-la parafraseando a sentença 44 por 'Há uma coisa e tal coisa é a única maçã sobre a mesa.' Assim, a sentença pode ser traduzida por: $\exists x[Mx\&\neg\exists y\,(My\&x \neq y)]$.

Semelhantemente, a sentença 45 pode ser parafraseada por 'Há duas maçãs diferentes sobre a mesa, e essas são as únicas maçãs sobre a mesa.' Isso pode ser traduzido por

$$\exists x \exists y[Mx \& My \& x \neq y \& \neg \exists z\,(Mz \& x \neq z \& y \neq z)].$$

Finalmente, considere a seguinte sentença:

46. Há no máximo duas coisas sobre a mesa.

É tentador adicionar um predicado de forma que Cx signifique 'x é uma coisa sobre a mesa.' Entretanto, isso é desnecessário. Já que o UD é o conjunto de coisas sobre a mesa, todos membros do UD estão sobre a mesa. Se quisermos falar sobre *uma coisa sobre a mesa*, precisamos apenas de utilizar um quantificador. A sentença 46 pode ser simbolizada como a sentença 43 (que disse que havia no máximo duas maçãs sobre a mesa), mas omitiremos o predicado. Assim, essa sentença pode ser traduzida por $\forall x \forall y \forall z(x = y \vee x = z \vee y = z)$.

Técnicas para simbolizar expressões de quantidade ('no máximo', 'no mínimo' e 'exatamente') estão resumidas na página xxx.

Descrições definidas

Recorde que uma constante de LQ deve se referir a um membro do UD. Essa restrição nos permite evitar o problema dos termos não-referenciais. Dado um UD que inclua apenas criaturas que realmente existam, mas incluindo em nossa chave de simbolização uma constante c que signifique 'quimera' (uma criatura mítica), sentenças que contivessem c se tornariam impossíveis de ser avaliadas.

A solução mais influente para esse problema foi introduzida por Bertrand Russell em 1905. Russell perguntou como deveríamos entender esta sentença:

47. O presente rei da França é careca.

A frase 'o presente rei da França' supõe selecionar um indivíduo do universo de discurso através de uma descrição definida. Entretanto, não havia rei da França em 1905 e nem agora há. Já que a descrição é um termo não-referenciais, não podemos simplesmente definir uma constante para significar 'o presente rei da França' e traduzi-la simplesmente por Rf

A ideia de Russell era que sentenças que contenham descrições definidas têm uma estrutura lógica diferente daquelas que contém nomes próprios, apesar de compartilharem a mesma forma gramatical. Qual o significado de quando utilizamos uma descrição referencial não problemática como 'o pico mais alto no estado de Washington'? Pressupomos a existência de tal pico, porque não poderíamos falar sobre ele se este não fosse o caso e conferimos, assim, significado à expressão. Ademais, também pressupomos que ele seja o único do seu tipo. Se houvesse outro pico no estado de Washington com exatamente a mesma altura que o Monte Rainier, então o Monte Rainier não seria *o* pico mais alto.

De acordo com essa análise, a sentença 47 está dizendo três coisas. Em primeiro lugar, ela faz um juízo existencial: atualmente há algum rei da França. Em segundo lugar, ela faz um juízo de singularidade: atualmente esse homem é o único homem rei da França. Em terceiro lugar, ela faz um juízo de predicação: esse homem é careca.

Para podermos simbolizar descrições definidas desse modo, precisamos identificar o predicado. Sem isso, não poderíamos traduzir o juízo de singularidade o qual (de acordo Russell) está implícito na descrição definida.

Façamos que UD seja *pessoas que vivem atualmente* e que Fx e Cx signifiquem, respectivamente, 'x é o atual rei da França' e 'x é careca.' Assim, a sentença 47 pode ser traduzida por

$$\exists x[Fx \& \neg\exists y\,(Fy \& x \neq y)\,\& Cx].$$

Isso diz que há algum homem que é o atual rei da França, que ele é atualmente o único rei da França e que ele é careca.

Entendida dessa forma, a sentença 47 tem sentido, mas é falsa. Ela diz que esse homem existe, mas esse não é o caso.

O problema dos termos não-referenciais é ainda mais estranho quando tentamos traduzir negações. Por exemplo, considere esta sentença:

48. O presente rei da França não é careca.

De acordo com Russell, essa sentença é ambígua em português. Ela poderia significar uma dessas duas coisas:

48a. Não é o caso que o presente rei da França seja careca.

48b. O presente rei da França é não-careca.

Ambos os significados possíveis negam a sentença 47, mas elas colocam a negação em lugares diferentes.

A sentença 48a é chamada de NEGAÇÃO EXTERNA, porque ela nega toda a sentença. Ela pode ser traduzida por:

$$\neg\exists x[Fx \& \neg\exists y\,(Fy \& x \neq y)\,\& Cx]$$

Isso nada diz sobre o atual rei da França, ao invés disso, ela diz que alguma sentença sobre o atual rei da França é falsa. Já que a sentença 47 é falsa, a sentença 48a é verdadeira.

A sentença 48b diz algo sobre o atual rei da França. Ela diz que ele está privado da propriedade da calvície. Como a sentença 47, ela faz um juízo existencial e de singularidade; ela apenas nega a predicação. Isso é chamado de NEGAÇÃO INTERNA. Ela pode ser traduzida por $\exists x[Fx \& \neg\exists y\,(Fy \&\ x \neq y)\,\& \neg Cx]$. Já que no presente não há rei da França, a sentença é falsa.

A teoria de Russell das descrições definidas resolve o problema dos termos não-referenciais e também explica o porquê isso parecia tão paradoxal. Antes de distinguirmos as negações externa e interna, parecia que sentenças como a 48 deveriam ser ambas verdadeiras e falsas. Mostrando que sentenças desse tipo são ambíguas, Russel demonstrou que elas são entendidas verdadeiras de uma maneira e falsas de outra.

Para uma discussão detalhada da teoria de Russell das descrições definidas, incluindo as objeções a ela, ver o verbete 'descrições' de Peter Ludlow em *The Stanford Encyclopedia of Philosophy*: edição do verão de 2005, editado por Edward N. Zalta, http://plato.stanford.edu/archives/sum2005/entries/descriptions/.

4.7 Exercícios práticos

⋆ **Parte A** Utilizando a chave de simbolização dada, traduza cada sentença do português para a LQ.

UD: todos animais

Jx: x é um jacaré.

Mx: x é um macaco.

Rx: x é um réptil.

Zx: x vive no zoológico.

Lxy: x ama y.

a: Amos

b: Bouncer

c: Cleo

1. Amos, Bouncer e Cleo vivem no zoológico.
2. Bouncer é réptil, mas não é um jacaré.
3. Se Cleo ama Bouncer, então Bouncer é um macaco.
4. Se ambos Bouncer e Cleo são jacarés, então Amos os ama.
5. Algum réptil vive no zoológico.
6. Todo jacaré é réptil.
7. Qualquer animal que vive no zoológico é um macaco ou um jacaré.
8. Há répteis que não são jacarés.
9. Cleo ama répteis.
10. Bouncer ama todos os macacos que vivem no zoológico.
11. Todos os macacos que Amos ama também o amam.
12. Se qualquer animal é um réptil, então Amos também é.

13. Se qualquer animal é um jacaré, então ele também é um réptil.

14. Todo macaco que Cleo ama é também amado por Amos.

15. Há um macaco que ama Bouncer, mas infelizmente Bouncer não o ama reciprocamente.

Parte B Esses são os modos silogísticos identificados por Aristóteles e seus sucessores, apresentados com seus nomes medievais. Traduza cada argumento para LQ.

Barbara: Todo B é C. Todo A é B. $\therefore$ Todo A é C.

Baroco: Todo C é B. Algum A não é B. $\therefore$ Algum A não é C.

Bocardo: Algum B não é C. Todo A é B. $\therefore$ Algum A não é C.

Celantes: Nenhum B é C. Todo A é B. $\therefore$ Nenhum C é A.

Celarent: Nenhum B é C. Todo A é B. $\therefore$ Nenhum A é C.

Camestres: Todo C é B. Nenhum A é B. $\therefore$ Nenhum A é C.

Cesare: Nenhum C é B. Todo A é B. $\therefore$ Nenhum A é C.

Dabitis: Todo B é C. Algum A é B. $\therefore$ Algum C é A.

Darii: Todo B é C. Algum A é B. $\therefore$ Algum C é A.

Datisi: Todo B é C. Algum A é B. $\therefore$ Algum A é C.

Disamis: Algum B é C. Todo A é B. $\therefore$ Algum A é C.

Ferison: Nenhum B é C. Algum A é B. $\therefore$ Algum A não é C.

Ferio: Nenhum B é C. Algum A é B. $\therefore$ Algum A não é C.

Festino: Nenhum C é B. Algum A é B. $\therefore$ Algum A não é C.

Baralipton: Todo B é C. Todo A é B. $\therefore$ Algum C é A.

Frisesomorum: Algum B é C. Nenhum A é B. $\therefore$ Algum C não é A.

Parte C Utilizando a chave de simbolização dada, traduza cada sentença do português para a LS.

UD: todos animais

Cx: x é um cachorro.

Sx: x gosta de filmes de samurai.

Mxy: x é maior que y.

b: Bertie

e: Emerson

f: Fergis

1. Bertie é um cachorro que gosta de filmes de samurai.
2. Bertie, Emerson e Fergis são cachorros.
3. Emerson é maior que Bertie e Fergis é maior que Emerson.
4. Todos cachorros gostam de filmes de samurai.
5. Apenas cachorros gostam de filmes de samurai.
6. Há um cachorro que é maior que Emerson.
7. Se há um cachorro que é maior que Fergis, então há um cachorro que é maior que Emerson.
8. Nenhum animal que gosta de fiimes de samurai é maior que Emerson.
9. Nenhum cachorro é maior que Fergis.
10. Qualquer animal que não goste de filmes de samurai é maior que Bertie.
11. Há um animal que tem um tamanho entre Bertie e Emerson.
12. Não há cachorro cujo tamanho esteja entre os de Bertie e Emerson.
13. Nenhum cachorro é maior que si mesmo.
14. Para todo cachorro, há algum outro cachorro maior que ele.
15. Há um animal que é menor que todo cachorro.
16. Se há um animal que é maior que qualquer cachorro, então esse animal não gosta de filmes de samurai.

Parte D Para cada argumento, escreva uma chave de simbolização e traduza o argumento para a LQ.

1. Nada na minha mesa escapa à minha atenção. Há um computador na minha mesa. Assim, há um computador que não escapa à minha atenção.
2. Todos os meus sonhos são em preto e branco. Programas de TV antigos são preto e branco. Portanto, alguns de meus sonhos são programas de TV antigos.
3. Nem Holmes nem Watson estiveram na Austrália. Uma pessoa pode ver um cangaru apenas se ela foi à Austrália ou ao zoológico. Apesar de Watson não ter visto um canguru, Holmes o viu. Portanto, Holmes foi ao zoológico.
4. Ninguém espera a Inquisição Espanhola. Ninguém conhece os problemas que eu vi. Portanto, qualquer um que espera a Inquisição Espanhola conhece os problemas que eu vi.
5. Um antílope é maior que um cesto de pão. Estou pensando em algo que seja maior que um cesto de pão e que seja um antílope ou um melão. Desse modo, estou pensando em um melão.
6. Todos bebês são ilógicos. Ninguém que seja ilógico pode lidar com um crocodilo. Berthold é um bebê. Por conseguinte, Berthold não pode lidar com um crocodilo.

⋆ **Parte E** Utilizando a chave de simbolização dada, traduza cada sentença do português para a LQ.

UD: doces

Cx: x tem chocolate.

Mx: x tem marzipã.

Ax: x tem açúcar.

Bx: Boris comeu x.

Lxy: x é melhor que y.

1. Boris nunca comeu um doce.
2. Marzipã é sempre feito com açúcar.
3. Algum doce não tem açúcar.

4. O melhor doce de todos é chocolate.
5. Nenhum doce é melhor que si mesmo.
6. Boris nunca comeu chocolate sem açúcar.
7. Boris comeu marzipã e chocolate, mas nunca juntos.
8. Qualquer doce com chocolate é melhor que qualquer doce sem ele.
9. Qualquer doce com chocolate e marzipã é melhor que qualquer doce sem os dois.

Parte F Utilizando a chave de simbolização dada, traduza cada sentença do português para a LQ.

UD: pessoas e pratos num churrasco.

Ax: x acabou.

Mx: x está sobre a mesa.

Cx: x é comida.

Px: x é pessoa.

Gxy: x gosta de y.

e: Eli

f: Francesca

g: a guacamole

1. Toda comida está sobre a mesa.
2. Se a guacamole não acabou, então ela está sobre a mesa.
3. Todos gostam de guacamole.
4. Se qualquer um gosta de guacamole, então Eli também.
5. Francesca gosta apenas dos pratos que acabaram.
6. Francesca não gosta de ninguém e ninguém gosta de Francesca.
7. Eli gosta de qualquer um que goste de guacamole.

8. Eli gosta de qualquer um que goste das pessoas que ele gosta.

9. Se há alguém sobre a mesa, então toda comida deve ter acabado.

⋆ **Parte G** Utilizando a chave de simbolização dada, traduza cada sentença do português para a LQ.

UD: pessoas.

Dx: x dança balé.

Mx: x é mulher.

Hx: x é homem.

Fxy: x é filho de y.

Ixy: x é irmão y.

e: Elmer

j: Jane

p: Patrick

1. Todos filhos de Patrick são dançarinos de balé.
2. Jane é filha de Patrick.
3. Patrick tem uma filha.
4. Jane é filha única.
5. Todas as filhas de Patrick são dançarinas de balé.
6. Patrick não tem filho algum.
7. Jane é a sobrinha de Elmer.
8. Patrick é o irmão de Elmer.
9. Os irmãos de Patrick não tem filho algum.
10. Jane é uma tia.
11. Todos que dançam balé têm uma irmã que também dança balé.

12. Todo homem que dança balé é o filho de alguém que dança balé.

Parte H Identifique quais variáveis estão ligadas e quais estão livres.

1. $\exists x Lxy \& \forall y Lyx$
2. $\forall x Ax \& Bx$
3. $\forall x(Ax \& Bx) \& \forall y(Cx \& Dy)$
4. $\forall x \exists y[Rxy \rightarrow (Jz \& Kx)] \vee Ryx$
5. $\forall x_1(Mx_2 \leftrightarrow Lx_2x_1) \& \exists x_2 Lx_3x_2$

$\star$ **Parte I**

1. Identifique quais das seguintes são instâncias de substituição de $\forall x Rcx$: $Rac, Rca, Raa, Rcb, Rbc, Rcc, Rcd, Rcx$.
2. Identifique quais das seguintes são instâncias de substituição de $\exists x \forall y Lxy$: $\forall y Lby, \forall x Lbx, Lab, \exists x Lxa$.

Parte J Utilizando a chave de simbolização dada, traduza cada sentença do português para a LQ com identidade. A última sentença é ambígua e pode ser traduzida de duas maneiras; você deve fazer as duas. (Dica: a identidade é necessária apenas nas últimas quatro sentenças).

UD: pessoas.

Sx: x sabe a combinação do cofre.

Ex: x é espião.

Vx: x é vegetariano.

Cxy: x confia em y.

h: Hofthor

i: Ingmar

1. Hofthor é um espião, mas nenhum vegetariano é espião.

2. Ninguém sabe a combinação do cofre, a menos que Ingmar saiba.
3. Nenhum espião sabe a combinação do cofre.
4. Nem Hofthor nem Ingmar são vegetarianos.
5. Hofthor confia em vegetarianos.
6. Todos que confiam em Ingmar confiam em um vegetariano.
7. Todos que confiam em Ingmar confiam em alguém que confia em um vegetariano.
8. Apenas Ingmar sabe a combinação do cofre.
9. Ingmar confia em Hofthor, mas em nada mais.
10. A pessoa que sabe a combinação do cofre é um vegetariano.
11. A pessoa que sabe a combinação do cofre não é um espião.

⋆ **Parte K** Utilizando a chave de simbolização dada, traduza cada sentença do português para a LQ com a identidade. As últimas duas sentenças são ambíguas e podem ser traduzidas de duas maneiras; você deve fazer as duas.

UD: cartas num deck padrão.

Rx: x é preta.

Px: x é paus.

Dx: x é dois.

Vx: x é valete.

Hx: x é rei de ouros.

Fx: x está de perfil.

Sx: x é coringa.

1. Todas cartas de paus são pretas.
2. Não há coringa.
3. Há pelo menos dois paus.

4. Há mais de um valete de perfil.
5. Há no máximo dois valetes de perfil.
6. Há dois valetes pretos.
7. Há quatro cartas de dois.
8. O dois de paus é uma carta preta.
9. O valete de perfil e o rei são coringas.
10. Se o dois de paus é um coringa, então há exatamente um coringa.
11. O rei não é um valete.
12. O dois de paus não é o rei.

Parte L Utilizando a chave de simbolização dada, traduza cada sentença do português para a LQ com identidade. As últimas duas sentenças são ambíguas e podem ser traduzidas de duas maneiras; você deve fazer as duas.

UD: animais no mundo.

Bx: x está no pasto do fazendeiro Brown.

Cx: x é cavalo.

Px: x é Pegáso.

Ax: x tem asas.

1. Há pelo menos três cavalos no mundo.
2. Há pelo menos três animais no mundo.
3. Há mais de um cavalo no pasto do fazendeiro Brown.
4. Há três cavalos no pasto do fazendeiro Brown.
5. Há pelo menos uma criatura alada no pasto do fazendeiro Brown; qualquer outra criatura no pasto é não alada.
6. O Pégaso é um cavalo alado.
7. O animal no pasto do fazendeiro Brown não é um cavalo.
8. O cavalo no pasto do fazendeiro Brown não tem asas.

Capítulo 5

Semântica formal

Neste capítulo, descreveremos uma semântica formal para LS e para LQ. A palavra 'semântica' vem da palavra grega para 'signo' e significa 'relacionado ao significado.' Desse modo, uma semântica formal será uma explicação matemática do significado na linguagem formal.

Uma linguagem formal é construída por dois tipos de elementos: símbolos lógicos e símbolos não lógicos. Conectivos (como '&') e quantificadores (como '$\forall$') são símbolos lógicos, porque o significados deles é especificado na linguagem formal. Ao escrever uma chave de simbolização, você não está autorizado a mudar o significado dos símbolos lógicos. Você não pode dizer, por exemplo, que o símbolo '$\neg$' significará 'não' em um argumento e 'talvez' em outro. O símbolo '$\neg$' sempre significa negação lógica. Ele é usado para traduzir a palavra do português 'não', mas é um símbolo de uma linguagem formal e é definido por suas condições de verdade.

As letras sentenciais em LS são símbolos não lógicos, porque seus significados não são definidos pela estrutura lógica da LS. Ao traduzirmos um argumento do português para a LS, por exemplo, a letra sentencial M não tem seu significado fixado *a priori*; na verdade, apenas fornecemos uma chave de simbolização que diz como M deverá ser interpretada naquele argumento. Na LQ, os predicados e as constantes são símbolos não lógicos.

Ao traduzirmos do português para uma linguagem formal, fornecemos chaves de simbolização que foram interpretações de todos os termos não lógicos que usamos na tradução. Uma INTERPRETAÇÃO dá um significado para todos os elementos não lógicos de uma linguagem.

É possível dar diferentes interpretações que não fazem nenhuma diferença formal. Na LS, por exemplo, podemos dizer que D significa 'Hoje é terça'; podemos dizer, em vez disso, que D significa 'Hoje é o dia depois de segunda-feira.' Essas são duas interpretações diferentes, porque elas usam diferentes sentenças portuguesas para o significado de D. Apesar disso, formalmente, não há nenhuma diferença entre elas. Tudo que importa, após termos simbolizado essas sentenças, é se elas são verdadeiras ou falsas. Para podermos caracterizar o que faz diferença na linguagem formal, precisamos saber o que faz uma sentença verdadeira ou falsa. Para isso, precisamos de uma caracterização formal da *verdade*.

Quando demos definições para uma sentença de LS e para uma de LQ, nós fizemos uma distinção entre a LINGUAGEM OBJETO e a METALINGUAGEM. A linguagem objeto é a linguagem da qual *falamos sobre*: ou LS ou LQ. A metalinguagem é linguagem que utilizamos para falar da LINGUAGEM OBJETO: português, suplementada com um pouco de jargão matemático. Será importante manter essa distinção em mente.

5.1 Semântica para LS

Esta seção fornece uma caracterização formal e rigorosa da *verdade* na LS que dá embasamento ao que já sabemos por fazer tabelas de verdade. Nós pudemos usar as tabelas de verdade para testar confiavelmente se uma sentença era uma tautologia na LS, ou se duas sentenças eram equivalentes, ou se um argumento era válido etc. Por exemplo: $\mathcal{A}$ é uma tautologia na LS se ocorrer V em todas as linhas de uma tabela de verdade completa.

Isso funcionou porque cada linha de uma tabela de verdade corresponde a uma maneira como o mundo pode ser. Nós consideramos todas as combinações possíveis de 1 e 0 para as letras sentenciais que faziam alguma diferença nas sentenças que tratamos. As tabelas de verdade nos permitiram determinar o que aconteceria dadas aquelas diferentes combinações.

Uma vez que construímos uma tabela de verdade, os símbolos '1' e '0' estão separados dos seus significados metalinguísticos de 'verdadeiro' e 'falso'. Nós interpretamos '1' como significando 'verdadeiro', mas as propriedades formais de 1 são definidas pelas tabelas de verdade específicas para os vários conectivos. Os símbolos, em uma tabela de verdade, têm um significado formal que podemos determinar em termos de como os conectivos operam. Por exemplo, se A tem valor 1, então $\neg A$ tem valor 0.

Em resumo: verdade em LS é somente a atribuição de um 1 ou um 0.

Para definirmos formalmente a verdade em LS, então queremos uma função que designe 1 ou 0 para cada uma das sentenças da LS. Podemos interpretar essa função como uma definição de verdade para a LS, se ela fixar 1 para todas as sentenças verdadeiras da LS e 0 para todas as sentenças falsas. Chame essa função 'v' (para 'valoração'). Queremos que v seja uma função de forma que, para qualquer sentença $\mathcal{A}$, $v(\mathcal{A}) = 1$, se $\mathcal{A}$ é verdadeira e, $v(\mathcal{A}) = 0$, se $\mathcal{A}$ é falsa.

Lembre-se que a definição recursiva de uma fbf para a LS tinha duas etapas: a primeira dizia que sentenças atômicas (letras sentenciais isoladas) são fbfs. A segunda permitiu que mais fbfs fossem construídas a partir de fbfs mais básicas. Existiram condições para definição de todos os conectivos sentenciais. Por exemplo, se $\mathcal{A}$ é uma fbf, então $\neg\mathcal{A}$ é uma fbf.

Nossa estratégia para definir a função de verdade, v, também terá duas etapas. A primeira etapa lidará com a verdade das sentenças atômicas; a segunda, com a verdade das sentenças compostas.

Verdade em LS

Como podemos definir verdade para uma sentença atômica na LS? Considere, por exemplo, a sentença M. Sem uma interpretação, não podemos dizer se M é ou verdadeira ou falsa; ela pode significar qualquer coisa. Se usarmos M para simbolizar 'A lua orbita a Terra', então M será verdadeira. Se usarmos M para simbolizar 'A lua é um nabo gigante', então M é falso.

Ademais, a maneira de você descobrir se M é ou não verdadeira depende do que M significa. Se M significasse 'Hoje é segunda-feira,' então você precisaria checar um calendário. Se M significasse 'A lua de Júpiter, Io, tem atividade vulcânica significante,' então você precisaria checar um texto astronômico — e os astrônomos sabem disso, porque enviaram satélites para observar Io.

Quando damos uma chave de simbolização para LS, nós fornecemos uma interpretação das letras sentenciais que usamos. A chave dá uma sentença em língua portuguesa para cada letra sentencial que utilizamos. Dessa forma, a interpretação especifica o quê que cada uma das letras sentenciais *significa*.

Entretanto, isso não é o suficiente para determinar se aquela sentença é verdadeira ou não. As sentenças sobre a lua, por exemplo, exigem que você saiba um pouco de astronomia básica. Imagine uma criança pequena que se convenceu que a lua é um nabo gigante. Ela poderia entender o quê que a sentença 'A lua é um nabo gigante' significa, mas erroneamente acreditaria que essa seria verdadeira.

Considere outro exemplo: Se M significar 'É manhã agora', então se isso será ou verdadeiro ou falso, dependerá de quando você está lendo isso. Eu sei o que a sentença significa, mas — desde que eu não sei quando você estará lendo-a — eu não sei se ela é ou verdadeira ou falsa.

Sendo assim, apenas uma interpretação não determina se uma sentença será verdadeira ou falsa. Verdade ou falsidade dependem também do que o mundo é. Se M significasse 'A lua é um nabo gigante' e a lua fosse um nabo gigante, então M seria verdadeira. Em síntese, a verdade ou a falsidade é determinada por uma interpretação *mais* o modo como o mundo é.

INTERPRETAÇÃO + ESTADO DO MUNDO $\Longrightarrow$ VERDADE/FALSIDADE

Ao prover uma definição lógica da verdade, não poderemos dar uma relato de como uma sentença atômica se torna verdadeira ou falsa no mundo. Em vez disso, nós introduziremos uma *atribuição de valor de verdade.* Formalmente, isso será uma função que nos dirá o valor de verdade de todas as sentenças atômicas. Chame essa função 'a' (para 'atribuição'). Nós definiremos a que para todas letras sentenciais $\mathcal{P}$, do tipo

$$a(\mathcal{P}) = \begin{cases} 1 & \text{se } \mathcal{P} \text{ é verdadeira,} \\ 0 & \text{caso contrário.} \end{cases}$$

Isso significa que a toma qualquer sentença da LS e a atribui ou um ou zero; um se a sentença for verdadeira, zero se a sentença for falsa. Os detalhes da função a são determinados pelo significado das letras sentenciais, combinadas com o estado do mundo. Se E significar 'Está escuro lá fora', então $a(E) = 1$, à noite ou em uma tempestade pesada, enquanto que $a(E) = 0$, num dia claro.

Você pode considerar a como uma linha de uma tabela de verdade. Ao passo que, uma linha de tabela de verdade atribui um valor de verdade para poucas sentenças atômicas, a atribuição do valor de verdade atribui um valor para toda sentença atômica de LS. Existem infinitas letras sentenciais e a atribuição de valor de verdade dá um valor a cada uma delas. Ao construirmos uma tabela de verdade, nós nos preocupamos apenas com as letras sentenciais que afetam o valor de verdade das sentenças que nos interessam. Consequentemente, nós ignoramos o resto. Estritamente falando, toda linha em uma tabela de verdade apresenta uma atribuição *parcial* do valor de verdade.

É importante notar que a atribuição do valor de verdade, a, não é parte da linguagem da LS. Pelo contrário, ela é parte do maquinário matemático que estamos utilizando para descrever a LS. Ela expressa quais sentenças atômicas são verdadeiras e quais são falsas. Agora, definiremos a função de verdade, v, usando a mesma estrutura recursiva que utilizamos para definir uma fbf da LS.

1. Se $\mathcal{A}$ é uma letra sentencial, então $v(\mathcal{A}) = a(\mathcal{A})$
2. Se $\mathcal{A}$ é $\neg\mathcal{B}$, para alguma sentença $\mathcal{B}$, então

$$v(\mathcal{A}) = \begin{cases} 1 & \text{se } v(\mathcal{B}) = 0, \\ 0 & \text{caso contrário.} \end{cases}$$

3. Se $\mathcal{A}$ é $(\mathcal{B}\&\mathcal{C})$, para algumas sentenças $\mathcal{B}$ e $\mathcal{C}$, então

$$v(\mathcal{A}) = \begin{cases} 1 & \text{se } v(\mathcal{B}) = 1 \text{ e } v(\mathcal{C}) = 1, \\ 0 & \text{caso contrário.} \end{cases}$$

Pode parecer que essa definição é circular porque utiliza a palavra 'e' na tentativa de definir 'e'. Note, entretanto, que isso não é uma definição da palavra portuguesa 'e'; é uma definição de verdade para sentenças de LS contendo o símbolo lógico '&'. Nós definimos a verdade para sentenças da linguagem objeto que contém o símbolo '&', utilizando a palavra da metalinguagem 'e.' Não há nada de circular nisso.

4. Se $\mathcal{A}$ é $(\mathcal{B} \vee \mathcal{C})$, para algumas sentenças $\mathcal{B}$ e $\mathcal{C}$, então

$$v(\mathcal{A}) = \begin{cases} 0 & \text{se } v(\mathcal{B}) = 0 \text{ e } v(\mathcal{C}) = 0, \\ 1 & \text{caso contrário.} \end{cases}$$

5. Se $\mathcal{A}$ é $(\mathcal{B} \to \mathcal{C})$, para algumas sentenças $\mathcal{B}$ e $\mathcal{C}$, então

$$v(\mathcal{A}) = \begin{cases} 0 & \text{se } v(\mathcal{B}) = 1 \text{ e } v(\mathcal{C}) = 0, \\ 1 & \text{caso contrário.} \end{cases}$$

6. Se $\mathcal{A}$ é $(\mathcal{B} \leftrightarrow \mathcal{C})$, para algumas sentenças $\mathcal{B}$ e $\mathcal{C}$, então

$$v(\mathcal{A}) = \begin{cases} 1 & \text{se } v(\mathcal{B}) = v(\mathcal{C}), \\ 0 & \text{caso contrário.} \end{cases}$$

Já que a definição de v tem a mesma estrutura que a definição de uma fbf, nós sabemos que v atribui um valor para toda fbf da LS. E uma vez que as sentenças da LS e as fbfs da LS são as mesmas, isso significa que v retorna o valor de verdade de toda sentença da LS.

A verdade em LS é sempre relativa à atribuição de algum valor de verdade, porque a definição da verdade para LS não diz se determinada sentença é verdadeira ou falsa. Em vez disso, ela diz como a verdade daquela sentença se relaciona com a atribuição de um valor de verdade.

Outros conceitos em LS

Temos trabalhado, até agora, com a LS sem uma definição precisa de 'tautologia', 'contradição' e etc. As tabelas de verdades forneceram-nos um meio para *checar se* uma sentença era uma tautologia na LS, mas elas não definiram o que *significa* ser uma tautologia na LS. Nós daremos definições desses conceitos de LS em termos de implicação.

A relação semântica, '$\mathcal{A}$ implica $\mathcal{B}$' significa que não há atribuição do valor de verdade para o qual $\mathcal{A}$ seja verdadeiro e $\mathcal{B}$ falso. Dito de outro modo, isso significa que $\mathcal{B}$ é verdadeiro para toda e qualquer atribuição de valor de verdade no qual $\mathcal{A}$ é verdadeiro. Nós abreviamos isso com um símbolo chamado de dupla catraca: $\mathcal{A} \models \mathcal{B}$ significa '$\mathcal{A}$ implica semanticamente$\mathcal{B}$.'

Podemos falar sobre implicações entre mais do que duas sentenças:

$$\{\mathcal{A}_1, \mathcal{A}_2, \mathcal{A}_3, \ldots\} \models \mathcal{B}$$

significa que não há atribuição de valor de verdade para o qual todas as sentenças no conjunto $\{\mathcal{A}_1, \mathcal{A}_2, \mathcal{A}_3, \ldots\}$ são verdadeiras e $\mathcal{B}$ é falso.

Podemos também usar o símbolo com apenas uma sentença: $\models \mathcal{C}$ significa que $\mathcal{C}$ é verdadeiro para todas as atribuições de valor de verdade. Isso é equivalente a dizer que a sentença é implicada por qualquer coisa.

O símbolo da dupla catraca nos permite dar uma definição concisa para vários conceitos da LS:

> UMA TAUTOLOGIA EM LS é uma sentença $\mathcal{A}$ do tipo $\models \mathcal{A}$.
> UMA CONTRADIÇÃO EM LS é uma sentença $\mathcal{A}$ do tipo $\models \neg\mathcal{A}$.
> UMA sentença é CONTIGENTE EM LS se e somente se ela não é nem uma tautologia nem uma contradição.
> Um argumento "$\mathcal{P}_1, \mathcal{P}_2, \ldots, \therefore \mathcal{C}$" é VÁLIDO NA LS se e somente se $\{\mathcal{P}_1, \mathcal{P}_2, \ldots\} \models \mathcal{C}$.
> Duas sentenças $\mathcal{A}$ e $\mathcal{B}$ são LOGICAMENTE EQUIVALENTES NA LS se e somente se, ambos $\mathcal{A} \models \mathcal{B}$ e $\mathcal{B} \models \mathcal{A}$.

Consistência lógica é mais difícil de ser definida em termos de implicação semântica. Em vez disso, a difiniremos da seguinte maneira:

> O conjunto $\{\mathcal{A}_1, \mathcal{A}_2, \mathcal{A}_3, \ldots\}$ é CONSISTENTE NA LS se e somente se existe ao menos uma atribuição de valor de verdade para o qual todas as sentenças são verdadeiras. O conjunto é INCONSISTENTE NA LS se e somente se não há tal atribuição.

5.2 Interpretação e modelos em LQ

Em LS, uma interpretação ou chave de simbolização especifica o que cada uma das sentenças significa. A interpretação de uma sentença juntamente com o estado de coisas determina se a letra sentencial é verdadeira ou falsa. Já que as unidades básicas são letras sentenciais, uma interpretação só é importante na medida em que torna as letras sentenciais verdadeiras ou falsas. Formalmente, a semântica da LS é dada estritamente em termos de atribuição de valores de verdade. Duas interpretações são idênticas, do ponto de vista formal, se elas fizerem a mesma atribuição de valores de verdade.

O que é uma interpretação em LQ? Como uma chave de simbolização para LQ, uma interpretação requer um UD, um significado esquemático para cada um dos predicados e objetos que são selecionados por cada uma das constantes. Por exemplo:

UD: personagens das histórias em quadrinhos

Lx: x luta contra o crime

b: Batman

w: Bruce Wayne

Considere a sentença Lb. A sentença é verdadeira nessa interpretação, todavia — assim como na LS — ela não é verdadeira *somente por causa* da interpretação. A maioria das pessoas na nossa cultura sabem que o Batman luta contra o crime, mas para isso é necessário que elas tenham um conhecimento mínimo sobre histórias em quadrinhos. A sentença Lb é verdadeira por causa da interpretação *juntamente com* alguns fatos sobre histórias em quadrinhos. Isso é especialmente óbvio quando consideramos Lw. Bruce Wayne é a identidade secreta do Batman nas histórias em quadrinhos — a afirmação da identidade $b = w$ é verdadeira — então Lw é verdadeira. Todavia, já que sua identidade é *secreta*, outros personagens não sabem que Lw é verdadeira, apesar deles saberem que Lb é verdadeira.

Poderíamos tentar caracterizar isso como uma atribuição de valor de verdade, como fizemos para a LS. A atribuição de valor de verdade assinalaria 0 ou 1 para cada fbf atômica: Lb, Lw e assim por diante. Entretanto, se fôssemos fazer isso, poderíamos traduzir as sentenças da LQ para a LS ao substituir Lb e Lw por letras sentenciais. Poderíamos então nos basear na definição de verdade para LS, mas ao custo de ignorar toda a estrutura lógica de predicados e termos. Ao escrever uma chave de simbolização para LQ, não damos definições separadas para Lb e Lw. Ao invés disso, damos os significados para L, b e w. Isso é essencial, pois pretendemos utilizar quantificadores. Não há maneira adequada de se traduzir $\forall x Lx$ para a LS.

Em vista disso, gostaríamos de uma contraparte formal para uma interpretação de predicados e constantes, não apenas de sentenças. Não podemos utilizar uma atribuição de valor de verdade para isso, porque um predicado não é verdadeiro nem falso. Na interpretação acima, L é verdadeira *do* Batman (i.e.,

Lb é verdadeira), mas não faz sentido algum em se perguntar se L é verdadeira em si mesma. Seria como perguntar-se se a parte '... luta contra o crime' é verdadeira.

O que uma interpretação faz a um predicado, se ela não o faz verdadeiro ou falso? Uma interpretação nos ajuda a selecionar os objetos aos quais o predicado se aplica. A interpretação de Lx ao significar 'x luta contra o crime', seleciona Batman, Super-homem, Homem-Aranha e outros heróis que são coisas que são Ls. Formalmente, esse é um conjunto de membros do UD ao qual o predicado se aplica; esse conjunto é chamado de EXTENSÃO do predicado.

Muitos predicados têm extensões indefinidamente amplas. Seria impraticável tentar e escrever individualmente todos aqueles que lutam contra o crime nas histórias em quadrinhos, ao invés disso, utilizamos uma expressão em português para interpretar o predicado. De alguma maneira isso é impreciso, porque apenas a interpretação não diz qual dos membros do UD estão na extensão do predicado. Para descobrir se um membro particular do UD está na extensão do predicado (por exemplo, para se descobrir se Raio Negro luta contra o crime), você precisa saber sobre histórias em quadrinhos. Em geral, a extensão de um predicado é resultado de uma interpretação *juntamente com* alguns fatos.

Algumas vezes é possível enumerar todas as coisas que estão na extensão de um predicado. Ao invés de escrevermos uma sentença esquemática em português, podemos escrever a extensão como um conjunto de coisas. Suponha que desejemos adicionar o predicado unário M à chave acima. Queremos que Mx signifique 'x vive na Mansão Wayne', então escrevemos a extensão como um conjunto de personagens:

$$\text{extensão}(M) = \{\text{Bruce Wayne, o mordomo Alfred, Dick Grayson}\}$$

Dada essa interpretação, não é necessário que você saiba coisa alguma sobre histórias em quadrinhos. Mw é verdadeira: Bruce Wayne somente foi especificado como uma das coisas que é M. Do mesmo modo, $\exists xMx$ é obviamente verdadeira nessa interpretação: há no mínimo um membro do UD que é um M — na verdade, existem três deles.

E sobre a sentença $\forall xMx$? Essa sentença é falsa, porque não é verdadeiro que todos os membros do UD são M. É necessário um conhecimento ainda mais superficial sobre histórias em quadrinhos para saber que existem outros personagens além desses três. Apesar de termos especificado a extensão de M

de maneira formalmente precisa, ainda especificamos o UD com uma descrição em português. Do ponto de vista formal, um UD é apenas um conjunto de membros.

O significado formal de um predicado é determinado por sua expressão, mas o que devemos dizer sobre constantes como b e w? O significado de uma constante determina qual dos membros do UD é selecionado pela constante. O indivíduo que é selecionado pela constante é chamado de REFERENTE daquela constante. Ambos b e w têm o mesmo referente, já que ambas se referem ao mesmo personagem de histórias em quadrinho. Você pode considerar uma letra de constante como um nome e o referente como a coisa nomeada. Em português, podemos utilizar os nomes diferentes 'Batman' e 'Bruce Wayne' para se referir ao mesmo personagem de história em quadrinho. Nessa interpretação, podemos utilizar constantes diferentes 'b' e 'w' para se referir ao mesmo membro do UD.

Conjuntos

Utilizamos chaves '{' e '}' para denotar conjuntos. Os membros do conjunto podem ser listados em qualquer ordem e são separados por vírgulas. O fato de conjuntos poderem ser definidos em qualquer ordem é importante, porque isso significa que {foo, bar} e {bar, foo} são o mesmo conjunto.

É possível ter um conjunto no qual não exista membro algum. Isso é chamado de CONJUNTO VAZIO. O conjunto vazio é algumas vezes escrito como {}, mas geralmente é escrito através do símbolo único $\emptyset$.

Modelos

Como vimos, uma interpretação na LQ é formalmente significativa na medida em que determina um UD, uma extensão para cada predicado e um referente para cada constante. Nós chamamos essa estrutura formal de MODELO em LQ.

Para ver como isso funciona, considere esta chave de simbolização:

UD: Pessoas que participaram dos Três Patetas

Cx: x tinha cabelo na cabeça

f: Senhor Fine

Se você não sabe coisa alguma sobre os Três Patetas, você não poderá dizer quais das sentenças da LQ são verdadeiras nessa interpretação. Talvez você se lembre somente de Larry, Curly e Moe. A sentença Cf é verdadeira ou falsa? Depende de qual dos patetas é o Senhor Fine.

Qual o modelo que corresponde a essa interpretação? Seis pessoas participaram do Três Patetas ao longo dos anos, então o UD terá seis membros: Larry Fine, Moe Howard, Curly Howard, Shemp Howard, Joe Besser e Curly Joe DeRita. Curly Howard, Joe Besser e Curly Joe eram os únicos patetas completamente calvos. O resultado é esse modelo:

UD={ Larry Fine, Moe Howard, Curly Howard, Shemp Howard, Joe Besser, Curly Joe DeRita}

extensão(C)={Larry, Moe, Shemp}

referente(f)=Larry

Não é necessário saber coisa alguma sobre os Três Patetas para poder avaliar se sentenças são verdadeiras ou falsas nesse *modelo*. Cf é verdadeira, já que o referente de f (Larry) está na extensão de C. Ambos $\exists xCx$ e $\exists x\neg Cx$ são verdadeiras, já que há pelo menos um membro do UD que está na extensão de C e também há pelo menos um membro que não está na extensão de C. Dessa forma, o modelo captura toda significação formal da interpretação.

Agora, considere esta interpretação:

UD: todos os números naturais menores que 10

Px: x é par

Nx: x é negativo

Mxy: x é menor que y

Vxyz: x vezes y é igual a z

Qual é o modelo que coincide com essa interpretação? O UD é o conjunto $\{1, 2, 3, 4, 5, 6, 7, 8, 9\}$.

A extensão de um predicado unário como P ou N é somente o subconjunto do UD no qual o predicado é verdadeiro. A grosso modo, a extensão do predicado P é o conjunto de Ps no UD. A extensão de P é o subconjunto

$\{2, 4, 6, 8\}$. Existem muitos números pares além desses quatro, mas esses são os únicos membros do UD que são pares. Não há números negativos no UD, então N tem uma extensão vazia; i.e., extensão$(N) = \emptyset$.

A extensão de um predicado binário como M é de alguma forma desconcertante. Parece que a extensão de M deveria conter 1, já que 1 é menor que todos os outros números; ela deve conter 2, já que dois é menor que todos os outros números exceto 1; e assim por diante. Todo membro do UD exceto 9 é menor que algum membro do UD. O que aconteceria se simplesmente escrevêssemos a extensão$(M) = \{1, 2, 3, 4, 5, 6, 7, 8\}$?

O problema é que os conjuntos podem ser escritos em qualquer ordem, então isso seria o mesmo que escrever a extensão$(M) = \{8, 7, 6, 5, 4, 3, 2, 1\}$. Isso não nos diz quais membros do conjunto são menores que quais membros do conjunto.

Precisamos de um modo para mostrar que 1 é menor que 8, mas 8 não é menor que 1. A solução é fazer com que a extensão de M consista em pares de números. Um PAR ORDENADO é como um conjunto com dois membros, exceto que a ordem desses é importante. Escrevemos pares ordenados com sinais de menor que ‘<’ e maior que ‘>’. O par ordenado <bar,foo> é diferente do par ordenado <foo, bar>. A extensão de M é uma coleção de pares ordenados, todos os pares de números no UD tal que o primeiro número seja menor que o segundo. Ela é completamente expressa por:

$$\text{extensão}(M) = \{<1,2>, <1,3>, <1,4>, <1,5>, <1,6>, <1,7>, <1,8>, <1,9>, <2,3>, <2,4>, <2,5>, <2,6>, <2,7>, <2,8>, <2,9>, <3,4>, <3,5>, <3,6>, <3,7>, <3,8>, <3,9>, <4,5>, <4,6>, <4,7>, <4,8>, <4,9>, <5,6>, <5,7>, <5,8>, <5,9>, <6,7>, <6,8>, <6,9>, <7,8>, <7,9>, <8,9>\}$$

Predicados ternários funcionarão similarmente; a extensão de um predicado ternário é um conjunto de triplas ordenadas nas quais o predicado é verdadeiro daquelas três coisas *naquela ordem*. Assim, a extensão de T nesse modelo conterá triplas ordenadas como $<2, 4, 8>$, porque $2 \times 4 = 8$.

De forma geral, a extensão de um predicado de n-lugares é um conjunto de todas as ênuplas ordenadas $\langle a_1, a_2, \ldots, a_n \rangle$ tal que $a_1 - a_n$ sejam membros de todo UD e o predicado é verdadeiro de $a_1 - a_n$ naquela ordem.

5.3 Semântica para identidade

A identidade é um predicado especial para LQ. A escrevemos um pouco diferente dos outros predicados de dois lugares: $x = y$ ao invés de Ixy. Tampouco é necessário escrevê-la na chave de simbolização. A sentença $x = y$ sempre significa 'x é idêntico a y' e não pode ser interpretada de outro modo. Da mesma maneira, quando você constrói um modelo não é preciso que se diga quais pares ordenados cairão na extensão do predicado de identidade. Ela contém somente o par ordenado de cada objeto no UD consigo mesmo.

A sentença $\forall xIxx$, que contém um predicado binário comum , é contingente. Se ela é verdadeira ou não para uma dada interpretação dependerá de como você interpreta I, e se ela é verdadeira ou não em um modelo, por sua vez, depende da extensão de I.

A sentença $\forall x\, x = x$ é uma tautologia. A extensão da identidade sempre a fará verdadeira.

Note que apesar da identidade ter sempre a mesma interpretação, nem sempre ela tem a mesma extensão. A extensão da identidade depende do UD. Se o UD, num dado modelo, é o conjunto $\{\text{Douglas}\}$, então extensão(=) naquele modelo é $\{< \text{Douglas}, \text{Douglas} >\}$. Se o UD é o conjunto $\{\text{Douglas}, \text{Felipe}\}$, então extensão(=) naquele modelo é $\{< \text{Douglas}, \text{Douglas} >, < \text{Felipe}, \text{Felipe} >\}$. E assim por diante.

Se o referente de duas constantes é o mesmo, então qualquer coisa que é verdadeira de uma também o é da outra. Por exemplo, se referente(a) = referente (b), então $Aa \leftrightarrow Ab$, $Ba \leftrightarrow Bb$, $Ca \leftrightarrow Cb$, $Rca \leftrightarrow Rcb$, $\forall xRxa \leftrightarrow \forall xRxb$ e assim por diante para quaisquer duas sentenças que contenham a e b. Todavia, o inverso não é verdadeiro, pois é possível que algo que seja verdadeiro de a também seja verdadeiro de b, apesar de a e b terem referentes distintos.

Isso pode parecer estranho, mas é fácil construir um modelo que mostre isso. Considere este modelo:

$$\text{UD}=\{\text{Rosencrantz}, \text{Guildenstern}\}$$

$$\text{referente}\,(a) = \text{Rosencrantz}$$

$$\text{referente}\,(b) = \text{Guildenstern}$$

$$\text{Para todos predicados } \mathcal{P},\ \text{extensao}\,(\mathcal{P}) = \emptyset$$

$$\text{extensão}\,(=) = \{< \text{Rosencrantz}, \text{Rosencrantz} >, < \text{Guildenstern}, \text{Guildenstern} >\}$$

Isso especifica uma extensão para cada predicado de LQ: todos os inumeráveis predicados são vazios. Isso significa que ambas Aa e Ab são falsas e equivalentes; Ba e Bb são falsas; e assim por diante para quaisquer duas sentenças que contenham a e b. Ainda assim, a e b referem-se a coisas diferentes. Escrevemos a extensão de identidade para deixar isso claro: o par ordenado $\langle \text{referente}\,(a)\,, \text{referente}\,(b)\rangle$ não está contido nela. Nesse modelo, $a = b$ é falsa e $a \neq b$ é verdadeira.

5.4 Trabalhando com modelos

Utilizaremos a dupla catraca para LQ tal qual fizemos para LS. '$\mathcal{A} \vDash \mathcal{B}$' significa que '$\mathcal{A}$ implica $\mathcal{B}$'. Quando $\mathcal{A}$ e $\mathcal{B}$ são duas sentenças de LQ, $\mathcal{A} \vDash \mathcal{B}$ significa que não há modelo no qual $\mathcal{A}$ é verdadeiro e $\mathcal{B}$ falso. $\vDash \mathcal{A}$ significa que $\mathcal{A}$ é verdadeira em todo modelo.

Isso nos permite dar definições para vários conceitos na LQ. Por estarmos utilizando o mesmo símbolo, essas definições parecerão similares às definições na LS. Todavia, lembre-se que as definições em LQ são em termos de *modelos*, ao invés de atribuições de valores de verdade.

Uma TAUTOLOGIA NA LQ é uma sentença $\mathscr{A}$ que é verdadeira em todo modelo; i.e., $\vDash \mathscr{A}$.
Uma CONTRADIÇÃO NA LQ é uma sentença $\mathscr{A}$ que é falsa em todo modelo; i.e., $\vDash \neg\mathscr{A}$.
Uma sentença é CONTINGENTE NA LQ se e somente se ela não é uma tautologia nem uma contradição.
Um argumento "$\mathcal{P}_1, \mathcal{P}_2, \ldots \therefore \mathcal{C}$ " é VÁLIDO NA LQ se e somente se não há modelo no qual todas as premissas sejam verdadeiras e a conclusão falsa; i.e., $\{\mathcal{P}_1, \mathcal{P}_2, \cdots\} \vDash \mathcal{C}$. De outro modo, o argumento será INVÁLIDO NA LQ.
Duas sentenças $\mathscr{A}$ e $\mathscr{B}$ são LOGICAMENTE EQUIVALENTES NA LQ se e somente se ambos $\mathscr{A} \vDash \mathscr{B}$ e $\mathscr{B} \vDash \mathscr{A}$.
O conjunto $\{\mathscr{A}_1, \mathscr{A}_2, \mathscr{A}_3, \cdots\}$ é CONSISTENTE NA LQ se e somente se há no mínimo um modelo no qual todas as sentenças sejam verdadeiras. O conjunto será INCONSISTENTE NA LQ, se e somente se não houver tal modelo.

Construindo modelos

Suponha que desejemos demonstrar que $\forall x Axx \rightarrow Bd$ *não* é uma tautologia. Para isso, é necessário que a sentença não seja verdadeira em todo modelo; i.e., que seja falsa em algum modelo. Se pudermos fornecer apenas um modelo no qual a sentença seja falsa, então teremos demonstrado que a sentença não é uma tautologia.

Como se pareceria tal modelo? Para que $\forall x Axx \rightarrow Bd$ seja falsa, a antecedente, $\forall x Axx$, deve ser verdadeira e a consequente, Bd, falsa.

Para construir tal modelo, nós começamos com um UD. Será mais fácil especificar extensões para predicados se tivermos um UD pequeno, então comecemos com um UD que tenha apenas um membro. Do ponto de vista formal, esse membro único pode ser qualquer coisa. Digamos que seja a cidade de Paris.

Queremos que $\forall x Axx$ seja verdadeira, então desejamos que todos membros do UD estejam emparelhados consigo mesmos na extensão de A; isso significa que a extensão de deve ser $\{< \text{Paris}, \text{Paris} >\}$.

Queremos que Bd seja falso, então o referente de d não pode ser a extensão de B. Daremos a B uma extensão vazia.

Já que Paris é o único membro do UD, isso deve ser o referente de d. O modelo que construímos parece assim:

UD=$\{$Paris$\}$

extensão(A)=$\{<$Paris, Paris$>\}$

extensão(B) $= \emptyset$

referente(d)=Paris

Estritamente falando, um modelo especifica uma extensão para *todo* predicado de LQ e um referente para *toda* constante. Assim, é geralmente impossível escrever um modelo completo. Para isso, seria necessário escrever infinitamente muitas extensões e referentes. Todavia, não precisamos considerar todo predicado para demonstrar que há modelos no qual $\forall x Axx \rightarrow Bd$ é falso. Predicados como H e referentes como f_{13} não fazem diferença quanto à verdade ou falsidade dessa sentença. É suficiente especificar extensões para A e B e um referente para d, como fizemos. Isso nos fornece um *modelo parcial* no qual a sentença é falsa.

Talvez você esteja imaginando: O que o predicado A significa em português? O modelo parcial poderia corresponder a uma interpretação como esta:

UD: Paris

Axy: x está no mesmo país que y

Bx: x foi fundada no século 20

d: a cidade das luzes

Todavia, tudo que esse modelo parcial nos diz é que A é um predicado que é verdadeiro de Paris e Paris. Há inumeráveis predicados em português que possuem essa extensão. Axy pode significar, em vez disso, que 'x é do mesmo tamanho que y' ou 'x e y são ambas cidades.′ Similarmente, Bx é um predicado que não se aplica a Paris; em vez disso, ele pode traduzir 'x está na ilha' ou 'x é um carro supercompacto.' Quando especificamos as extensões de A e B, não especificamos os predicados em português que A e B devem ser utilizados para traduzir. Nós estamos interessados na verdade ou falsidade de $\forall x Axx \rightarrow Bd$ e tudo que é relevante para a verdade ou a falsidade na LQ é a informação no modelo: o UD, as extensões dos predicados e os referentes das constantes.

Podemos facilmente demonstrar que $\forall x Axx \rightarrow Bd$ não é uma contradição. Precisamos apenas de especificar um modelo no qual $\forall x Axx \rightarrow Bd$ é verdadeiro; i.e., um modelo no qual $\forall x Axx$ é falso ou Bd é verdadeiro. Aqui está um modelo parcial:

UD={Paris}

extensão(A)={<Paris, Paris>}

extensão(B)={Paris}

referente(d)=Paris

Demonstramos que $\forall x Axx \rightarrow Bd$ não é tautologia nem contradição. Pela definição de 'contingente na LQ,' isso significa que $\forall x Axx \rightarrow Bd$ é contingente. Em geral, são necessários dois modelos para mostrar que uma sentença é contingente: um no qual a sentença é verdadeira e outro no qual é falsa.

Suponha que queiramos mostrar que $\forall x Sx$ e $\exists x Sx$ não são logicamente equivalentes. Precisamos construir um modelo no qual as duas sentenças têm valores de verdade diferentes; queremos que uma delas seja verdadeira e a outra falsa. Começamos por especificar um UD. Mais uma vez, construíremos um pequeno para que possamos especificar suas extensões facilmente. Precisaremos de no mínimo dois membros. Façamos o UD ser $\{\text{Davi}, \text{Carlos}\}$. (Se tivéssemos escolhido um UD com apenas um membro, ambas as sentenças teriam o mesmo valor de verdade. Para entender o porquê, tente construir alguns modelos parciais de UDs com apenas um membro).

Podemos fazer $\exists x Sx$ verdadeira incluindo algo na extensão de S e podemos fazer $\forall x Sx$ falsa deixando algo fora da extensão de S. Não faz diferença qual incluímos ou qual deixamos fora. Fazendo Davi o único S, temos um modelo parcial que se parece com isso:

UD={Davi, Carlos}

extensão(S)={Davi}

Isso nos mostra que as duas sentenças *não* são logicamente equivalentes.

De volta à pagina 78, dissemos que este argumento seria inválido na LQ:

$(Rc \& K_1 c) \& Tc$

$\therefore\ Tc \& K_2 c$

Para mostrarmos que isso é inválido, precisamos mostrar que há um modelo no qual as premissas são verdadeiras e a conclusão é falsa. Podemos construir tal modelo deliberadamente. Eis uma forma de fazê-lo:

UD={Björk}

extensão(T)={Björk}

extensão(K_1)={Björk}

extensão(K_2) $= \emptyset$

extensão(R)={Björk}

referente(c)=Björk

Similarmente, podemos mostrar que um conjunto de sentenças é consistente por se construir um modelo no qual todas elas são verdadeiras.

Raciocinando sobre todos os modelos

Por um lado, podemos mostrar que uma sentença *não* é uma tautologia apenas fazendo um modelo cuidadosamente escolhido: um no qual a sentença é falsa. Por outro lado, para mostrar que algo é uma tautologia não seria suficiente construir dez, cém ou até mesmo mil modelos no qual a sentença é verdadeira. Algo só é tautológico se for verdadeiro em todo modelo, e existem infinitos modelos. Isso não pode ser evitado simplesmente construindo modelos parciais, porque também existem infinitos modelos parciais.

Por exemplo, considere a sentença $Raa \leftrightarrow Raa$. Existem dois modelos parciais logicamente distintos dessa sentença que têm um UD de membro único. Existem 32 modelos parciais distintos que tem um UD de dois membros. Existem 1526 modelos parciais distintos que tem um UD de três membros. Existem 262.144 modelos parciais distintos que tem um UD de quatro membros. E assim por diante infinitamente. Para mostrar que essa sentença é uma tautologia,

	SIM	NÃO
$\mathcal{A}$ uma é tautologia?	Mostrar que $\mathcal{A}$ deve ser verdadeira em qualquer modelo.	*Construir um modelo* no qual que $\mathcal{A}$ é falsa.
$\mathcal{A}$ é uma contradição?	Mostrar que $\mathcal{A}$ deve ser falsa em qualquer modelo.	*Construir um modelo* no qual $\mathcal{A}$ é verdadeira.
$\mathcal{A}$ é contingente?	*Construir dois modelos*, um no qual $\mathcal{A}$ é verdadeira e outro no qual $\mathcal{A}$ é falsa	Mostrar que $\mathcal{A}$ é uma tautologia ou que $\mathcal{A}$ é uma contradição.
$\mathcal{A}$ e $\mathcal{B}$ são equivalentes?	Mostrar que $\mathcal{A}$ e $\mathcal{B}$ devem ter mesmo valor de verdade em qualquer modelo..	*Construir um modelo* no qual $\mathcal{A}$ e $\mathcal{B}$ tem valores de verdade diferentes.
O conjunto $\mathcal{A}$ é consistente?	*Construir um modelo* no qual todas as sentenças em $\mathcal{A}$ sejam verdadeiras.	Mostrar que as sentenças não poderiam ser verdadeiras em qualquer modelo.
O argumento '$\mathcal{P}, \therefore \mathcal{C}$' é válido?	Mostrar que qualquer modelo no qual $\mathcal{P}$ é verdadeiro deve ser um modelo no qual $\mathcal{C}$ também o é.	*Construir um modelo* no qual $\mathcal{P}$ é verdadeiro e $\mathcal{C}$ falso.

Tabela 5.1: É relativamente fácil responder uma pergunta se você começar construindo um modelo ou dois. É muito mais difícil se você tiver que pensar sobre todos os modelos possíveis. Essa tabela mostra quando construir modelos é suficiente:

precisamos mostrar algo sobre todos esses modelos. Não há como fazer isso tratando cada um dos modelos individualmente.

Ainda assim, $Raa \leftrightarrow Raa$ é obviamente uma tautologia. Podemos provar isso através de um argumento simples:

> Existem dois tipos de modelos: aqueles no quais $\langle \text{referente}(a), \text{referente}(b) \rangle$ estão na extensão de R e aqueles nos quais não estão. No primeiro tipo de modelo, Raa é verdadeiro; pela tabela de verdade para bicondicional, $Raa \leftrightarrow Raa$ é também verdadeira. No segundo tipo de modelo, Raa é falsa; isso faz $Raa \leftrightarrow Raa$ verdadeira. Já que a sentença é verdadeira em ambos tipos de modelos e já que todo modelo é um desses dois tipos, $Raa \leftrightarrow Raa$ é verdadeira em todo modelo. Portanto, é uma tautologia.

Esse argumento é válido e sua conclusão é verdadeira. Todavia, ele não é um argumento em LQ. Ao invés disso, é um argumento em português *sobre* a LQ; esse é um argumento na metalinguagem. Não há procedimento formal para se avaliar ou construir argumentos da linguagem natural como esse. A imprecisão da linguagem natural é a própria razão do porquê começamos a pensar sobre linguagens formais.

Considere a sentença $\forall x\,(Rxx \leftrightarrow Rxx)$, outra tautologia óbvia. Pode ser tentador pensa-la dessa maneira: '$Rxx \leftrightarrow Rxx$ é verdadeira em todo modelo, então $\forall x\,(Rxx \leftrightarrow Rxx)$ também deve ser.' O problema é que $Rxx \leftrightarrow Rxx$ *não* é verdadeira em todo modelo. Ela sequer é uma sentença, então *não* é verdadeira *nem* falsa. Não temos ainda o vocabulário para dizer o que queremos expressar com $Rxx \leftrightarrow Rxx$. Na próxima seção, introduziremos o conceito de *satisfação*; após termos feito isso, estaremos melhor preparados para dar um argumento que $\forall x\,(Rxx \leftrightarrow Rxx)$ é uma tautologia.

É necessário raciocinar sobre uma infinidade de modelos para mostrar que uma sentença é uma tautologia. Do mesmo modo, é necessário raciocinar sobre uma infinidade de modelos para mostrar que uma sentença é uma contradição, que duas sentenças são equivalentes, que um conjunto de sentenças é consistente ou que um argumento é válido. Existem outras coisas que podemos mostrar ao construir cuidadosamente um modelo ou dois. A tabela 5.1 resume quais são.

5.5 Verdade em LQ

Para a LS, dividimos a definição de verdade em duas partes: uma atribuição de valor de verdade para (a) para letras sentenciais e uma função de verdade (v) para todas as sentenças. A função de verdade nos permitiu construir sentenças complexas a partir de letras sentenciais e conectivos.

Da mesma maneira que a verdade em LS é sempre a *verdade, dada uma atribuição de valor de verdade*, a verdade em LQ é sempre *verdadeira em um modelo*. A sentença atômica mais simples de LQ consiste de um predicado unário seguido de uma constante, como Pj. Ela é verdadeira num modelo $\mathbb{M}$ se e somente se o referente de j está na extensão de P em $\mathbb{M}$.

Poderíamos seguir essa interpretação para definir a verdade para todas sentenças atômicas que contenham apenas predicados e constantes: Considere qualquer sentença da forma $\mathcal{R}c_1 \dots c_n$ em que $\mathcal{R}$ é um predicado de lugar-n e os cs são constantes. Ela é verdadeira em $\mathbb{M}$ se e somente se $\langle \text{referente}(c_1), \dots, \text{referente}(c_n) \rangle$ está na extensão de P em $\mathbb{M}$.

Assim, a verdade para as sentenças construídas com conectivos sentenciais seria definida do mesmo modo que fizemos em LS. Por exemplo, a sentença $(Pj \rightarrow Mda)$ é verdadeira em $\mathbb{M}$ se Pj é falsa em $\mathbb{M}$ ou Mda é verdadeira em $\mathbb{M}$.

Infelizmente, essa interpretação falhará ao considerarmos sentenças com quantificadores. Considere $\forall x Px$. Quando ela é verdadeira em $\mathbb{M}$? A resposta não pode depender de se Px é verdadeira ou falsa em $\mathbb{M}$, porque o x em Px é uma variável livre. Px não é uma sentença, portanto, não é verdadeira nem falsa.

Pudemos dar uma definição recursiva de verdade para LQ, porque toda fbf de LS tem um valor de verdade. Isso não é o caso em LQ, então não podemos definir verdade a partir da verdade de sentenças atômicas e, a partir de então, construir sentenças mais complexas. Também precisamos considerar as fórmulas atômicas que não são sentenças. Para fazermos isso definiremos *satisfação*; toda fbf em LQ será satisfeita ou não, mesmo se ela não tiver um valor de verdade. Assim, poderemos definir verdade para sentenças de LQ em termos de satisfação.

Satisfação

A fórmula Px diz, a grosso modo, que x é um dos Ps. Entretanto, isso não pode estar correto, porque x é uma variável e não uma constante. Ela não nomeia qualquer membro específico do UD, em vez disso, seu significado em uma sentença é determinado pelo quantificador que se aplica a ela. Na sentença $\forall xPx$, a variável x deve poder ser substituído por qualquer membro do UD, mas em $\exists xPx$, x deve poder ser substituído por um membro do UD. Já que queremos a definição de satisfação dar conta de Px sem qualquer tipo de quantificador, começaremos por dizer como interpretar uma variável livre como x em Px.

Faremos isso introduzindo uma *atribuição de variável.* Formalmente, essa é uma função que combina cada variável com um membro do UD. Chame essa função de 'a.' (O 'a' é para 'atribuição', mas isso não é o mesmo que a atribuição de valor de verdade que usamos ao definir verdade para LS.)

A fórmula Px é satisfeita num modelo $\mathbb{M}$ por uma atribuição de variável a se e somente se $a\,(x)$, o objeto que a atribui para x está na extensão de P em $\mathbb{M}$.

Quando $\forall xPx$ está satisfeita? Não é suficiente se Px for satisfeita em $\mathbb{M}$ por a, porque isso significa apenas que $a\,(x)$ está na extensão (P). $\forall xPx$ requer que todo outro membro do UD esteja na extensão (P) também.

Para isso, precisaremos de outro elemento de notação técnica: Para qualquer membro Ω do UD e qualquer variável χ, façamos a $a[\Omega|\chi]$ ser a atribuição de variável que atribui Ω para χ, mas que concorda com a em todas outras instâncias. Utilizamos Ω, a letra grega ômega, para enfatizar o fato de que é algum membro do UD e não algum símbolo de LQ. Por exemplo, suponhamos que o UD é presidente dos Estados Unidos. A função $a[\text{Grover Cleveland}|x]$ atribui Grover Cleveland para a variável x, independente do que a atribui para x; para qualquer outra variável, $a[\text{Grover Cleveland}|x]$ concorda com a.

Podemos agora dizer concisamente que $\forall xPx$ é satisfeita num modelo $\mathbb{M}$ por uma atribuição de variável a se e somente se, para todo objeto Ω no UD de $\mathbb{M}$, Px é satisfeito em $\mathbb{M}$ por $a[\Omega|x]$.

Você pode perguntar se essa descrição é circular, pois ela dá as condições de satisfação para a sentença $\forall xPx$ usando a frase 'para todo objeto'. Todavia, é importante se lembrar da diferença entre um símbolo lógico como '$\forall$' e uma palavra em português como 'todo.' A palavra é parte da metalinguagem que utilizamos para definir as condições de satisfação para sentenças da linguagem objeto que contenham o símbolo.

Podemos agora dar uma definição geral de satisfação, a partir dos casos que discutimos. Definimos uma função s (para 'satisfação') em um modelo $\mathbb{M}$ tal que para quaisquer fbfs $\mathcal{A}$ e atribuição de variável a, $s\,(\mathcal{A}, a) = 1$ se $\mathcal{A}$ é satisfeita em $\mathbb{M}$ por a; caso contrário, $s\,(\mathcal{A}, a) = 0$.

1. Se $\mathcal{A}$ é uma fbf atômica da forma $\mathcal{P}t_1 \ldots t_n$ e Ω_i é o objeto selecionado por t_i, então

$$s(\mathcal{A}, a) = \begin{cases} 1 & \text{se } \langle \Omega_1 \ldots \Omega_n \rangle \text{ is na extensão } (\mathcal{P}) \text{ no } \mathbb{M}, \\ 0 & \text{caso contrário.} \end{cases}$$

   Para cada termo t_i: Se t_i é uma constante, então $\Omega_i = \text{referente}\,(t_i)$.
   Se t_i é uma variável, então $\Omega_i = a\,(t_i)$.

2. Se $\mathcal{A}$ é $\neg\mathcal{B}$ para alguma fbf $\mathcal{B}$, então

$$s(\mathcal{A}, a) = \begin{cases} 1 & \text{se } s(\mathcal{B}, a) = 0, \\ 0 & \text{caso contrário.} \end{cases}$$

3. Se $\mathcal{A}$ é $(\mathcal{B}\&\mathcal{C})$ para algumas fbfs $\mathcal{B}$, $\mathcal{C}$, então

$$s(\mathcal{A}, a) = \begin{cases} 1 & \text{se } s(\mathcal{B}, a) = 1 \text{ e } s(\mathcal{C}, a) = 1, \\ 0 & \text{caso contrário.} \end{cases}$$

4. Se $\mathcal{A}$ é $(\mathcal{B} \vee \mathcal{C})$ para algumas fbfs $\mathcal{B}$, $\mathcal{C}$, então

$$s(\mathcal{A}, a) = \begin{cases} 0 & \text{se } s(\mathcal{B}, a) = 0 \text{ e } s(\mathcal{C}, a) = 0, \\ 1 & \text{caso contrário.} \end{cases}$$

5. Se $\mathcal{A}$ é $(\mathcal{B} \to \mathcal{C})$ para algumas fbfs $\mathcal{B}$, $\mathcal{C}$, então

$$s(\mathcal{A}, a) = \begin{cases} 0 & \text{se } s(\mathcal{B}, a) = 1 \text{ e } s(\mathcal{C}, a) = 0, \\ 1 & \text{caso contrário.} \end{cases}$$

6. Se $\mathcal{A}$ é $(\mathcal{B} \leftrightarrow \mathcal{C})$ para algumas fbfs $\mathcal{B}$, $\mathcal{C}$, então

$$s(\mathcal{A}, a) = \begin{cases} 1 & \text{se } s(\mathcal{B}, a) = s(\mathcal{C}, a), \\ 0 & \text{caso contrário.} \end{cases}$$

7. Se $\mathcal{A}$ é $\forall \chi \mathcal{B}$ para alguma fbf $\mathcal{B}$ e alguma variável χ, então

$$s(\mathcal{A}, a) = \begin{cases} 1 & \text{se } s(\mathcal{B}, a[\Omega|\chi]) = 1 \text{ para todo membro } \Omega \text{ do UD,} \\ 0 & \text{caso contrário.} \end{cases}$$

8. Se $\mathcal{A}$ é $\exists \chi \mathcal{B}$ para alguma fbf $\mathcal{B}$ e alguma variável χ, então

$$s(\mathcal{A}, a) = \begin{cases} 1 & \text{se } s(\mathcal{B}, a[\Omega|\chi]) = 1 \text{ para no mínimo um membro } \Omega \text{ do UD,} \\ 0 & \text{caso contrário.} \end{cases}$$

Essa definição segue a mesma estrutura que a definição de fbf para LQ, assim sabemos que toda fbf de LQ será compreendida por essa definição. Para um modelo $\mathbb{M}$ e uma atribuição de variável a, qualquer fbf será satisfeita ou não. Nenhuma fbfs são excluídas dela ou têm valores conflitantes atribuídos.

Verdade

Considere uma sentença simples como $\forall x Px$. Pela linha 7 na definição de satisfação, essa sentença é satisfeita se $a\,[\Omega|x]$ satisfaz Px em $\mathbb{M}$ para todo Ω no UD. Pela linha 1 da definição, esse será o caso se todo Ω está na extensão de P. Se $\forall x Px$ é satisfeita, não depende da atribuição de uma variável específica para a. Se essa sentença é satisfeita, então é verdadeira. Essa é uma formalização do que vínhamos dizendo: $\forall x Px$ é verdadeira se tudo no UD está na extensão de P.

A mesma coisa se dá em qualquer sentença de LQ. Porque todas as variáveis estão ligadas, uma sentença é satisfeita ou não independente dos detalhes da atribuição de variável. Desse modo, podemos definir verdade do seguinte modo: uma sentença $\mathscr{A}$ é VERDADEIRA EM $\mathbb{M}$ se e somente se alguma atribuição de variável satisfaz $\mathscr{A}$ em $\mathbb{M}$; do contrário,$\mathscr{A}$ é FALSA EM $\mathbb{M}$.

Verdade em LQ é *verdade em um modelo*. Sentenças de LQ não são absolutamente verdadeiras ou falsas enquanto meros símbolos, elas o são relativamente a um modelo. Um modelo fornece o significado dos símbolos, na medida em que isso afeta a verdade ou a falsidade.

Raciocinando sobre todos os modelos (reprise)

Ao final da seção 5.4, falhamos ao tentar mostrar que $\forall x\,(Rxx \to Rxx)$ é uma tautologia. Uma vez que definimos satisfação, podemos raciocinar do seguinte modo:

> Considere algum modelo arbitrário $\mathbb{M}$. Agora considere um membro arbitrário do UD; por conveniência, o denominaremos Ω. Ele deve ser o caso se $\langle\Omega, \Omega\rangle$ está na extensão de R ou se não estiver. Se $\langle\Omega, \Omega\rangle$ está na extensão de R, então Rxx é satisfeita por uma atribuição de variável que atribui Ω para x (pela linha da definição de satisfação), já que a consequente de $Rxx \to Rxx$ é satisfeito, o condicional é satisfeito (pela linha 5). Se $\langle\Omega, \Omega\rangle$ não está na extensão de R, então Rxx não é satisfeita pela atribuição de variável que atribui Ω para x (pela linha 1); já que a antecedente $Rxx \to Rxx$ não é satisfeita, a condicional é satisfeita (pela linha 5). Em qualquer dos casos, $Rxx \to Rxx$ é satisfeita. Isso é verdadeira para qualquer membro do UD, então $\forall x\,(Rxx \to Rxx)$ é satisfeita para qualquer atribuição de valor de verdade (pela linha 7). Portanto, $\forall x\,(Rxx \to Rxx)$ é verdadeira em $\mathbb{M}$ (pela definição de verdade). Esse argumento será o caso independentemente do UD específico e independentemente da extensão específica de R, então $\forall x\,(Rxx \to Rxx)$ é verdadeira em qualquer modelo. Portanto, a sentença é uma tautologia.

Fornecer argumentos sobre todos os modelos possíveis geralmente requer combinações astutas de duas estratégias:

1. Dividir os casos entre dois tipos possíveis, de modo que todo caso seja de um tipo ou do outro. Por exemplo, no argumento da página 119, distinguimos dois tipos de modelos em função da presença ou não de um par

ordenado específico na extensão (R). No argumento acima, distinguimos dois casos nos quais um par ordenado estava na extensão (R) e os nos quais não estava.

2. Considere um objeto arbitrário como uma maneira de mostrar algo mais geral. No argumento acima, foi crucial que Ω fosse apenas algum membro arbitrário do UD. Não supomos coisa alguma de especial. Assim, seja lá o que mostrássemos ser o caso quanto a Ω, este também deveria ser o caso de todo membro do UD — se pudéssemos mostrá-lo para Ω, poderíamos mostrar para qualquer coisa. Da mesma maneira, não supomos coisa alguma de especial para $\mathbb{M}$ e, seja lá o que mostrássemos de $\mathbb{M}$, deveria ser o caso para todos os modelos.

Considere mais um exemplo. O argumento $\forall x\,(Hx \& Jx) \therefore \forall x Hx$ é obviamente válido. Podemos apenas mostrar que esse argumento é válido considerando o que deve ser verdadeiro em todo modelo no qual a premissa é verdadeira.

> Considere algum modelo arbitrário $\mathbb{M}$ no qual a premissa $\forall x\,(Hx \& Jx)$ é verdadeira. A conjunção $Hx \& Jx$ é satisfeita independentemente do que é atribuído a x, então Hx também deve ser (pela linha 3 da definição de satisfação). Assim, $\forall x Hx$ é satisfeita por qualquer atribuição de variável (pela linha 7 da definição de satisfação). Já que não supomos coisa alguma de especial sobre $\mathbb{M}$, além de $\forall x\,(Hx \& Jx)$ ser verdadeira, $\forall x Hx$ deve ser verdadeira em qualquer modelo no qual $\forall x\,(Hx \& Jx)$ é verdadeira. Portanto, $\forall x\,(Hx \& Jx) \vDash \forall x Hx$.

Mesmo para um argumento simples como esse, até certo ponto o raciocínio é complicado. Para argumentos mais longos, o raciocínio pode ser impraticável. O problema surge porque falar sobre uma infinidade de modelos requer raciocinar sobre coisas em português. O que devemos fazer?

Podemos tentar formalizar nosso raciocínio sobre modelos, codificando as estratégias de "dividir para conquistar" que utilizamos acima. Essa abordagem, originalmente chamada de *tableau semântico*, foi desenvolvida por volta de 1950 por Everth Beth e Jaakko Hintikka. O tableau deles é agora mais comumente chamado de *árvores semânticas.*

Uma abordagem mais tradicional é considerar argumentos dedutivos como provas. Um *sistema de prova* consiste em regras que distinguem formalmente entre argumentos legítimos e ilegítimos — sem se considerar modelos ou significados dos símbolos. No próximo capítulo, desenvolveremos um sistema de prova para LS e LQ.

5.6 Exercícios práticos

$\star$ **Parte A** Determine se cada sentença é verdadeira ou falsa no modelo dado.

UD ={Corwin, Benedict}

extensão(A) = {Corwin, Benedict}

extensão(B) = {Benedict}

extensão(N) = $\emptyset$

referente(c) = Corwin

1. Bc
2. $Ac \leftrightarrow \neg Nc$
3. $Nc \leftrightarrow (Ac \vee Bc)$
4. $\forall x Ax$
5. $\forall x \neg Bx$
6. $\exists x(Ax \& Bx)$
7. $\exists x(Ax \rightarrow Nx)$
8. $\forall x(Nx \vee \neg Nx)$
9. $\exists x Bx \rightarrow \forall x Ax)$

$\star$ **Parte B** Determine se cada sentença é verdadeira ou falsa no modelo dado.

UD =.{Waylan, Willy, Johnny}

extensão(H) = {Waylan, Willy, Johnny}

extensão(W) = {Waylan, Willy}

extensão(R) = {<Waylan, Willy>, <Willy, Johnny>, <Johnny, Waylan>}

referente(m) = Johnny

1. $\exists x(Rxm \& Rmx)$
2. $\forall x(Rxm \vee Rmx)$
3. $\forall x(Hx \leftrightarrow Wx)$
4. $\forall x(Rxm \rightarrow Wx)$
5. $\forall x[Wx \rightarrow (Hx \& Wx)]$
6. $\exists x Rxx$
7. $\exists x \exists y Rxy$
8. $\forall x \forall y Rxy$
9. $\forall x \forall y(Rxy \vee Ryx)$
10. $\forall x \forall y \forall z[(Rxy \& Ryz) \rightarrow Rxz]$

Parte C Determine se cada sentença é verdadeira ou falsa no modelo dado.

UD =.{Lemmy, Courtney, Eddy}

extensão(G) = {Lemmy, Courtney, Eddy}

extensão(H) = {Courtney}

extensão(M) = {Lemmy, Eddy}

referente(c) = Courtney

referente(e) = Eddy

1. Hc
2. He
3. $Mc \vee Me$
4. $Gc \vee \neg Gc$

5. $Mc \rightarrow Gc$
6. $\exists x Hx$
7. $\forall x Hx$
8. $\exists x \neg Mx$
9. $\exists x(Hx \& Gx)$
10. $\exists x(Mx \& Gx)$
11. $\forall x(Hx \vee Mx)$
12. $\exists x Hx \& \exists x Mx$
13. $\forall x(Hx \leftrightarrow \neg Mx)$
14. $\exists x Gx \& \exists x \neg Gx$
15. $\forall x \exists y(Gx \& Hy)$

⋆ **Parte D** Construa o modelo que corresponde à interpretação dada.

UD: números naturais de 10 a 13.

Ix: x é impar.

Sx: x é menor que 7.

Dx: x é um número de dois digitos.

Mx: x traz má sorte.

Pxy: x é o próximo número após y.

Parte E Mostre que cada uma das sentenças seguintes é contingente.

⋆1. $Da \& Db$

⋆2. $\exists x Txh$

⋆3. $Pm \& \neg \forall x Px$

1. $\forall z Jz \leftrightarrow \exists y Jy$
2. $\forall x(Wxmn \vee \exists y Lxy)$

3. $\exists x(Gx \rightarrow \forall y My)$

⋆ **Parte F** Mostre que cada par das sentenças seguintes não é logicamente equivalente.

1. Ja, Ka
2. $\exists x Jx, Jm$
3. $\forall x Rxx, \exists x Rxx$
4. $\exists x Px \rightarrow Qc, \exists x(Px \rightarrow Qc)$
5. $\forall x(Px \rightarrow \neg Qx), \exists x(Px \& \neg Qx)$
6. $\exists x(Px \& Qx), \exists x(Px \rightarrow Qx)$
7. $\forall x(Px \rightarrow Qx), \forall x(Px \& Qx)$
8. $\forall x \exists y Rxy, \forall x \exists y Ryx$

Parte G Mostre que os conjuntos das sentenças seguintes são consistentes.

1. $\{Ma, \neg Na, Pa, \neg Qa\}$
2. $\{Lee, Lef, \neg Lfe, \neg Lff\}$
3. $\{\neg(Ma \& \exists x Ax), Ma \vee Fa, \forall x(Fx \rightarrow Ax)\}$
4. $\{Ma \vee Mb, Ma \rightarrow \forall x \neg Mx\}$
5. $\{\forall y Gy, \forall x(Gx \rightarrow Hx), \exists y \neg Iy)\}$
6. $\{\exists x(Bx \vee Ax), \forall x \neg Cx, \forall x[(Ax \& Bx) \rightarrow Cx]\}$
7. $\{\exists x Xx, \exists x Yx, \forall x(Xx \leftrightarrow \neg Yx)\}$
8. $\{\forall x(Px \vee Qx), \exists x \neg(Qx \& Px)\}$
9. $\{\exists z(Nz \& Ozz), \forall x \forall y(Oxy \rightarrow Oyx)\}$
10. $\{\neg \exists x \forall y Rxy, \forall x \exists y Rxy\}$

Parte H Construa modelos que mostrem que os argumentos seguintes são inválidos.

1. $\forall x(Ax \to Bx) .\because . \exists xBx$
2. $\forall x(Rx \to Dx), \forall x(Rx \to Fx) .\because . \exists x(Dx\&Fx)$
3. $\exists x(Px \to Qx) .\because . \exists xPx$
4. $Na\&Nb\&Nc .\because . \forall xNx$
5. $Rde, \exists xRxd .\because . Red$
6. $\exists x(Ex\&Fx), \exists xFx \to \exists xGx .\because . \exists x(Ex\&Gx)$
7. $\forall xOxc, \forall xOcx .\because . \forall xOxx$
8. $\exists x(Jx\&Kx), \exists x\neg Kx, \exists x\neg Jx .\because . \exists x(\neg Jx\&\neg Kx)$
9. $Lab \to \forall xLxb, \exists xLxb .\because . Lbb$

Parte I

⋆1. Mostre que $\{\neg Raa, \forall x(x = a \vee Rxa)\}$ é consistente.

⋆2. Mostre que$\{\forall x\forall y\forall z(x = y \vee y = z \vee x = z), \exists x\exists y\, x \neq y\}$ é consistente.

⋆3. Mostre que $\{\forall x\forall y\, x = y, \exists x\, x \neq a\}$ é inconsistente.

1. Mostre que $\exists x(x = h\&x = i)$ é contingente.
2. Mostre que$\{\exists x\exists y(Zx\&Zy\&x = y), \neg Zd, d = s\}$ é consistente.
3. Mostre que $'\forall x(Dx \to \exists yTyx) .\because . \exists y\exists z\, y \neq z'$ é inválido.

Parte J

1. Muitos livros de lógica definem consistência e inconsistência da seguinte maneira: “Um conjunto $\{A_1, A_2, A_3, ...\}$é inconsistente se e somente se $\{A_1, A_2, A_3, ...\} \models (B\&\neg B)$ para alguma sentença B. Um conjunto é consistente se ele não é inconsistente.” Essa definição faz com que conjuntos sejam considerados consistentes de modo diferente daquela definição que demos na p. 106? Justifique sua resposta.

⋆2. Nossa definição de verdade diz que uma sentença $\mathcal{A}$ é VERDADEIRA EM $\mathbb{M}$ se e somente se alguma atribuição da variável satisfaz $\mathcal{A}$ em M. Em vez disso, faria diferença se disséssemos que $\mathcal{A}$ é VERDADEIRA EM $\mathbb{M}$ se e somente se *toda* atribuição de variável satisfizesse $\mathcal{A}$ em M? Justifique sua resposta.

Capítulo 6

Provas

Considere dois argumentos em LS:

Argumento A	Argumento B
$P \vee Q$	$P \rightarrow Q$
$\neg P$	P
$\therefore Q$	$\therefore Q$

É claro que esses argumentos são válidos. Você pode confirmar sua validade construindo tabelas de verdade de quatro linhas. O argumento A utiliza uma inferência lógica que é sempre válida: dada uma disjunção e a negação de um dos disjuntos, o outro disjunto segue como consequência válida. Essa regra é chamada de *silogismo disjuntivo*.

O argumento B utiliza uma forma válida diferente: dada uma condicional e sua antecedente, sua consequente segue como consequência válida. Essa regra é chamada de *modus ponens*.

Ao construirmos tabelas de verdade, não precisamos dar nomes a diferentes formas de inferência. Não há razão para distinguirmos um *modus ponens* de um silogismo disjuntivo. Entretanto, por essa mesma razão, o método da tabela de verdade não mostra o *porquê* um argumento é válido. Se você fizesse uma tabela de verdade de 1024 linhas para um argumento que possui dez letras sentenciais, então você poderia checar se há alguma linha na qual as premissas são todas verdadeiras e a conclusão é falsa. Se você não viu tal linha e contanto que não tenha cometido nenhum erro ao fazer a tabela, então você saberia que o argumento seria válido. Todavia, ainda assim você não saberia nada além do porquê esse argumento em particular é válido.

O objetivo de um *sistema de prova* é mostrar que argumentos particulares são válidos de um modo que nos permita compreender o raciocínio envolvido em tal argumento. Começamos com formas de argumentos básicos, como o silogismo disjuntivo e o *modus ponens*. Por sua vez, essas formas podem ser combinadas para fazer argumentos mais complicados, como esse:

1. $\neg L \to (J \vee L)$
2. $\neg L$

$\therefore J$

Através do *modus ponens,* 1 e 2 implicam $(J \vee L)$. Essa é uma *conclusão intermediária*, pois segue logicamente das premissas, mas não é a conclusão que buscamos. Ademais, $(J \vee L)$ e a linha (2) implicam J, por silogismo disjuntivo. Não precisamos de uma nova regra para esse argumento. Sua prova mostra que se trata apenas da combinação de regras que já introduzimos.

Formalmente, uma PROVA é uma sequência de sentenças. As primeiras sentenças da sequência são pressuposições, essas são as premissas do argumento. Toda sentença posterior na sequência segue das sentenças anteriores através de uma das regras de prova. A sentença final da sequência é a conclusão do argumento.

Esse capítulo começa com um sistema de prova para LS que, por sua vez, é estendido para compreender a LQ e a LQ com a relação de identidade.

6.1 Regras básicas para LS

Ao construirmos um sistema de prova, poderíamos simplesmente começar com o silogismo disjuntivo e o *modus ponens*. Todas as vezes que descobríssemos um argumento válido que não pudesse ser provado com as regras já dadas, poderíamos introduzir novas regras. Procedendo dessa maneira, teríamos uma porção de regras não sistemáticas. Poderíamos até adicionar algumas regras estranhas e certamente acabaríamos com mais regras do que precisamos.

Portanto, ao invés disso, desenvolveremos o que é chamado de um sistema de DEDUÇÃO NATURAL. Em um sistema de dedução natural, teremos duas regras para cada operador lógico: uma regra de INTRODUÇÃO que nos permite provar uma sentença que tem esse operador como seu operador lógico principal e uma regra de ELIMINAÇÃO que nos permite provar algo dada uma sentença que possui tal operador como seu operador lógico principal.

Além das regras para cada operador lógico, também teremos uma regra de reiteração. Se você já tiver demonstrado algo ao longo de uma prova, a regra de reiteração permite você repeti-lo em uma nova linha. Por exemplo:

1	$\mathcal{A}$	
2	$\mathcal{A}$	R 1

Quando adicionamos uma linha à prova, escrevemos a regra que justifica aquela linha. Também escrevemos os números das linhas aos quais a regra se aplica. A regra de reiteração acima é justificada pela linha 1, a linha que você está reiterando. Assim, o 'R1' na linha 2 da prova significa que a linha é justificada pela regra de reiteração (R) aplicada à linha 1.

Obviamente, a regra de reiteração não nos permite demonstrar algo *novo*. Para isso, precisaremos de mais regras. O restante dessa seção dará regras de introdução e eliminação para todos os conectivos sentenciais. Isso nos dará um sistema de prova completo para LS. Mais à frente nesse capítulo, introduziremos as regras para quantificadores e identidade.

Todas as regras desse capítulo estão resumidas na página 189.

Conjunção

Pense por um momento: O que você precisaria demonstrar para provar $E\&F$?

Você poderia demonstrar $E\&F$ provando E e separadamente provando F. Esse é o caso mesmo se os dois conjuntos não forem sentenças atômicas. Se você pode provar $[(A \vee J) \rightarrow V]$ e $[(V \rightarrow L) \leftrightarrow (F \vee N)]$, então você efetivamente provou $[(A \vee J) \rightarrow V]\&[(V \rightarrow L) \leftrightarrow (F \vee N)]$.

Essa será nossa regra de introdução de conjunção, que abreviaremos $\&I$:

$$\begin{array}{r|ll} m & \mathcal{A} & \\ n & \mathcal{B} & \\ & \mathcal{A}\,\&\,\mathcal{B} & \quad \&\,\mathrm{I}\ m,\ n \end{array}$$

Uma linha da prova deve ser justificada por alguma regra, aqui temos '$\&I\ m, n$.' Isso significa: introdução de conjunção aplicada às linhas m e n. Essas são variáveis e não os números das linhas reais; m é alguma linha e n alguma outra linha. Em uma prova definitiva, as linhas são números 1,2,3,..., e regras devem ser aplicadas a números específicos de linhas. Todavia, quando definimos a regra utilizamos variáveis para destacar o fato de que a regra pode ser aplicada para duas linhas quaisquer que já estejam na prova. Se você tem K na linha 8 e L na linha 15, você pode provar $(K\&L)$ em algum passo posterior da prova com a justificação '$\&I\ 8, 15$.'

Agora, considere a regra de eliminação para conjunção. O que é permitido concluir a partir de uma sentença como $E\&F$? Certamente, você pode concluir E; se $E\&F$ fosse verdadeira, então E também seria. Similarmente, você pode concluir F. Essa será nossa regra de eliminação da conjunção , que abreviaremos $\&E$:

$$\begin{array}{r|ll} m & \mathcal{A}\,\&\,\mathcal{B} & \\ & \mathcal{A} & \quad \&\,\mathrm{E}\ m \\ & \mathcal{B} & \quad \&\,\mathrm{E}\ m \end{array}$$

Quando você tem uma conjunção em alguma linha da prova, você pode utilizar $\&E$ para derivar qualquer um dos conjuntos. A regra $\&E$ necessita apenas de uma sentença, então escrevemos um número de linha como justificativa de sua aplicação.

Mesmo com apenas essas regras podemos realizar algumas provas. Considere este argumento.

$$[(A \vee B) \rightarrow (C \vee D)]\,\&\,[(E \vee F) \rightarrow (G \vee H)]$$

$$[\therefore]\ [(E \vee F) \rightarrow (G \vee H)]\,\&\,[(A \vee B) \rightarrow (C \vee D)]$$

O operador principal tanto na premissa quanto na conclusão é a conjunção. Já que a conjunção é simétrica, o argumento é obviamente válido. Para fazermos uma prova, começamos escrevendo a premissa. Depois das premissas, escrevemos uma linha horizontal — tudo abaixo dessa linha deve ser justificado por uma regra de prova. Então o início da prova deve parecer com o seguinte:

1	$[(A \lor B) \to (C \lor D)] \,\&\, [(E \lor F) \to (G \lor H)]$	

A partir dessa premissa, podemos concluir cada um dos conjuntos através de $\&E$. A prova agora parece com o seguinte:

1	$[(A \lor B) \to (C \lor D)] \,\&\, [(E \lor F) \to (G \lor H)]$	
2	$[(A \lor B) \to (C \lor D)]$	& E 1
3	$[(E \lor F) \to (G \lor H)]$	& E 1

A regra $\&I$ depende de termos os dois conjuntos disponíveis em algum passo da prova. Eles podem estar separados um do outro e podem aparecer em qualquer ordem. Assim, ao se aplicar a regra $\&I$ às linhas 3 e 2, derivamos a conclusão desejada. A prova completa se assemelha a isso:

1	$[(A \lor B) \to (C \lor D)] \,\&\, [(E \lor F) \to (G \lor H)]$	
2	$[(A \lor B) \to (C \lor D)]$	& E 1
3	$[(E \lor F) \to (G \lor H)]$	& E 1
4	$[(E \lor F) \to (G \lor H)] \,\&\, [(A \lor B) \to (C \lor D)]$	& I 3, 2

Essa prova é trivial, mas ela mostra que podemos utilizar as regras de prova conjuntamente para demonstrar a validade de uma forma de argumento. Não obstante: se utilizássemos uma tabela de verdade para mostrar a validade desse argumento, ela teria 256 linhas, devido às oito letras sentenciais do argumento.

Disjunção

Se M fosse verdadeira, então $M \lor N$ também seria. Assim, a regra de introdução de disjunção ($\lor$I) nos permite derivar uma disjunção se tivermos um dos dois disjuntos:

m	$\mathcal{A}$	
	$\mathcal{A} \lor \mathcal{B}$	$\lor$I m
	$\mathcal{B} \lor \mathcal{A}$	$\lor$I m

Note que $\mathcal{B}$ pode ser qualquer sentença. Então a prova seguinte será legítima:

$$
\begin{array}{r|ll}
1 & M & \\
\hline
2 & M \vee ([(A \leftrightarrow B) \rightarrow (C \,\&\, D)] \leftrightarrow [[E \,\&\, F]) & \vee\text{I } 1
\end{array}
$$

Pode parecer estranho que apenas por saber M possamos derivar uma conclusão que inclua sentenças como $\mathcal{A}$, $\mathcal{B}$ e o resto – sentenças que nada tem a ver com M. Ainda assim, a conclusão segue imediatamente de $\vee I$. E isso deve ser assim: as condições de verdade para a disjunção significam que, se $\mathcal{A}$ é verdadeira, então $\mathcal{A} \vee \mathcal{B}$ é verdadeira independentemente do que é $\mathcal{B}$. A conclusão não poderia ser falsa se a premissa fosse verdadeira ; desse modo, o argumento é válido.

Agora considere a regra de eliminação da disjunção. O que você pode concluir a partir de $M \vee N$? Você não pode concluir M. Pode ser o caso da verdade de M tornar $M \vee N$ também verdadeira, como no exemplo acima, mas esse também pode não ser o caso. A partir de $M \vee N$ apenas, não se pode concluir coisa alguma sobre M ou N especificamente. Entretanto, se você soubesse que N é falso, você poderia concluir M.

Trata-se tão só silogismo disjuntivo, que será a regra de eliminação da disjunção ($\vee E$).

$$
\begin{array}{r|ll}
m & \mathcal{A} \vee \mathcal{B} & \\
n & \neg\mathcal{B} & \\
 & \mathcal{A} & \vee\text{E } m,\, n
\end{array}
\qquad\qquad
\begin{array}{r|ll}
m & \mathcal{A} \vee \mathcal{B} & \\
n & \neg\mathcal{A} & \\
 & \mathcal{B} & \vee\text{E } m,\, n
\end{array}
$$

Condicional

Considere este argumento:

$$
\begin{array}{rl}
 & R \vee F \\
\therefore & \neg R \rightarrow F
\end{array}
$$

Esse argumento é certamente válido. Como deve ser a regra de introdução de condicional, de forma que possamos tirar essa conclusão?

Começamos a prova escrevendo a premissa do argumento e desenhando uma linha horizontal, como essa:

1	$R \vee F$	

Se tivéssemos $\neg R$ como uma outra premissa, poderíamos derivar F pela regra $\vee E$. Porém, não possuímos $\neg R$ como premissa desse argumento, nem podemos derivá-lo diretamente a partir da premissa que temos — então não podemos simplesmente provar F. Em vez disso, começaremos uma *subprova*, uma prova dentro da prova principal. Ao começarmos uma subprova, desenhamos outra linha vertical para indicar que não estamos mais na prova principal. Então, escrevendo dentro dessa subprova uma hipótese para a subprova. Que pode ser qualquer coisa que queiramos. Aqui, será útil supor $\neg R$. Nossa prova parece com o seguinte:

1	$R \vee F$	
2	$\quad \neg R$	

É importante notar que não estamos asserindo que provamos $\neg R$. Não precisamos escrever qualquer justificação para a linha de hipótese em uma subprova. Você pode pensar na subprova como se ela abrisse uma questão: o que poderíamos demonstrar se $\neg R$ fosse verdadeira? Poderíamos, então, derivar F. Assim, fazemos:

1	$R \vee F$	
2	$\quad \neg R$	
3	$\quad F$	$\vee$E 1, 2

Isso demonstra que *se* tivéssemos $\neg R$ como uma premissa, *então* poderíamos provar F. Na verdade, provamos $\neg R \to F$. Dessa maneira, a regra de introdução de condicional ($\to I$) nos permitirá fechar a subprova e derivar $\neg R \to F$ na prova principal. Nossa prova final parece com o seguinte:

1	$R \vee F$	
2	$\quad \neg R$	
3	$\quad F$	$\vee$E 1, 2
4	$\neg R \to F$	$\to$I 2–3

Note que a justificação por se aplicar a regra $\rightarrow I$ é a subprova inteira. Geralmente, essa justificação se estende por mais de duas linhas.

Pode até parecer que a habilidade de supor qualquer coisa em uma subprova levaria ao caos: mas será que isso de fato nos permite provar qualquer conclusão a partir de qualquer premissa? A resposta é não. Considere essa prova:

$$
\begin{array}{r|l l}
1 & \underline{\mathcal{A}} & \\
2 & \quad \vert\ \underline{\mathcal{B}} & \\
3 & \quad \vert\ \mathcal{B} & \text{R } 2
\end{array}
$$

Isso pode parecer uma prova que você pode derivar quaisquer conclusões $\mathcal{B}$ a partir de qualquer premissa $\mathcal{A}$. Quando a linha vertical para a subprova termina, a subprova está *fechada*. Para completar a subprova, você deve fechar todas as subprovas. Todavia, você não pode fechar a subprova e utilizar a regra R de novo na linha 4 para derivar $\mathcal{B}$ na prova principal. Uma vez que você fechou uma subprova, você não pode se referir de volta às linhas individuais dentro daquela.

Fechar uma subprova é chamado de *descarte* das hipóteses daquela subprova. Então podemos resumir essa questão da seguinte maneira: você não pode completar uma subprova até você ter descartado todas as hipóteses, com exceção das premissas originais do argumento.

Claro que é legítimo fazer o seguinte:

$$
\begin{array}{r|l l}
1 & \underline{\mathcal{A}} & \\
2 & \quad \vert\ \underline{\mathcal{B}} & \\
3 & \quad \vert\ \mathcal{B} & \text{R } 2 \\
4 & \mathcal{B} \rightarrow \mathcal{B} & \rightarrow\text{I } 2\text{–}3
\end{array}
$$

Entretanto, isso não parece tão estranho. Já que $\mathcal{B} \rightarrow \mathcal{B}$ é uma tautologia, nenhuma premissa particular deve ser previamente necessária para derivá-la. (Na verdade, como veremos, uma tautologia segue a partir de quaisquer premissas).

De uma forma geral, a regra $\rightarrow$I parece com o seguinte:

$$
\begin{array}{r|l l}
m & \quad \mathcal{A} & \text{busco } \mathcal{B} \\
n & \quad \mathcal{B} & \\
 & \mathcal{A} \to \mathcal{B} & \to\text{I } m\text{–}n
\end{array}
$$

Quando introduzimos uma subprova, geralmente escrevemos o que queremos derivar na coluna. Fazemos isso apenas para nos lembrar o porquê começamos a subprova, caso ela se estenda por cinco ou dez linhas. Não há regra 'busco ...'. Isso é apenas uma nota para nós mesmos e não uma parte formal da prova.

Apesar de sempre ser possível abrir uma subprova com qualquer hipótese que você queira, existe uma estratégia envolvida para se escolher a hipótese adequada. Começar uma subprova com uma hipótese arbitrária ou insensata seria apenas um desperdício de linhas da prova. Por exemplo, para você derivar uma condicional através de $\to$I, você deve supor o antecedente da condicional em uma subprova.

A regra $\to$I também depende de a consequente do condicional ser a última linha da subprova. Sempre é possível fechar uma subprova e descartar suas hipóteses, mas não será útil fazer isso até você ter derivado o que você buscava.

Agora considere a regra de eliminação do condicional. Nada segue a partir de $M \to N$ apenas, mas se tivermos ambos $M \to N$ e M, então podemos concluir N. Essa regra, *modus ponens*, será a regra de eliminação do condicional $\to$E.

$$
\begin{array}{r|l l}
m & \mathcal{A} \to \mathcal{B} & \\
n & \mathcal{A} & \\
 & \mathcal{B} & \to\text{E } m, n
\end{array}
$$

Agora que possuímos as regras para a condicional, considere o seguinte argumento:

$$
\begin{array}{l}
P \to Q \\
Q \to R \\
\therefore\ P \to R
\end{array}
$$

Começamos a prova escrevendo as duas premissas como hipóteses. Já que o operador principal na conclusão é um condicional, esperamos utilizar a regra $\to I$. Para isso, precisaremos de uma subprova — então escrevemos o antecedente do condicional como a hipótese da subprova:

$$
\begin{array}{r|l}
1 & P \to Q \\
2 & Q \to R \\
\hline
3 & \quad \begin{array}{|l} P \\ \hline \end{array}
\end{array}
$$

Tornamos P disponível ao supô-lo em nossa subprova, o que permite utilizar →E na primeira premissa. Isso nos dá Q, que nos permite utilizar →E na segunda premissa. Uma vez que derivamos R, fechamos a subprova. Portanto, ao supormos P conseguimos provar R, então aplicamos a regra →I e terminamos a prova.

$$
\begin{array}{r|lll}
1 & P \to Q & & \\
2 & Q \to R & & \\
\hline
3 & \quad \begin{array}{|l} P \\ \hline Q \\ R \end{array} & \quad & \begin{array}{l} \text{busco } R \\ \to\text{E } 1, 3 \\ \to\text{E } 2, 4 \end{array} \\
6 & P \to R & & \to\text{I } 3\text{–}5
\end{array}
$$

Bicondicional

As regras para o bicondicional serão versões duplicadas daquelas para o condicional.

Por exemplo, para derivar $W \leftrightarrow X$, você deve provar X ao supor W *e* provar W ao supor X. A regra de introdução de bicondicional necessita de duas subprovas. Essas subprovas podem vir em qualquer ordem e a segunda subprova não precisa vir imediatamente após a primeira — mas esquematicamente a regra é:

$$
\begin{array}{r|l l}
m & \quad \mathcal{A} & \text{busco } \mathcal{B} \\
n & \quad \mathcal{B} & \\
p & \quad \mathcal{B} & \text{busco } \mathcal{A} \\
q & \quad \mathcal{A} & \\
 & \mathcal{A} \leftrightarrow \mathcal{B} & \leftrightarrow\text{I } m\text{–}n,\ p\text{–}q
\end{array}
$$

A regra de eliminação do bicondicional ($\leftrightarrow$E) permite fazer mais coisas que a regra análoga do condicional. Se você tem a subsentença do lado esquerdo do bicondicional, você pode derivar a subsentença do lado direito. Se você tem a subsentença do lado direito do bicondicional, você pode derivar a subsentença do lado esquerdo. Essa é a regra:

$$
\begin{array}{r|l l}
m & \mathcal{A} \leftrightarrow \mathcal{B} & \\
n & \mathcal{A} & \\
 & \mathcal{B} & \leftrightarrow\text{E } m,\ n
\end{array}
\qquad\qquad
\begin{array}{r|l l}
m & \mathcal{A} \leftrightarrow \mathcal{B} & \\
n & \mathcal{B} & \\
 & \mathcal{A} & \leftrightarrow\text{E } m,\ n
\end{array}
$$

Negação

Considere este simples argumento matemático em português:

> Suponha que há algum número que seja o maior número natural. Chame-o A.
>
> Este número mais um é também um número natural.
>
> Obviamente, $A + 1 > A$.
>
> Então, há um número natural maior que A.
>
> Isso é impossível, já que se supõe A ser o maior número natural.
>
> $\therefore$ O maior número natural não existe.

Essa forma de argumento é tradicionalmente chamado de *reductio ad absurdum*, ou redução ao absurdo. Em uma *reductio*, supomos algo a fim de corroborarmos com nosso argumento — por exemplo, que o maior número natural existe. Então, mostramos que essa hipótese leva a duas sentenças contraditórias — por exemplo, que A seja e não seja o maior número natural. Dessa maneira, mostramos que a hipótese original deve ser falsa.

As regras básicas da negação permitem argumentos desse tipo. Se pressupormos algo e mostrarmos que isso nos leva a sentenças contraditórias, então provamos a negação da hipótese. Essa regra é a introdução de negação (¬I):

$$
\begin{array}{l|l|ll}
m & & \underline{\mathcal{A}} & \text{por redução} \\
n & & \mathcal{B} & \\
n+1 & & \neg\mathcal{B} & \\
n+2 & \neg\mathcal{A} & & \neg\text{I } m\text{–}n+1
\end{array}
$$

Para aplicar essa regra, as duas últimas linhas de uma subprova devem ser uma contradição explícita: alguma sentença deve ser seguida na próxima linha pela sua negação. Escrevemos 'por redução' como uma nota para nós mesmos, um lembrete do porquê começamos aquela subprova. Isso não é uma parte formal da subprova e você pode não fazer se achar assim conveniente.

Para entender como a regra funciona, suponha que queiramos provar a lei de não-contradição: $\neg(G \& \neg G)$. Podemos provar isso sem quaisquer premissas, começando imediatamente com uma subprova. Queremos aplicar ¬I à subprova, então assumimos $(G \& \neg G)$. Então derivamos uma contradição explícita por &E. A prova se assemelha a isso:

$$
\begin{array}{l|l|ll}
1 & & \underline{G \,\&\, \neg G} & \text{por redução} \\
2 & & G & \&\text{E } 1 \\
3 & & \neg G & \&\text{E } 1 \\
4 & \neg(G \,\&\, \neg G) & & \neg\text{I } 1\text{–}3
\end{array}
$$

A regra ¬E funcionará analogamente. Se supormos $\neg\mathcal{A}$ e mostrarmos que leva a uma contradição, nós provaremos $\mathcal{A}$. A regra é assim:

$$
\begin{array}{l|l|ll}
m & & \underline{\neg\mathcal{A}} & \text{por redução} \\
n & & \mathcal{B} & \\
n+1 & & \neg\mathcal{B} & \\
n+2 & \mathcal{A} & & \neg\text{E } m\text{–}n+1
\end{array}
$$

6.2 Regras derivadas

As regras do sistema de dedução natural devem ser sistemáticas. Há uma regra de introdução e eliminação para cada operador lógico, mas por que usamos essas regras básicas e não outras? Muitos sistemas de dedução natural têm uma regra de eliminação da disjunção que funciona assim:

m	$\mathcal{A} \vee \mathcal{B}$	
n	$\mathcal{A} \to \mathcal{C}$	
o	$\mathcal{B} \to \mathcal{C}$	
	$\mathcal{C}$	DIL m, n, o

Chamamos essa regra Dilema (DIL). Pode parecer que há algumas provas que não podemos demonstrar com nosso sistema de provas, porque não temos essa regra. Mas esse não é o caso. Qualquer prova que você pode fazer utilizando a regra do Dilema pode ser feita com as regras básicas do nosso sistema de dedução natural. Considere essa prova:

1	$\mathcal{A} \vee \mathcal{B}$	
2	$\mathcal{A} \to \mathcal{C}$	
3	$\mathcal{B} \to \mathcal{C}$	busco $\mathcal{C}$
4	$\mid\ \neg\mathcal{C}$	por redução
5	$\mid\ \mid\ \mathcal{A}$	por redução
6	$\mid\ \mid\ \mathcal{C}$	$\to$E 2, 5
7	$\mid\ \mid\ \neg\mathcal{C}$	R 4
8	$\mid\ \neg\mathcal{A}$	$\neg$I 5–7
9	$\mid\ \mid\ \mathcal{B}$	por redução
10	$\mid\ \mid\ \mathcal{C}$	$\to$E 3, 9
11	$\mid\ \mid\ \neg\mathcal{C}$	R 4
12	$\mid\ \mathcal{B}$	$\vee$E 1, 8
13	$\mid\ \neg\mathcal{B}$	$\neg$I 9–11
14	$\mathcal{C}$	$\neg$E 4–13

$\mathcal{A}$, $\mathcal{B}$, e $\mathcal{C}$ são metavariáveis. Eles não são símbolos da LS, mas representam quaisquer sentenças arbitrárias da LS. Então não é, estritamente falando, uma prova em LS; e sim uma receita. Ela nos fornece um padrão que pode provar qualquer coisa que a regra Dilema pode provar, utilizando apenas as regras básicas da LS. Isso significa que a regra Dilema não é necessária. Adicioná-la à lista de regras básicas não nos permitiria derivar nada além do que já derivaríamos sem ela.

Não obstante, a regra Dilema é conveniente. Ela nos permite fazer em uma linha o que requer onze linhas e diversas subprovas utilizando as regras básicas. Portanto, a adicionaremos ao nosso sistema de prova como uma regra derivada.

Uma REGRADA DERIVADA é uma regra de prova que não torna nenhum tipo de prova nova possível. Qualquer coisa que pode ser provada com uma regra derivada pode ser provada sem ela. Você pode pensar em uma prova simplificada que utiliza uma regra derivada como um atalho de uma prova maior que utiliza apenas as regras básicas. Toda vez que você utilizar a regra Dilema, você pode utilizar dez linhas extras e provar a mesma coisa sem usá-la.

Por conveniência, adicionaremos várias outras regras derivadas. Uma delas é o *modus tollens* (MT).

m	$\mathcal{A} \rightarrow \mathcal{B}$	
n	$\neg\mathcal{B}$	
	$\neg\mathcal{A}$	MT m, n

Deixaremos a prova dessa regra como um exercício. Note que se já tivéssemos provado a regra MT, então a prova da regra DIL poderia ser feita em apenas cinco linhas.

Também adicionaremos o silogismo hipotético (SH) como uma regra derivada. Já havíamos dado uma prova dele na p. 140.

m	$\mathcal{A} \rightarrow \mathcal{B}$	
n	$\mathcal{B} \rightarrow \mathcal{C}$	
	$\mathcal{A} \rightarrow \mathcal{C}$	SH m, n

6.3 Regras de substituição

Considere como você provaria este argumento: $F \to (G \& H)\,, \therefore F \to G$.

Pode ser tentador escrever a premissa e aplicar a regra &E à conjunção $(G \& H)$. Entretanto, isso não é aceitável, porque as regras básicas de prova só podem ser aplicadas a sentenças completas. Precisamos ter $(G \& H)$ em uma linha isolada. Podemos provar esse argumento dessa maneira:

1	$F \to (G \,\&\, H)$	
2	$\quad F$	busco G
3	$\quad G \,\&\, H$	$\to$E 1, 2
4	$\quad G$	&E 3
5	$F \to G$	$\to$I 2–4

Introduziremos agora algumas regras derivadas que podem ser aplicadas a partes de uma sentença. Essas são chamadas de REGRAS DE SUBSTITUIÇÃO, porque elas podem ser utilizadas para substituir parte de uma sentença com uma expressão logicamente equivalente. Uma regra simples de substituição é a comutatividade (Com), que assere que podemos inverter a ordem dos conjuntos em uma conjunção ou a ordem dos disjuntos em uma disjunção. Podemos definir a regra da seguinte maneira:

$$
\begin{aligned}
(\mathcal{A} \,\&\, \mathcal{B}) &\iff (\mathcal{B} \,\&\, \mathcal{A}) \\
(\mathcal{A} \vee \mathcal{B}) &\iff (\mathcal{B} \vee \mathcal{A}) \\
(\mathcal{A} \leftrightarrow \mathcal{B}) &\iff (\mathcal{B} \leftrightarrow \mathcal{A}) \quad \text{Com}
\end{aligned}
$$

O símbolo de equivalência significa que você pode tomar uma subfórmula de um lado da seta e substituí-la pela subfórmula do outro lado. A seta é dupla porque as regras de substituição funcionam em ambas as direções.

Considere este argumento: $(M \vee P) \to (P \& M) \therefore (P \vee M) \to (M \& P)$.

É possível provar isso utilizando apenas as regras básicas, mas ela será longa e inconveniente. Com a regra Com, podemos facilmente fazer uma prova:

1	$(M \vee P) \rightarrow (P \,\&\, M)$	
2	$(P \vee M) \rightarrow (P \,\&\, M)$	Com 1
3	$(P \vee M) \rightarrow (M \,\&\, P)$	Com 2

Outra regra de substituição é a dulpa negação (DN). Com a regra DN, você pode remover ou inserir um par de negações em qualquer lugar de uma sentença. Esta é a regra:

$$\neg\neg\mathcal{A} \Longleftrightarrow \mathcal{A} \quad \text{DN}$$

Outras duas regras de substituição são chamadas de leis de De Morgan e que levam o nome do lógico britânico do século XIX, Augustus De Morgan. (Apesar de De Morgan ter descoberto essas leis, ele não foi o primeiro). Essas regras capturam relações úteis entre negação, conjunção e disjunção. Aqui estão as regras, que abreviamos DeM:

$$\begin{aligned} \neg(\mathcal{A} \vee \mathcal{B}) &\Longleftrightarrow (\neg\mathcal{A} \,\&\, \neg\mathcal{B}) \\ \neg(\mathcal{A} \,\&\, \mathcal{B}) &\Longleftrightarrow (\neg\mathcal{A} \vee \neg\mathcal{B}) \quad \text{DeM} \end{aligned}$$

Já que $\mathcal{A} \rightarrow \mathcal{B}$ é uma *condicional material*, ela é equivalente a $\neg\mathcal{A} \vee \mathcal{B}$. Uma regra de substituição que captura essa equivalência é chamada de CM (condicional material). Ela assume duas formas:

$$\begin{aligned} (\mathcal{A} \rightarrow \mathcal{B}) &\Longleftrightarrow (\neg\mathcal{A} \vee \mathcal{B}) \\ (\mathcal{A} \vee \mathcal{B}) &\Longleftrightarrow (\neg\mathcal{A} \rightarrow \mathcal{B}) \quad \text{CM} \end{aligned}$$

Agora considere esse argumento: $\neg\,(P \rightarrow Q) \mathrel{.\,\cdot\,.} (P\&\neg Q)$.

Como sempre, podemos provar esse argumento utilizando apenas nossas regras básicas. Todavia, com as regras de substituição a prova é muito mais simples:

1	$\neg(P \rightarrow Q)$	
2	$\neg(\neg P \vee Q)$	CM 1
3	$\neg\neg P \,\&\, \neg Q$	DeM 2
4	$P \,\&\, \neg Q$	DN 3

Uma última regra de substituição captura a relação entre condicionais e bicondicionais. Chamamos essa regra de troca de bicondicional e a abreviaremos por $\leftrightarrow$c.

$$[(\mathcal{A} \to \mathcal{B}) \,\&\, (\mathcal{B} \to \mathcal{A})] \Longleftrightarrow (\mathcal{A} \leftrightarrow \mathcal{B}) \quad \leftrightarrow\text{c}$$

6.4 Regras para quantificadores

Para provas em LQ, utilizamos todas as regras básicas da LS e mais quatro outras regras básicas: as regras de introdução e eliminação para cada quantificador.

Já que todas as regras derivadas em LS são derivadas das regras básicas, elas também serão válidas em LQ. Adicionaremos ainda outra regra derivada chamada negação de quantificador.

Instâncias de substituição

Para podermos formular resumidamente as regras para os quantificadores, precisamos de uma maneira de marcar a relação entre sentenças quantificadas e suas instâncias. Por exemplo, a sentença Pa é uma instância particular da asserção geral $\forall x Px$.

Para uma fbf $\mathcal{A}$, uma constante $\mathcal{c}$ e uma variável χ, definimos $\mathcal{A}\,\boxed{\chi \Rightarrow \mathcal{c}}$ como significando a fbf que obtemos ao substituirmos todas as ocorrências de χ em $\mathcal{A}$ por $\mathcal{c}$.

$\mathcal{A}\,\boxed{\chi \Rightarrow \mathcal{c}}$ é chamada UMA INSTÂNCIA DA SUBSTITUIÇÃO de $\forall x\mathcal{A}$ e $\exists x\mathcal{A}$. Por sua vez, $\mathcal{c}$ é chamada CONSTANTE DE INSTANCIAÇÃO.

- ▷ $Aa \to Ba$, $Af \to Bf$ e $Ak \to Bk$ são todas instâncias da substituição de $\forall x\,(Ax \to Bx)$; as instâncias de substituição são respectivamente a, f, k.
- ▷ Raj, Rdj e Rjj são todas instâncias da substituição de $\exists z Rzj$; as instâncias de substituição são respectivamente a, d, j.

Eliminação universal

Se você tem $\forall x Ax$, é legítimo inferir que qualquer coisa é um A. Você pode inferir Aa, Ab, Az, Ad_3. Isto é, você pode inferir qualquer instância de substituição – em suma, você pode inferir Ac para qualquer constante c. Essa é a forma geral da regra de eliminação universal ($\forall$E):

$$\begin{array}{r|ll} m & \forall \mathcal{X}\mathcal{A}\mathcal{X} & \\ & \mathcal{A}\boxed{\mathcal{X}\Rightarrow\mathcal{C}} & \forall\text{E } m \end{array}$$

Lembre-se que a caixa da instância de substituição não é um símbolo de LQ, então você não pode escrevê-la diretamente na prova. Em vez disso, escreva a sentença substituída com a consoante c substituindo todas as ocorrências da variável x em $\mathcal{A}$. Por exemplo:

$$\begin{array}{r|ll} 1 & \forall x(Mx \to Rxd) & \\ \hline 2 & Ma \to Rad & \forall\text{E } 1 \\ 3 & Md \to Rdd & \forall\text{E } 1 \end{array}$$

Introdução existencial

Quando é legítimo inferir $\exists x Ax$? Quando você souber que algo é um A — por exemplo, quando você tiver Aa disponível na prova.

Essa é a regra de introdução do existência ($\exists$I):

$$\begin{array}{r|ll} m & \mathcal{A}\mathcal{C} & \\ & \exists \mathcal{X}\mathcal{A}\boxed{\boxed{\mathcal{X}\Rightarrow\mathcal{C}}} & \exists\text{I } m \end{array}$$

É importante notar que $\mathcal{A}\boxed{\boxed{\mathcal{X}\Rightarrow\mathcal{C}}}$ não é necessariamente uma instância de substituição. Escrevemos com 'caixas duplas' para mostrar que a variável $\mathcal{X}$ não precisa ser substituída em todas as ocorrências da constante $\mathcal{C}$. Você pode decidir quais ocorrências devem ser substituídas e quais devem ser ignoradas.

Por exemplo:

1	$Ma \to Rad$	
2	$\exists x(Ma \to Rax)$	$\exists$I 1
3	$\exists x(Mx \to Rxd)$	$\exists$I 1
4	$\exists x(Mx \to Rad)$	$\exists$I 1
5	$\exists y \exists x(Mx \to Ryd)$	$\exists$I 4
6	$\exists z \exists y \exists x(Mx \to Ryz)$	$\exists$I 5

Introdução universal

Uma asserção universal como $\forall x Px$ seria provada se toda substituição de suas instâncias fossem provadas, isto é, se todas sentenças $Pa, Pb, \ldots$ estivessem disponíveis na prova. Infelizmente, não há como provar *toda* instância de substituição. Para isso seria necessário provar $Pa, Pb, \ldots, P_{j_2}, \ldots, P_{s_7}, \ldots$ e assim ao infinito. Existem infinitas constantes em LQ, portanto, esse processo nunca chegaria a termo.

Considere um argumento simples: $\forall x Mx \therefore \forall y My$.

Ele em nada modifica o significado da sentença, independentemente se utilizamos as variáveis x ou y, portanto, esse argumento é obviamente válido. Suponha que comecemos dessa maneira:

1	$\forall x Mx$	busco $\forall y My$
2	Ma	$\forall$E 1

Derivamos Ma. Nada nos impede de utilizar a mesma justificação para derivar $Mb, \ldots, M_{j_2}, \ldots, M_{s_7}, \ldots$ e assim por diante até que nosso espaço ou paciência tenha se esgotado. Nós mostramos uma maneira de provar $M\mathcal{c}$ para qualquer constante $\mathcal{c}$. $\forall y My$ segue o seguinte.

1	$\forall x Mx$	
2	Ma	$\forall$E 1
3	$\forall y My$	$\forall$I 2

É importante que a seja tão somente uma constante arbitrária. Não precisamos fazer qualquer hipótese especial sobre ela. Se Ma fosse uma premissa do argumento, então isso nada mostraria sobre *todo* y. Por exemplo:

1	$\forall x Rxa$	
2	Raa	$\forall$E 1
3	$\forall y Ryy$	não permitido!

Essa é a forma esquemática da regra de introdução universal ($\forall$I) :

m	$\mathcal{A}$	
	$\forall \chi \mathcal{A}$ $\boxed{c* \Rightarrow \chi}$	$\forall$I m

*A constante c não deve ocorrer em qualquer hipótese não descartada.

Note que podemos fazer isso para qualquer constante que não ocorre em uma hipótese não descartada e para qualquer variável.

Note também que a constante não pode ocorrer em qualquer hipótese *não descartada*, mas pode ocorrer nas hipóteses de uma subprova que já fechamos. Por exemplo, podemos provar $\forall z\,(Dz \to Dz)$ sem quaisquer premissas.

1	$\quad Df$	busco Df
2	$\quad Df$	R 1
3	$Df \to Df$	$\to$I 1–2
4	$\forall z (Dz \to Dz)$	$\forall$I 3

Eliminação existencial

Uma sentença com um quantificador existencial nos diz que há *algum* membro do UD que satisfaz uma fórmula. Por exemplo, $\exists x Sx$ nos diz que há pelo menos um S. Contudo, ela não nos diz *qual* membro do UD satisfaz S. Não podemos concluir imediatamente $Sa, S_{f_{23}}$, ou qualquer outra instância de substituição da sentença. O que podemos fazer?

Suponha que conhecemos ambos $\exists x Sx$ e $\forall x\,(Sx \to Tx)$. Poderíamos raciocinar da seguinte maneira:

A partir de $\exists x Sx$, sabemos que existe algo que é um S. Não sabemos

> quais constantes se referem a essa coisa. Se alguma o faz, nós a denominaremos ‘Ishmael’. A partir de $\forall x(Sx \rightarrow Tx)$, segue-se que se Ishmael é um S, então ele é um T. Portanto, Ishmael é um T. Porque sabemos que Ishmael é um T, sabemos também que $\exists xTx$.

Nesse parágrafo, introduzimos um nome para o que é um S. Conferimos a isso um nome arbitrário (‘Ishmael’) de forma que pudéssemos raciocinar sobre, então, derivamos algumas consequências sobre ele ser um S. Já que ‘Ishmael’ é apenas um nome fictício introduzido com fins à prova e não uma constante genuína, não podíamos mencioná-lo na conclusão. Ainda assim, conseguimos derivar uma sentença que não menciona Ishmael; isto é, $\exists xTx$. Essa sentença segue das premissas.

Queremos que a regra de eliminação existencial funcione de maneira similar. Todavia, já que nomes em português não são símbolos de LQ, não podemos utilizá-los em provas formais. Em vez disso, utilizaremos constantes de LQ que não apareceram previamente na prova.

Uma constante que é utilizada para representar seja lá o que for que satisfaça uma asserção existencial é chamada de INDICADORA. O raciocínio com uma indicadora deve ocorrer dentro de uma subprova e a indicadora não pode ser uma constante que já tenha sido utilizada anteriormente na prova.

Essa é a forma esquemática da regra de eliminação existencial ($\exists$E):

m	$\exists \chi \mathcal{A}\chi$	
n	$\quad \mathcal{A}\ \boxed{c* \Rightarrow \chi}$	
p	$\quad \mathcal{B}$	
	$\mathcal{B}$	$\exists$E m, n–p

*A constante c não deve ocorrer fora da subprova.

Lembre-se que a constante indicadora não pode ocorrer em $\mathcal{B}$, a sentença que você prova utilizando $\exists$E.

Seria suficiente exigir que a constante indicadora não ocorresse em $\exists\mathcal{A}\chi$, $\mathcal{B}$ ou qualquer outra hipótese não descartada. Contudo, tendo em vista que essa constante indicadora é tão somente um parâmetro utilizado dentro de uma subprova, exigimos uma constante inteiramente nova que não ocorra em qualquer outro passo na prova.

Podemos dar uma prova formal de que $\exists x Sx$ e $\forall x(Sx \to Tx)$ conjuntamente implicam $\exists x Tx$.

1	$\exists x Sx$	
2	$\forall x(Sx \to Tx)$	busco $\exists x Tx$
3	$\quad Si$	
4	$\quad Si \to Ti$	$\forall$E 2
5	$\quad Ti$	$\to$E 3, 4
6	$\quad \exists x Tx$	$\exists$I 5
7	$\exists x Tx$	$\exists$E 1, 3–6

Note que essa prova tem efetivamente a mesma estrutura que o argumento em português com o qual começamos, exceto que a subprova utiliza a constante indicadora 'a' em vez do nome fictício 'Ishmael'.

Negação de quantificador

Ao traduzirmos uma sentença do português para a LQ, notamos que $\neg\exists x \neg\mathcal{A}$ é logicamente equivalente a $\forall x \mathcal{A}$. Em LQ, eles são demonstrativamente equivalentes. Podemos demonstrar a metade da equivalência com uma prova um tanto feia:

1	$\forall x Ax$	busco $\neg\exists x \neg Ax$
2	$\quad \exists x \neg Ax$	por redução
3	$\quad\quad \neg Ac$	por $\exists$E
4	$\quad\quad\quad \forall x Ax$	por redução
5	$\quad\quad\quad Ac$	$\forall$E 1
6	$\quad\quad\quad \neg Ac$	R 3
7	$\quad\quad \neg\forall x Ax$	$\neg$I 4–6
8	$\quad \forall x Ax$	R 1
9	$\quad \neg\forall x Ax$	$\exists$E 2, 3–7
10	$\neg\exists x \neg Ax$	$\neg$I 2–9

Para provarmos que as duas sentenças são genuinamente equivalentes, precisaríamos de uma segunda prova que supusesse $\neg\exists\neg\mathcal{A}$ e derivasse $\forall x \mathcal{A}$. Deixamos essa prova como exercício para o leitor.

A tradução entre quantificadores, por adicionar ou subtrair negações dessa maneira, será bastante útil. Então, adicionaremos duas regras derivadas para essa finalidade. Essas regras são chamadas de negação do quantificador (NQ):

$$\neg\forall\chi\mathcal{A} \iff \exists\chi\neg\mathcal{A}$$
$$\neg\exists\chi\mathcal{A} \iff \forall\chi\neg\mathcal{A} \quad \text{NQ}$$

Já que a NQ é uma regra de substituição, ela pode ser utilizada em sentenças completas ou em subfórmulas.

6.5 Regras para identidade

O predicado de identidade não é parte da LQ, mas o adicionamos quando precisamos simbolizar certas sentenças. Para provas envolvendo identidade, adicionamos duas regras de prova.

Suponha que você sabe que muitas coisas que são verdadeiras para a também são para b. Por exemplo: $Aa\&Bb$, $Ba\&Bb$, $\neg Ca\&\neg Cb$, $Da\&Db$, $\neg Ea\&\neg Eb$ e assim por diante. Isso não seria suficiente para justificar $a = b$. (Ver p. 113). Em geral, não há sentenças que já não possuam o predicado de identidade que possam justificar a conclusão $a = b$. Isso significa que a regra de introdução de identidade não justificará $a = b$ ou qualquer outra asserção de identidade que contem duas constantes diferentes.

Contudo, é sempre verdadeiro que $a = a$. Em geral, nenhuma premissa é necessária para concluir que algo é idêntico a si mesmo. Portanto, essa será a regra de introdução de identidade, abreviada =I:

$$\mid\ \mathcal{c} = \mathcal{c} \qquad =\text{I}$$

Note que a regra =I não necessita se referir a qualquer linha anterior da prova. Para qualquer constante $\mathcal{c}$ você pode sempre escrever $\mathcal{c} = \mathcal{c}$ a qualquer passo com apenas a regra =I como justificação.

Se você tiver demonstrado que $a = b$, então qualquer coisa que seja verdade sobre a também deve ser verdade sobre b. Em qualquer sentença que contenha a, você pode substituir alguma ou todas as ocorrências de a por b e produzir assim uma sentença equivalente. Por exemplo, se você já sabe que Raa é o caso, então você está justificado em concluir Rab, Rba, Rbb. Lembre-se que $\mathcal{A}\boxed{\boxed{a \Rightarrow b}}$ é a sentença produzida ao substituir a em $\mathcal{A}$ por b. Isso não é o mesmo que uma instância de substituição, pois b pode substituir alguma ou todas as ocorrências de a. A regra de eliminação da identidade (=E) justifica a substituição de termos por outros termos que são idênticos a eles:

$$
\begin{array}{l|ll}
m & c = d & \\
n & \mathcal{A} & \\
 & \mathcal{A}\boxed{\boxed{a \Rightarrow b}} & =\text{E } m, n \\
 & \mathcal{A}\boxed{\boxed{b \Rightarrow a}} & =\text{E } m, n
\end{array}
$$

Para ver como essas regras são aplicadas, considere essa prova:

$$
\begin{array}{l|lll}
1 & \forall x \forall y\ x = y & & \\
2 & \exists x Bx & & \\
3 & \forall x(Bx \to \neg Cx) & & \text{busco } \neg\exists x Cx \\
\hline
4 & & Be & \\
5 & & \forall y\ e = y & \forall\text{E } 1 \\
6 & & e = f & \forall\text{E } 5 \\
7 & & Bf & =\text{E } 6, 4 \\
8 & & Bf \to \neg Cf & \forall\text{E } 3 \\
9 & & \neg Cf & \to\text{E } 8, 7 \\
10 & \neg Cf & & \exists\text{E } 2, 4\text{–}9 \\
11 & \forall x \neg Cx & & \forall\text{I } 10 \\
12 & \neg\exists x Cx & & \text{NQ } 11
\end{array}
$$

6.6 Estratégia de prova

Não há receita simples para provas e não há substituto para a prática. Todavia, aqui estão algumas regras práticas e estratégias que devemos ter em mente.

Retroceda a partir do que você busca. O objetivo final é derivar a conclusão. Olhe para a conclusão e se pergunte qual é a regra de introdução para seu operador lógico principal. Isso lhe dá uma ideia do que deve acontecer *um pouco* antes da última linha da prova. Aí você pode tratar dessa linha como se ela fosse seu objetivo. Pergunte o que você poderia fazer para derivá-la.

Por exemplo: se sua conclusão é uma condicional $\mathcal{A} \to \mathcal{B}$, planeje utilizar a regra $\to$I. Para isso, é necessário começar uma subprova na qual você supõe $\mathcal{A}$. Na subprova, você deve busca derivar $\mathcal{B}$.

Avance a partir do que você tem. Quando você estiver começando uma prova, olhe as premissas; posteriormente, olhe para as sentenças que você já derivou até então. Pense sobre as regras de eliminação para os operadores lógicos principais dessas sentenças. Elas dirão quais opções você tem.

Por exemplo: se você tem $\forall x\mathcal{A}$, pense sobre instanciá-la em qualquer outra constante que lhe possa ser útil. Se você tiver $\exists x\mathcal{A}$ e pretender utilizar a regra $\exists$E, então você deve supor $\mathcal{A}[c|x]$ para algum c que não está sendo utilizado e então derivar a conclusão que não contem c.

Para uma prova curta, você pode eliminar as premissas e introduzir a conclusão. Uma prova longa é formalmente apenas um número de provas curtas encadeadas conjuntamente, então você pode preencher a lacuna ao alternar retrocedendo a partir da conclusão e avançando a partir das premissas.

Mude o que você está vendo. Regras de substituição muitas vezes podem fazer sua vida mais fácil. Se uma prova parece impossível, tente algumas substituições diferentes.

Por exemplo: em geral é muito difícil provar uma disjunção utilizando as regras básicas. Se você quer demonstrar $\mathcal{A} \lor \mathcal{B}$,em geral, é mais fácil derivar $\neg\mathcal{A} \to \mathcal{B}$ para então utilizar a regra CM.

Demonstrar $\neg\exists x\mathcal{A}$ pode também ser difícil e, em geral, é mais fácil derivar $\forall x\neg\mathcal{A}$ para, então, utilizar a regra NQ.

A utilização de algumas regras de substituição deve se tornar sua segunda natureza. Por exemplo, se você notar uma disjunção negada, então você deve imediatamente pensar na regra de De Morgan.

Não se esqueça da prova indireta. Se você não consegue encontrar uma forma de demonstrar algo diretamente, tente supor sua negação.

Lembre-se que a maioria das provas podem ser feitas indireta ou diretamente. Uma forma pode ser mais fácil — ou talvez uma estimula sua imaginação mais que a outra —, mas qualquer uma das duas é formalmente legítima.

Repita o tanto quanto for necessário. Uma vez que você já tenha decidido como pode ser capaz de derivar a conclusão, pergunte-se como pode fazer o mesmo com as premissas. Então pense nas sentenças que você busca de novo e se pergunte como você pode conseguir derivá-las.

Persista. Tente coisas diferentes. Se uma abordagem falhar, tente outra.

6.7 Conceitos da teoria da prova

Utilizaremos o símbolo '$\vdash$' para indicar que uma prova é possível. Esse símbolo é chamado de catraca. Note que ele é chamado 'catraca' para sublinhar o fato de que ele não é o símbolo 'dupla catraca' ($\models$) que utilizamos para representar a implicação semântica no capítulo 5.

Quando escrevemos $\{\mathcal{A}_1, \mathcal{A}_2, \ldots\} \vdash \mathcal{B}$ isso significa que é possível fazer uma demonstração de $\mathcal{B}$ com $\mathcal{A}_1, \mathcal{A}_2, \ldots$ como premissas. Com apenas uma premissa, omitimos as chaves, de forma que $\mathcal{A} \vdash \mathcal{B}$ significa que há alguma prova de $\mathcal{B}$ com $\mathcal{A}$ como premissa. Naturalmente, $\vdash \mathcal{C}$ significa que há uma prova de $\mathcal{C}$ sem premissa alguma.

Frequentemente, provas lógicas são chamadas de derivações. Assim, $\mathcal{A} \vdash \mathcal{B}$ pode ser lido como ' $\mathcal{B}$ é derivável de $\mathcal{A}$.'

UM TEOREMA é uma sentença que é derivável sem quaisquer premissas; isto é, $\mathcal{T}$ é um teorema se e somente se $\vdash \mathcal{T}$.

Não é muito difícil de mostrar que algo é um teorema – você apenas tem que dar uma prova dele. Como você poderia demonstrar que algo *não* é um teorema? Se sua negação for um teorema, você poderia dar uma prova. Por exemplo, é fácil provar $\neg(Pa \& \neg Pa)$, o que demonstra que $(Pa \& \neg Pa)$ não pode ser um teorema. Entretanto, para uma sentença que não é um teorema, tampouco a negação de um teorema, não é fácil mostrar isso. Você precisaria demonstrar que não apenas certas estratégias de prova falham, mas que nenhuma prova é possível. Mesmo se você falhar em tentar provar uma sentença de milhares de formas diferentes, isso apenas significa que talvez a prova seja demasiado longa e complexa para você descobrir sua solução.

Duas sentenças $\mathcal{A}$ e $\mathcal{B}$ são DEMONSTRATIVAMENTE EQUIVALENTES se e somente se uma puder ser derivada da outra; isto é, $\mathcal{A} \vdash \mathcal{B}$ e $\mathcal{B} \vdash \mathcal{A}$.

É relativamente fácil de mostrar que duas sentenças são demonstrativamente equivalentes – isso só requer um par de provas. Mostrar que duas sentenças *não* são demonstrativamente equivalentes seria muito mais difícil. Seria tão difícil quanto demonstrar que uma sentença não é um teorema. (E de fato, esses problemas são intercambiáveis. Você consegue pensar em uma sentença que seria um teorema se e somente se $\mathcal{A}$ e $\mathcal{B}$ fossem demonstrativamente equivalentes?)

O conjunto de sentenças $\{\mathcal{A}_1, \mathcal{A}_2, \ldots\}$ é DEMONSTRATIVAMENTE INCONSISTENTE se e somente se uma contradição é derivável dele; isto é, para alguma sentença $\mathcal{B}$, $\{\mathcal{A}_1, \mathcal{A}_2, \ldots\} \vdash \mathcal{B}$ e $\{\mathcal{A}_1, \mathcal{A}_2, \ldots\} \vdash \neg\mathcal{B}$.

É fácil de mostrar que um conjunto é demonstrativamente inconsistente: você precisa apenas de supor as sentenças do conjunto e provar uma contradição. Mostrar que um conjunto *não* é demonstrativamente inconsistente será muito mais difícil. Para isso, seria necessário mais que apenas uma prova ou duas; seria necessário demonstrar que provas de um certo tipo são impossíveis.

6.8 Provas e modelos

Como você pode suspeitar, há uma conexão entre *teoremas* e *tautologias*.

Há uma maneira formal de mostrar que uma sentença é um teorema: prove-a. Em cada linha, podemos checar se ela segue da regra citada. Pode ser difícil produzir uma prova de vinte linhas, mas não é tão difícil de checar cada linha da prova e confirmar que ela é legítima. Contudo, para se demonstrar que uma sentença é uma tautologia, é necessário raciocinar em português sobre todos os modelos possíveis. Não há maneira formal para checar se o raciocínio é correto. Se pudermos escolher entre demonstrar que uma sentença é um teorema e mostrar que ela é uma tautologia, é mais fácil demonstrar que ela é um teorema.

Ao contrário, não há maneira formal de mostrar que uma sentença *não* é um teorema. Seria necessário que raciocinássemos em português sobre todas as provas possíveis. Ainda assim, há um método formal para demonstrar que uma sentença não é uma tautologia. Precisamos apenas construir um modelo no qual a sentença é falsa. Se pudermos escolher entre demonstrar que uma não é um teorema e demonstrar que ela não é uma tautologia, é mais fácil demonstrar que ela não é uma tautologia.

Felizmente, uma sentença é um teorema se e somente se ela é uma tautologia. Se fizermos uma prova de $\vdash \mathcal{A}$ e assim mostrarmos que ela é um teorema, segue-se que $\mathcal{A}$ é uma tautologia; isto é, $\vDash \mathcal{A}$. Similarmente, se construirmos um modelo no qual $\mathcal{A}$ é falsa e assim mostrarmos que ela não é uma tautologia, segue-se que $\mathcal{A}$ não é um teorema.

Em geral, $\mathcal{A} \vdash \mathcal{B}$ se e somente se $\mathcal{A} \vDash \mathcal{B}$. Desse modo:

▷ Um argumento é *válido* se e somente se *a conclusão é derivável das premissas.*

▷ Duas sentenças são *logicamente equivalentes* se e somente se elas são *demonstrativamente equivalentes.*

▷ Um conjunto de sentenças é *consistente* se e somente se ele *não é demonstrativamente inconsistente.*

Você pode decidir quando pensar em termos de provas e quando pensar em termos de modelos, utilizando o método mais fácil para dada tarefa. A tabela 6.1 abaixo resume quando é melhor fazer provas e quando é melhor construir modelos.

Dessa maneira, provas e modelos nos proporcionam um conjunto de ferramentas versáteis para considerar os argumentos. Se pudermos traduzir um argumento para LQ, então poderemos julgar seu valor lógico de maneira puramente formal. Se ele for dedutivamente válido, podemos fornecer uma prova formal; se ele for inválido, podemos fonecer um contraexemplo formal.

6.9 Correção e completude

Esse conjunto de ferramentas é muito conveniente. Ele também é intuitivo, porque parece natural que demonstrabilidade e implicação semântica coincidam.

	SIM	NÃO
$\mathcal{A}$ é tautológica?	Prove $\vdash \mathcal{A}$	Construa um modelo no qual $\mathcal{A}$ seja falsa
$\mathcal{A}$ é contraditória?	$\vdash \neg\mathcal{A}$	Construa um modelo no qual $\mathcal{A}$ seja verdadeira
$\mathcal{A}$ é contingente?	Construa um modelo no qual $\mathcal{A}$ seja verdadeira e outro no qual $\mathcal{A}$ seja falsa	Prove $\vdash \mathcal{A}$ ou $\vdash \neg\mathcal{A}$
$\mathcal{A}$ e $\mathcal{B}$ são equivalentes?	Prove $\mathcal{A} \vdash \mathcal{B}$ e $\mathcal{B} \vdash \mathcal{A}$	Construa um modelo no qual $\mathcal{A}$ e B tenham valores de verdade diferentes
O conjunto $\mathbb{A}$ é consistente?	Construa um modelo no qual todas as sentenças em $\mathbb{A}$ são verdadeiras	A partir das sentenças de $\mathbb{A}$, prove $\mathcal{B}$ e $\neg\mathcal{B}$
O argumento $\mathcal{P} \therefore \mathcal{C}$ é válido?	Prove $\mathcal{P} \vdash \mathcal{C}$	Construa um modelo no qual $\mathcal{P}$ é verdadeiro e $\mathcal{C}$ é falso

Tabela 6.1: Algumas vezes é mais fácil demonstrar algo fazendo provas do que construindo modelos. Algumas vezes o contrário. Depende do que você está querendo demonstrar.

Contudo, não se engane com a semelhança dos símbolos '$\vDash$' e '$\vdash$'. O fato de que esses dois são realmente intercambiáveis não é uma coisa simples de provar.

Por que devemos pensar que um argumento que *pode ser provado* é necessariamente um argumento *válido*? Isto é, porque pensar que $\mathcal{A} \vdash \mathcal{B}$ implica $\mathcal{A} \vDash \mathcal{B}$?

Esse é o problema da correção. Um sistema de prova é CORRETO se não há provas de argumentos inválidos. Para demonstrar que um sistema de prova é correto, é necessário mostrar que *qualquer* prova possível é a prova de um argumento válido. Não seria suficiente simplesmente conseguir provar muitos argumentos válidos e falhar ao tentar provar os inválidos.

Felizmente, há uma maneira de abordar isso pormenorizadamente. Se utilizarmos a regra &E na última linha de uma prova, nunca poderemos fazer de um argumento válido, um argumento inválido. Ou seja, utilizar a mesma regra várias vezes não torna inválido um argumento válido. Similarmente, se utilizarmos as regras &E e $\vee$E individualmente na última linha de uma prova, nunca poderemos tornar um argumento inválido. Ou seja, utilizá-las conjuntamente também não o torna inválido.

A estratégia é mostrar que, para toda regra de inferência, elai soladamente não tornaria inválido um argumento válido. Já que uma prova é apenas uma série de linhas, cada uma justificada por uma regra de inferência, isso demonstraria que todo argumento demonstrável é válido.

Por exemplo, considere a regra &I. Suponha que a usemos para adicionar $\mathcal{A}\&\mathcal{B}$ em um argumento válido. Para a regra ser aplicada, $\mathcal{A}$ e $\mathcal{B}$ já devem ter ocorrido na prova. Já que até então o argumento é válido, $\mathcal{A}$ e $\mathcal{B}$ são ou premissas do argumento ou consequências válidas das premissas. Desse modo, quaisquer modelos nos quais as premissas são verdadeiras também são modelos nos quais $\mathcal{A}$ e $\mathcal{B}$ são verdadeiras. De acordo com a definição de VERDADE EM LQ, isso significa que $\mathcal{A}\&\mathcal{B}$ também é verdadeira em tal modelo. Portanto, $\mathcal{A}\&\mathcal{B}$ segue validamente das premissas. Isso significa que utilizar a regra &E para desenvolver uma prova válida produz outra prova válida.

Para provar que um sistema de provas é correto, precisaríamos mostrar que esse é o caso para as outras regras de inferência. Já que as regras derivadas são consequências das regras básicas, seria suficiente fazer argumentos similares para as outras 16 regras básicas. Esse exercício tedioso está além do escopo deste livro.

Dada uma prova de que o sistema de prova é correto, segue-se que todo teorema é uma tautologia.

Ainda é possível perguntar: por que pensamos que *todo* argumento válido é um argumento que pode ser provado? Isto é, porque pensar que $\mathcal{A} \models \mathcal{B}$ implica $\mathcal{A} \vdash \mathcal{B}$?

Esse é o problema da COMPLETUDE. Um sistema de prova é COMPLETO se há uma prova para todo argumento válido. A completude para uma linguagem como a LQ foi provada pela primeira por Kurt Gödel em 1929. A prova está além do escopo deste livro.

Felizmente, a questão importante é que o sistema de prova para LQ é tanto correto quanto completo. Esse não é o caso para todos os sistemas de prova e todas linguagens formais. Dado que isso é verdadeiro para a LQ, podemos escolher entre fazer provas ou construir modelos — o que seja o mais fácil para a tarefa em questão.

Resumo de definições

- ▷ Uma sentença $\mathcal{A}$ é um TEOREMA se e somente se $\vdash \mathcal{A}$.
- ▷ Duas sentenças $\mathcal{A}$ e $\mathcal{B}$ são DEMONSTRATIVAMENTE EQUIVALENTES se e somente se $\mathcal{A} \vdash \mathcal{B}$ e $\mathcal{B} \vdash \mathcal{A}$.
- ▷ $\{\mathcal{A}_1, \mathcal{A}_2, \ldots\}$ é demonstrativamente inconsistente se e somente se para alguma sentença $\mathcal{B}$, $\{\mathcal{A}_1, \mathcal{A}_2, \ldots\} \vdash \mathcal{B} \& \neg\mathcal{B}$.

6.10 Exercícios práticos

⋆ **Parte A** Forneça uma justificação (regra e números da linha) para cada linha de prova que requer justificação.

$$
\begin{array}{r|l}
1 & W \to \neg B \\
2 & A \,\&\, W \\
3 & B \lor (J \,\&\, K) \\
\hline
4 & W \\
5 & \neg B \\
6 & J \,\&\, K \\
7 & K
\end{array}
$$

$$
\begin{array}{r|l}
1 & L \leftrightarrow \neg O \\
2 & L \lor \neg O \\
\hline
3 & \quad \begin{array}{|l} \neg L \\ \hline \end{array} \\
4 & \quad \begin{array}{|l} \neg O \end{array} \\
5 & \quad \begin{array}{|l} L \end{array} \\
6 & \quad \begin{array}{|l} \neg L \end{array} \\
7 & L
\end{array}
$$

1	$Z \rightarrow (C \,\&\, \neg N)$		
2	$\neg Z \rightarrow (N \,\&\, \neg C)$		
3		$\neg(N \vee C)$	
4		$\neg N \,\&\, \neg C$	
5			Z
6			$C \,\&\, \neg N$
7			C
8			$\neg C$
9		$\neg Z$	
10		$N \,\&\, \neg C$	
11		N	
12		$\neg N$	
13	$N \vee C$		

$\star$ **Parte B** Forneça uma prova para cada argumento em LS.

1. $K \,\&\, L, \therefore K \leftrightarrow L$
2. $A \rightarrow (B \rightarrow C), \therefore (A \,\&\, B) \rightarrow C$
3. $P \,\&\, (Q \vee R),\ P \rightarrow \neg R, \therefore Q \vee E$
4. $(C \,\&\, D) \vee E, \therefore E \vee D$
5. $\neg F \rightarrow G,\ F \rightarrow H, \therefore G \vee H$
6. $(X \,\&\, Y) \vee (X \,\&\, Z),\ \neg(X \,\&\, D),\ D \vee M\ \therefore M$

Parte C Forneça uma prova para cada argumento em LS.

1. $Q \rightarrow (Q \,\&\, \neg Q),\ \therefore\ \neg Q$
2. $J \rightarrow \neg J,\ \therefore\ \neg J$
3. $E \vee F,\ F \vee G,\ \neg F,\ \therefore\ E \,\&\, G$
4. $A \leftrightarrow B,\ B \leftrightarrow C,\ \therefore\ A \leftrightarrow C$
5. $M \vee (N \rightarrow M),\ \therefore\ \neg M \rightarrow \neg N$
6. $S \leftrightarrow T,\ \therefore\ S \leftrightarrow (T \vee S)$
7. $(M \vee N) \,\&\, (O \vee P),\ N \rightarrow P,\ \neg P,\ \therefore\ M \,\&\, O$
8. $(Z \,\&\, K) \vee (K \,\&\, M),\ K \rightarrow D,\ \therefore\ D$

Parte D Mostre que as seguintes sentenças são teoremas em LS.

1. $O \rightarrow O$
2. $N \vee \neg N$
3. $\neg(P \,\&\, \neg P)$
4. $\neg(A \rightarrow \neg C) \rightarrow (A \rightarrow C)$
5. $J \leftrightarrow [J \vee (L \,\&\, \neg L)]$

Parte E Mostre que os seguintes pares de sentenças são equivalentes em LS.

1. $\neg\neg\neg\neg G$, G
2. $T \rightarrow S$, $\neg S \rightarrow \neg T$
3. $R \leftrightarrow E$, $E \leftrightarrow R$
4. $\neg G \leftrightarrow H$, $\neg(G \leftrightarrow H)$
5. $U \rightarrow I$, $\neg(U \,\&\, \neg I)$

Parte F Forneça provas para cada um dos seguintes argumentos.

1. $M \,\&\, (\neg N \rightarrow \neg M) \vdash (N \,\&\, M) \vee \neg M$
2. $\{C \rightarrow (E \,\&\, G), \neg C \rightarrow G\} \vdash G$
3. $\{(Z \,\&\, K) \leftrightarrow (Y \,\&\, M), D \,\&\, (D \rightarrow M)\} \vdash Y \rightarrow Z$
4. $\{(W \vee X) \vee (Y \vee Z), X \rightarrow Y, \neg Z\} \vdash W \vee Y$

Parte G Forneça provas para as seguintes sentenças, utilizando apenas as regras básicas. As provas serão mais longas do que elas seriam se as regras derivadas fossem utilizadas.

1. Mostre que o MT é uma regra derivada legítima. Utilizando apenas as regras básicas, prove o seguinte: $\mathcal{A}\rightarrow\mathcal{B}$, $\neg\mathcal{B}$ $\therefore$ $\neg\mathcal{A}$
2. Mostre que a Com é uma regra legítima para a bicondicional. Utilizando apenas as regras básicas, prove o seguinte: $\mathcal{A} \leftrightarrow \mathcal{B}$ e $\mathcal{B} \leftrightarrow \mathcal{A}$ são equivalentes.
3. Utilizando apenas as regras básicas, prove a seguinte instância das Leis de De Morgan: $(\neg A \,\&\, \neg B)$ $\therefore$ $\neg(A \vee B)$
4. Sem utilizar a regra NQ, prove $\neg\exists x\neg\mathcal{A} \vdash \forall x\mathcal{A}$
5. Mostre que $\leftrightarrow c$ é uma regra derivada legítima. Utilizando apenas as regras básicas, prove que $D \leftrightarrow E$ e $(D \rightarrow E) \,\&\, (E \rightarrow D)$ são equivalentes.

⋆ **Parte H** Forneça uma justificação (a regra e os números das linhas) para cada linha da prova que necessitar de uma.

1	$\forall x \exists y(Rxy \lor Ryx)$
2	$\forall x \neg Rmx$
3	$\exists y(Rmy \lor Rym)$
4	$\quad\mid Rma \lor Ram$
5	$\quad\mid \neg Rma$
6	$\quad\mid Ram$
7	$\quad\mid \exists x Rxm$
8	$\exists x Rxm$

1	$\forall x(\exists y Lxy \to \forall z Lzx)$
2	Lab
3	$\exists y Lay \to \forall z Lza$
4	$\exists y Lay$
5	$\forall z Lza$
6	Lca
7	$\exists y Lcy \to \forall z Lzc$
8	$\exists y Lcy$
9	$\forall z Lzc$
10	Lcc
11	$\forall x Lxx$

1	$\forall x(Jx \to Kx)$
2	$\exists x \forall y Lxy$
3	$\forall x Jx$
4	$\quad\mid \forall y Lay$
5	$\quad\mid Ja$
6	$\quad\mid Ja \to Ka$
7	$\quad\mid Ka$
8	$\quad\mid Laa$
9	$\quad\mid Ka \,\&\, Laa$
10	$\quad\mid \exists x(Kx \,\&\, Lxx)$
11	$\exists x(Kx \,\&\, Lxx)$

1	$\quad\mid \neg(\exists x Mx \lor \forall x \neg Mx)$
2	$\quad\mid \neg\exists x Mx \,\&\, \neg\forall x \neg Mx$
3	$\quad\mid \neg\exists x Mx$
4	$\quad\mid \forall x \neg Mx$
5	$\quad\mid \neg\forall x \neg Mx$
6	$\exists x Mx \lor \forall x \neg Mx$

⋆ **Parte I** Forneça uma prova para cada sentença.

1. $\vdash \forall x Fx \lor \neg\forall x Fx$
2. $\{\forall x(Mx \leftrightarrow Nx), Ma \,\&\, \exists x Rxa\} \vdash \exists x Nx$
3. $\{\forall x(\neg Mx \lor Ljx), \forall x(Bx \to Ljx), \forall x(Mx \lor Bx)\} \vdash \forall x Ljx$
4. $\forall x(Cx \,\&\, Dt) \vdash \forall x Cx \,\&\, Dt$
5. $\exists x(Cx \lor Dt) \vdash \exists x Cx \lor Dt$

Parte J Forneça uma prova do argumento sobre Billy na p. 77.

Parte K Retorne ao exercício 'Parte B' na p. 92. Forneça provas que mostrem que cada uma das formas de argumento é válida em LQ.

Parte L Aristóteles e seus sucessores identifcaram outros modos silogísticos além daqueles que apresentamos no exercício Parte B do Capítulo 4. Simbolize cada um dos seguintes modos de argumentos em LQ e acrescente as seguintes hipóteses 'Há algum A' e 'Há algum B.' Então prove que as formas de argumento abaixo são válidas em LQ.

Darapti: Todo A é B. Todo A é C. $\therefore$ Algum B é C.

Felapton: Nenhum B é C. Todo A é B. $\therefore$ Algum A não é C.

Barbari: Todo B é C. Todo A é B. $\therefore$ Algum A é C.

Camestros: Todo C é B. Nenhum A é B. $\therefore$ Algum A não é C.

Celaront: Nenhum B é C. Todo A é B. $\therefore$ Algum A não é C.

Cesaro: Nenhum C é B. Todo A é B. $\therefore$ Algum A não é C.

Fapesmo: Todo B é C. Nenhum A é B. $\therefore$ Algum C não é A.

Parte M Forneça uma prova para cada argumento.

1. $\forall x \forall y Gxy \vdash \exists x Gxx$
2. $\forall x \forall y (Gxy \rightarrow Gyx) \vdash \forall x \forall y (Gxy \leftrightarrow Gyx)$
3. $\{\forall x (Ax \rightarrow Bx), \exists x Ax\} \vdash \exists x Bx$
4. $\{Na \rightarrow \forall x (Mx \leftrightarrow Ma), Ma, \neg Mb\} \vdash \neg Na$
5. $\vdash \forall z (Pz \vee \neg Pz)$
6. $\vdash \forall x Rxx \rightarrow \exists x \exists y Rxy$
7. $\vdash \forall y \exists x (Qy \rightarrow Qx)$

Parte N Mostre que cada par de sentenças é demonstrativamente equivalente.

1. $\forall x (Ax \rightarrow \neg Bx)$, $\neg \exists x (Ax \,\&\, Bx)$
2. $\forall x (\neg Ax \rightarrow Bd)$, $\forall x Ax \vee Bd$
3. $\exists x Px \rightarrow Qc$, $\forall x (Px \rightarrow Qc)$

Parte O Mostre que cada par de sentenças é demonstrativamente inconsistente.

1. $\{Sa \to Tm,\ Tm \to Sa,\ Tm \,\&\, \neg Sa\}$
2. $\{\neg\exists x Rxa,\ \forall x\forall y Ryx\}$
3. $\{\neg\exists x\exists y Lxy,\ Laa\}$
4. $\{\forall x(Px \to Qx),\ \forall z(Pz \to Rz),\ \forall y Py,\ \neg Qa \,\&\, \neg Rb\}$

⋆ **Parte P** Escreva uma chave de simbolização para o seguinte argumento, traduza e prove:

Há alguem que gosta de todos que gostam de todos que ele gosta.

Portanto, há alguém que gosta de si mesmo.

Parte Q Forneça uma prova para cada afirmação.

1. $\{Pa \vee Qb, Qb \to b = c, \neg Pa\} \vdash Qc$
2. $\{m = n \vee n = o, An\} \vdash Am \vee Ao$
3. $\{\forall x x = m, Rma\} \vdash \exists x Rxx$
4. $\neg\exists x x \neq m \vdash \forall x\forall y(Px \to Py)$
5. $\forall x\forall y(Rxy \to x = y) \vdash Rab \to Rba$
6. $\{\exists x Jx, \exists x \neg Jx\} \vdash \exists x\exists y\ x \neq y$
7. $\{\forall x(x = n \leftrightarrow Mx), \forall x(Ox \vee \neg Mx)\} \vdash On$
8. $\{\exists x Dx, \forall x(x = p \leftrightarrow Dx)\} \vdash Dp$
9. $\{\exists x\big[Kx \,\&\, \forall y(Ky \to x = y) \,\&\, Bx\big], Kd\} \vdash Bd$
10. $\vdash Pa \to \forall x(Px \vee x \neq a)$

Parte R Retorne ao exercício 'Parte D' na p. 93. E para cada um daqueles argumentos: se ele for válido em LQ, forneça sua prova. Se ele for inválido, construa um modelo para mostrar que ele é inválido.

⋆ **Parte S** Para cada uma dos seguintes pares de sentenças: se elas forem logicamente equivalentes em LQ, forneça provas que mostrem isso. Se elas não forem, construa um modelo que mostre isso.

1. $\forall x Px \to Qc,\ \forall x(Px \to Qc)$
2. $\forall x Px \,\&\, Qc,\ \forall x(Px \,\&\, Qc)$

3. $Qc \vee \exists x Qx$, $\exists x(Qc \vee Qx)$
4. $\forall x \forall y \forall z Bxyz$, $\forall x Bxxx$
5. $\forall x \forall y Dxy$, $\forall y \forall x Dxy$
6. $\exists x \forall y Dxy$, $\forall y \exists x Dxy$

⋆ **Parte T** Para cada um dos seguintes argumentos: se ele for válido em LQ, forneça uma prova. Se ele for inválido, construa um modelo que mostre que ele é inválido.

1. $\forall x \exists y Rxy \therefore \exists y \forall x Rxy$
2. $\exists y \forall x Rxy \therefore \forall x \exists y Rxy$
3. $\exists x(Px \,\&\, \neg Qx) \therefore \forall x(Px \to \neg Qx)$
4. $\forall x(Sx \to Ta)$, $Sd \therefore Ta$
5. $\forall x(Ax \to Bx)$, $\forall x(Bx \to Cx) \therefore \forall x(Ax \to Cx)$
6. $\exists x(Dx \vee Ex)$, $\forall x(Dx \to Fx) \therefore \exists x(Dx \,\&\, Fx)$
7. $\forall x \forall y(Rxy \vee Ryx) \therefore Rjj$
8. $\exists x \exists y(Rxy \vee Ryx) \therefore Rjj$
9. $\forall x Px \to \forall x Qx$, $\exists x \neg Px \therefore \exists x \neg Qx$
10. $\exists x Mx \to \exists x Nx$, $\neg \exists x Nx \therefore \forall x \neg Mx$

Parte U

1. Se você sabe que $\mathcal{A} \vdash \mathcal{B}$, o que você pode dizer sobre $(\mathcal{A} \,\&\, \mathcal{C}) \vdash \mathcal{B}$? Explique sua resposta.
2. Se você sabe que $\mathcal{A} \vdash \mathcal{B}$, o que você pode dizer sobre $(\mathcal{A} \vee \mathcal{C}) \vdash \mathcal{B}$? Explique sua resposta.

Apêndice A

Outras notações simbólicas

Na história da lógica formal, diferentes símbolos foram utilizados em tempos diferentes por autores diferentes. Frequentemente, os autores eram forçados a utilizar a notação que suas impressoras podiam imprimir.

Em certo sentido, os símbolos utilizados para várias constantes lógicas são arbitrários. Não há coisa alguma escrita no céu que diga: '$\neg$' é um símbolo para a negação verofuncional. Poderíamos ter escolhido outro símbolo para a mesma função. Contudo, uma vez que já demos as definições de fórmulas bem formadas (fbf*)* e de verdade em nossa linguagem lógica, a utilização do símbolo '$\neg$' não é mais arbitrária. Esse é o símbolo para negação neste manual de lógica, assim, ele também é o símbolo para a negação ao escrevemos sentenças em nossas linguagens de LS ou LQ.

Esse apêndice apresenta alguns símbolos comuns, para que você possa reconhecê-los ao encontrá-los em um artigo ou em outro livro.

Negação Dois símbolos comumente utilizados são a negação, '$\neg$', e o til '$\sim$'. Em alguns sistemas formais mais avançados, é necessário distinguir dois tipos de negação; a distinção é, as vezes, representada usando '$\neg$' e '$\sim$'.

Disjunção O símbolo '$\vee$' é geralmente utilizado para simbolizar a disjunção inclusiva.

Conjunção A conjunção é frequentemente simbolizada com o termo '&', chamado pelo expressão inglesa *ampersand* ou 'sinal tironiano' em sua versão portuguesa. Tal símbolo é uma forma decorativa da palavra latina 'et' e significa 'e'. Como um símbolo em um sistema formal, a cojunção não é a palavra 'e';

seu significado é dado pela semântica formal da linguagem. A fim de se evitar essa confusão, alguns sistemas usam um símbolo diferente para a conjunção. Por exemplo, '$\wedge$' é a contraparte do símbolo utilizado para disjunção. Algumas vezes um simples ponto '$\bullet$' é utilizado. Em alguns textos mais antigos, não há símbolo algum para conjunção; 'A e B' é simplesmente escrito $'AB'$.

Condicional material Existem dois símbolos comuns para a condicional material: a seta, '$\rightarrow$' e a ferradura, '$\supset$'.

Bicondicional material A seta de dupla implicação, '$\leftrightarrow$', é utilizada em sistemas que usam a seta para representar o condicional material. Sistemas que utilizam a ferradura para o condicional geralmente utilizam a barra tripla, '$\equiv$', para o bicondicional.

Quantificadores O quantificador universal é geralmente simbolizado com um A virado de cabeça para baixo, '$\forall$', e o quantificador existencial com um E invertido, '$\exists$'. Em alguns textos, não há símbolo extra para o quantificador universal. Em vez disso, a variável é apenas escrita entre parênteses em frente da fórmula a qual ela está ligada. Por exemplo, 'todos x são P' é escrito $(x)\,Px$.

Em alguns sistemas, os quantificadores são simbolizados com versões maiores dos símbolos utilizados para a conjunção e a disjunção. Apesar de expressões quantificadas não poderem ser traduzidas em expressões sem quantificadores, tanto há uma conexão conceitual entre o quantificador universal e a conjunção, quanto há uma entre o quantificador existencial e a disjunção. Por exemplo, considere a sentença $\exists xPx$. Isso significa que *ou* o primeiro membro do UD é um P, *ou* o segundo é, *ou* o terceiro é etc. Tal sistema utiliza o símbolo '$\bigvee$' em vez de '$\exists$'.

Notação polonesa

Essa seção discute brevemente a lógica sentencial na notação polonesa, um sistema de notação introduzido ao final da década de 1920 pelo lógico polonês Jan Łukasiewicz.

Letras minúsculas são utilizadas como letras sentencias. A letra maiúscula N é utilizada para negação. A utilizada para disjunção, K para conjunção, C para condicional, E para bicondicional. ('A' significa alternância, outro nome para a disjunção lógica; 'E' significa equivalência.)

Na notação polonesa, um conectivo binário é escrito *antes* das duas sentenças que ele conecta. Por exemplo, a sentença $A\&B$ de LS seria escrita Kab na notação polonesa.

As sentenças $\neg A \rightarrow B$ e $\neg\,(A \rightarrow B)$ são muito diferentes; o operador lógico principal da primeira é o condicional, mas o principal da segunda é a negação. Em LS, podemos mostrar isso colocando parênteses ao redor do condicional na segunda sentença. Na notação polonesa, parênteses nunca são necessários. O conectivo mais à esquerda é sempre o conectivo principal. A primeira sentença seria simplesmente escrita CNab e a segunda NCab.

notação de LS	notação polonesa
$\neg$	N
$\&$	K
$\vee$	A
$\rightarrow$	C
$\leftrightarrow$	E

Essa característica da notação polonesa significa que é possível avaliar sentenças simplesmente trabalhando com os símbolos da direita para a esquerda. Por exemplo, se você estivesse construindo uma tabela de verdade para $NKab$, você primeiro consideraria os valores de verdade 'atribuídos' a a e b e, então, negaria o resultado. A regra geral para saber o que deve ser avaliado a seguir na LS nem de longe é tão simples. Em LS, tabelas de verdade para $\neg(A\&B)$ requerem que olhemos para A e B, para então olharmos o meio da sentença, na conjunção, para finalmente olharmos para o início da sentença, na negação. Uma vez que a ordem das operações pode ser especificada na notação polonesa, variantes da notação polonesa são utilizadas na estrutura interna de muitas linguagens de programação de computadores.

Apêndice B

Soluções para os exercícios selecionados

Muitos exercícios podem ser respondidos corretamente de várias maneiras. Quando isso for o caso, a solução aqui representa uma resposta correta possível.

Capítulo 1 Parte C

1. consistente
2. inconsistente
3. consistente
4. consistente

Capítulo 1 Parte D 1, 2, 3, 6, 8, e 10 são possíveis.

Capítulo 2 Parte A

1. $\neg M$
2. $M \vee \neg M$
3. $G \vee C$
4. $\neg C \,\&\, \neg G$
5. $C \rightarrow (\neg G \,\&\, \neg M)$
6. $M \vee (C \vee G)$

Capítulo 2 Parte C

1. $E_1 \& E_2$
2. $F_1 \rightarrow S_1$
3. $F_1 \vee E_1$
4. $E_2 \& \neg S_2$
5. $\neg E_1 \& \neg E_2$
6. $E_1 \& E_2 \& \neg(S_1 \vee S_2)$
7. $S_2 \rightarrow F_2$
8. $(\neg E_1 \rightarrow \neg E_2) \& (E_1 \rightarrow E_2)$
9. $S_1 \leftrightarrow \neg S_2$
10. $(E_2 \& F_2) \rightarrow S_2$
11. $\neg(E_2 \& F_2)$
12. $(F_1 \& F_2) \leftrightarrow (\neg E_1 \& \neg E_2)$

Capítulo 2 Parte D

A: Alice é uma espiã.
D: A Duquesa.
C: O código foi decifrado.
E: A embaixada alemã estará uma confusão.

1. $A \& D$
2. $(A \vee D) \rightarrow C$
3. $\neg(A \vee D) \rightarrow \neg C$
4. $E \vee C$
5. $(C \vee \neg C) \& E$
6. $(A \vee D) \& \neg(A \& D)$

Capítulo 2 Parte G

1. (a) não (b) não
2. (a) não (b) sim
3. (a) sim (b) sim
4. (a) não (b) não
5. (a) sim (b) sim
6. (a) não (b) não
7. (a) não (b) sim
8. (a) não (b) sim
9. (a) não (b) não

Capítulo 3 Parte A

1. tautológica
2. contraditória
3. contingente
4. tautológica
5. tautológica
6. contingente
7. tautológica
8. contraditória
9. tautológica
10. contraditória
11. tautológica
12. contingente
13. contraditória
14. contingente
15. tautológica
16. tautológica
17. contingente
18. contingente

Capítulo 3 Parte B 2, 3, 5, 6, 8, e 9 são logicamente equivalentes.

Capítulo 3 Parte C 1, 3, 6, 7, e 8 são consistentes.

Capítulo 3 Parte D 3, 5, 8, e 10 são válidos.

Capítulo 3 Parte E

1. $\mathcal{A}$ e $\mathcal{B}$ têm o mesmo valor de verdade em cada linha de uma tabela de verdade completa, então $\mathcal{A} \leftrightarrow \mathcal{B}$ é verdadeira em todas as linhas. Ela é uma tautologia.
2. A sentença é falsa em uma linha de uma tabela de verdade completa. Naquela linha, $\mathcal{A}$ e $\mathcal{B}$ são verdadeiras e $\mathcal{C}$ é falsa. Portanto, o argumento é inválido.
3. Já que não linha alguma em uma tabela de verdade completa na qual todas as três sentenças são verdadeiras, a conjunção é falsa em todas as linhas. Portanto, ela é uma contradição.

4. Já que $\mathcal{A}$ é falsa em todas as linhas de uma tabela de verdade completa, não há linha alguma na qual $\mathcal{A}$ e $\mathcal{B}$ são verdadeiras e $\mathcal{C}$ é falsa. Portanto, o argumento é válido.
5. Já que $\mathcal{C}$ é verdadeira em toda linha d euma tabela de verdade completa, não há linha alguma na qual $\mathcal{A}$ e $\mathcal{B}$ são verdadeiras e $\mathcal{C}$ é falsa. Portanto, o argumento é válido.
6. Não há muito o que se dizer. $(\mathcal{A} \vee \mathcal{B})$ é uma tautologia se $\mathcal{A}$ e $\mathcal{B}$ são tautologias; ela é uma contradição se elas são contradições; ela é contingente se elas são contingentes.
7. $\mathcal{A}$ e $\mathcal{B}$ têm valores de verdade diferentes em ao menos uma linha de uma tabela de verdade completa e $(\mathcal{A} \vee \mathcal{B})$ é verdadeira naquela linha. Nas outras linhas, ela pode ser verdadeira ou falsa. Portanto, $(\mathcal{A} \vee \mathcal{B})$ é ou tautológica ou contingente; ela *não* é uma contradição.

Capítulo 3 Parte F

1. $\neg A \rightarrow B$
2. $\neg(A \rightarrow \neg B)$
3. $\neg[(A \rightarrow B) \rightarrow \neg(B \rightarrow A)]$

Capítulo 4 Parte A

1. $Za \,\&\, Zb \,\&\, Zc$
2. $Rb \,\&\, \neg Jb$
3. $Lcb \rightarrow Mb$
4. $(Jb \,\&\, Jc) \rightarrow (Lab \,\&\, Lac)$
5. $\exists x(Rx \,\&\, Zx)$
6. $\forall x(Jx \rightarrow Rx)$
7. $\forall x\big[Zx \rightarrow (Mx \vee Jx)\big]$
8. $\exists x(Rx \,\&\, \neg Jx)$
9. $\exists x(Rx \,\&\, Lcx)$
10. $\forall x\big[(Mx \,\&\, Zx) \rightarrow Lbx\big]$
11. $\forall x\big[(Mx \,\&\, Lax) \rightarrow Lxa\big]$
12. $\exists x Rx \rightarrow Ra$
13. $\forall x(Jx \rightarrow Rx)$
14. $\forall x\big[(Mx \,\&\, Lcx) \rightarrow Lax\big]$
15. $\exists x(Mx \,\&\, Lxb \,\&\, \neg Lbx)$

Capítulo 4 Parte E

1. $\neg\exists x Bx$
2. $\forall x(Mx \to Ax)$
3. $\exists x \neg Ax$
4. $\exists x[Cx \,\&\, \neg\exists y Lyx]$
5. $\neg\exists x Lxx$
6. $\neg\exists x(Cx \,\&\, \neg Ax \,\&\, Bx)$
7. $\exists x(Cx \,\&\, Bx) \,\&\, \exists x(Mx \,\&\, Bx) \,\&\, \neg\exists x(Cx \,\&\, Mx \,\&\, Bx)$
8. $\forall x[Cx \to \forall y(\neg Cy \to Lxy)]$
9. $\forall x\big((Cx \,\&\, Mx) \to \forall y[(\neg Cy \,\&\, \neg My) \to Lxy]\big)$

Capítulo 4 Parte G

1. $\forall x(Fxp \to Dx)$
2. $Fjp \,\&\, Mj$
3. $\exists x(Fxp \,\&\, Mx)$
4. $\neg\exists x Ixj$
5. $\forall x\big[(Fxp \,\&\, Fx) \to Dx\big]$
6. $\neg\exists x(Fxp \,\&\, Hx)$
7. $\exists x(Fjx \,\&\, Ixe \,\&\, Mj)$
8. $Ipe \,\&\, Hp$
9. $\forall x\big[(Ixp \,\&\, Hx) \to \neg\exists y Fyx\big]$
10. $\exists x(Ixj \,\&\, \exists y Fyx \,\&\, Mj)$
11. $\forall x\big[Dx \to \exists y(Ixy \,\&\, My \,\&\, Dy)\big]$
12. $\forall x\big[(Hx \,\&\, Dx) \to \exists y(Fxy \,\&\, Dy)\big]$

Capítulo 6 Parte ??

1. Rca, Rcb, Rcc e Rcd são instâncias de substituição de $\forall x Rcx$.
2. Das expressões listadas, apenas $\forall y Lby$ é uma instância de susbtituição de $\exists x\forall y Lxy$.

Capítulo 4 Parte J

1. $\forall x(Px \to Rx)$
2. $\neg\exists x Sx$

3. $\exists x \exists y(Px \,\&\, Py \,\&\, x \neq y)$
4. $\exists x \exists y(Vx \,\&\, Fx \,\&\, Vy \,\&\, Fy \,\&\, x \neq y)$
5. $\forall x \forall y \forall z\big[(Vx \,\&\, Fx \,\&\, Vy \,\&\, Fy \,\&\, Vz \,\&\, Fz) \to (x = y \vee x = z \vee y = z)\big]$
6. $\exists x \exists y\big(Vx \,\&\, Rx \,\&\, Vy \,\&\, Ry \,\&\, x \neq y \,\&\, \forall z[(Vz \,\&\, Rz) \to (x = z \vee y = z)]\big)$
7. $\exists x_1 \exists x_2 \exists x_3 \exists x_4\big[Dx_1 \,\&\, Dx_2 \,\&\, Dx_3 \,\&\, Dx_4 \,\&\, x_1 \neq x_2 \,\&\, x_1 \neq x_3 \,\&\, x_1 \neq x_4 \,\&\, x_2 \neq x_3 \,\&\, x_2 \neq x_4 \,\&\, x_3 \neq x_4 \,\&\, \neg\exists y(Dy \,\&\, y \neq x_1 \,\&\, y \neq x_2 \,\&\, y \neq x_3 \,\&\, y \neq x_4)\big]$
8. $\exists x\big(Dx \,\&\, Px \,\&\, \forall y[(Dy \,\&\, Py) \to x = y] \,\&\, Rx\big)$
9. $\forall x\big[(Fx \,\&\, Vx) \to Sx\big] \,\&\, \exists x\big[Hx \,\&\, \forall y(Hy \to x = y) \,\&\, Sx\big]$
10. $\exists x\big(Dx \,\&\, Px \,\&\, \forall y[(Dy \,\&\, Py) \to x = y] \,\&\, Sx\big) \to \exists x \forall y(Sx \leftrightarrow x = y)$
11. negação externa: $\neg\exists x\big[Hx \,\&\, \forall y(Hy \to x = y) \,\&\, Vx\big]$
    negação interna: $\exists x\big[Hx \,\&\, \forall y(Hy \to x = y) \,\&\, \neg Vx\big]$
12. negação externa: $\neg\exists x \exists z\big(Dx \,\&\, Px \,\&\, Hz \,\&\, \forall y[(Dy \,\&\, Py) \to x = y] \,\&\, \forall y[(Hy \to z = y) \,\&\, x = z]\big)$
    negação interna: $\exists x \exists z\big(Dx \,\&\, Px \,\&\, Hz \,\&\, \forall y[(Dy \,\&\, Py) \to x = y] \,\&\, \forall y[(Hy \to z = y) \,\&\, x \neq z]\big)$

Capítulo 5 Parte A 2, 3, 4, 6, 8, e 9 são verdadeiras no modelo.

Capítulo 5 Parte B 4, 5, e 7 são verdadeiras no modelo.

Capítulo 5 Parte D

$$
\begin{aligned}
\text{UD} &= \{10,11,12,13\}\\
\text{extensão}(I) &= \{11,13\}\\
\text{extensão}(S) &= \emptyset\\
\text{extensão}(D) &= \{10,11,12,13\}\\
\text{extensão}(M) &= \{13\}\\
\text{extensão}(P) &= \{<11,10>,<12,11>,<13,12>\}
\end{aligned}
$$

Capítulo 5 Parte E

1. A sentença é verdadeira no seguinte modelo:

$$
\begin{aligned}
\text{UD} &= \{\text{Stan}\}\\
\text{extensão}(D) &= \{\text{Stan}\}\\
\text{referente}(a) &= \text{Stan}\\
\text{referente}(b) &= \text{Stan}
\end{aligned}
$$

   E é falsa no seguinte modelo:

$$\text{UD} = \{\text{Stan}\}$$
$$\text{extensão}(D) = \emptyset$$
$$\text{referente}(a) = \text{Stan}$$
$$\text{referente}(b) = \text{Stan}$$

2. A sentença é verdadeira no seguinte modelo:

$$\text{UD} = \{\text{Stan}\}$$
$$\text{extensão}(T) = \{<\text{Stan, Stan}>\}$$
$$\text{referente}(h) = \text{Stan}$$

   E é falsa no seguinte modelo:

$$\text{UD} = \{\text{Stan}\}$$
$$\text{extensão}(T) = \emptyset$$
$$\text{referente}(h) = \text{Stan}$$

3. A sentença é verdadeira no seguinte modelo:

$$\text{UD} = \{\text{Stan, Ollie}\}$$
$$\text{extensão}(P) = \{\text{Stan}\}$$
$$\text{referente}(m) = \text{Stan}$$

   E é falsa no seguinte modelo:

$$\text{UD} = \{\text{Stan}\}$$
$$\text{extensão}(P) = \emptyset$$
$$\text{referente}(m) = \text{Stan}$$

Capítulo 5 Parte F Existem muitas respostas corretas possíveis. Seguem-se algumas:

1. Fazendo a primeira sentença verdadeira e a segunda falsa:

$$\text{UD} = \{\text{alpha}\}$$
$$\text{extensão}(J) = \{\text{alpha}\}$$
$$\text{extensão}(K) = \emptyset$$
$$\text{referente}(a) = \text{alpha}$$

2. Fazendo a primeira sentença verdadeira e a segunda falsa:

$$\text{UD} = \{\text{alpha, omega}\}$$
$$\text{extensão}(J) = \{\text{alpha}\}$$
$$\text{referente}(m) = \text{omega}$$

3. Fazendo a primeira sentença falsa e a segunda verdadeira:

$$\text{UD} = \{\text{alpha, omega}\}$$
$$\text{extensão}(R) = \{<\text{alpha,alpha}>\}$$

4. Fazendo a primeira sentença falsa e a segunda verdadeira :

$$\begin{aligned}\text{UD} &= \{\text{alpha, omega}\}\\ \text{extensão}(P) &= \{\text{alpha}\}\\ \text{extensão}(Q) &= \emptyset\\ \text{referente}(c) &= \text{alpha}\end{aligned}$$

5. Fazendo a primeira sentença verdadeira e a segunda falsa:
$$\begin{aligned}\text{UD} &= \{\text{iota}\}\\ \text{extensão}(P) &= \emptyset\\ \text{extensão}(Q) &= \emptyset\end{aligned}$$
6. Fazendo a primeira sentença falsa e a segunda verdadeira:
$$\begin{aligned}\text{UD} &= \{\text{iota}\}\\ \text{extensão}(P) &= \emptyset\\ \text{extensão}(Q) &= \{\text{iota}\}\end{aligned}$$
7. Fazendo a primeira sentença verdadeira e a segunda falsa:
$$\begin{aligned}\text{UD} &= \{\text{iota}\}\\ \text{extensão}(P) &= \emptyset\\ \text{extensão}(Q) &= \{\text{iota}\}\end{aligned}$$
8. Fazendo a primeira sentença verdadeira e a segunda falsa:
$$\begin{aligned}\text{UD} &= \{\text{alpha, omega}\}\\ \text{extensão}(R) &= \{<\text{alpha, omega}>, <\text{omega, alpha}>\}\end{aligned}$$
9. Fazendo a primeira sentença falsa e a segunda verdadeira:
$$\begin{aligned}\text{UD} &= \{\text{alpha, omega}\}\\ \text{extensão}(R) &= \{<\text{alpha, alpha}>, <\text{alpha, omega}>\}\end{aligned}$$

Capítulo 5 Parte I

1. Existem muitas respostas possíveis. Segue-se uma:
$$\begin{aligned}\text{UD} &= \{\text{Harry, Sally}\}\\ \text{extensão}(R) &= \{<\text{Sally, Harry}>\}\\ \text{referente}(a) &= \text{Harry}\end{aligned}$$
2. Não há predicados ou constantes, então precisamos apenas fornecer um UD. Qualquer UD com 2 membros será suficiente.
3. Precisamos mostrar que é impossível construir um modeol no qual essas sentenças são ambas verdadeiras. Suponha que $\exists x\, x \neq a$ é verdadeira em um modelo. Há algo no universo de discurso que *não* é o referente de a. Portanto, há pelo menos duas coisas no universo de discurso: referente(a) e essa outra coisa. Chamemos essa outra de β — sabemos que $a \neq \beta$. Mas se $a \neq \beta$, então $\forall x \forall y\, x = y$ é falsa. Logo, a primeira sentença deve ser falsa se a segunda sentença for verdadeira. Consequentemente, não

há modelo algum no qual elas sejam ambas verdadeiras. Por fim, elas são inconsistentes.

Capítulo 5 Parte J

2. Não, não faria diferença alguma. A satisfação de uma fórmula com uma ou mais variáveis livres depende do que a atribuição de variável determina para aquelas variáveis. Porque uma sentença não tem variáveis livres, entretanto, sua satisfação não depende da atribuição de variáveis. Então, uma sentença que é satisfeita por *alguma* atribuição de variável é também satisfeita por *qualquer* outra atribuição de variável.

Capítulo 6 Parte A

1	$W \to \neg B$	
2	$A \,\&\, W$	
3	$B \lor (J \,\&\, K)$	
4	W	$\&$E 2
5	$\neg B$	$\to$E 1, 4
6	$J \,\&\, K$	$\lor$E 3, 5
7	K	$\&$E 6

1	$L \leftrightarrow \neg O$	
2	$L \lor \neg O$	
3	$\quad \neg L$	
4	$\quad \neg O$	$\lor$E 2, 3
5	$\quad L$	$\leftrightarrow$E 1, 4
6	$\quad \neg L$	R 3
7	L	$\neg$E 3–6

1	$Z \to (C \,\&\, \neg N)$	
2	$\neg Z \to (N \,\&\, \neg C)$	
3	$\quad \neg(N \lor C)$	
4	$\quad \neg N \,\&\, \neg C$	DeM 3
5	$\quad\quad Z$	
6	$\quad\quad C \,\&\, \neg N$	$\to$E 1, 5
7	$\quad\quad C$	$\&$E 6
8	$\quad\quad \neg C$	$\&$E 4
9	$\quad \neg Z$	$\neg$I 5–8
10	$\quad N \,\&\, \neg C$	$\to$E 2, 9
11	$\quad N$	$\&$E 10
12	$\quad \neg N$	$\&$E 4
13	$N \lor C$	$\neg$E 3–12

Capítulo 6 Parte B

1.

1	$K \mathbin{\&} L$	Busco $K \leftrightarrow L$
2	$\quad K$	Busco L
3	$\quad L$	$\&$E 1
4	$\quad L$	Busco K
5	$\quad K$	$\&$E 1
6	$K \leftrightarrow L$	$\leftrightarrow$I 2–3, 4–5

2.

1	$A \to (B \to C)$	Busco $(A \mathbin{\&} B) \to C$
2	$\quad A \mathbin{\&} B$	Busco C
3	$\quad A$	$\&$E 2
4	$\quad B \to C$	$\to$E 1, 3
5	$\quad B$	$\&$E 2
6	$\quad C$	$\to$E 4, 5
7	$(A \mathbin{\&} B) \to C$	$\to$I 2–6

3.

1	$P \mathbin{\&} (Q \vee R)$	
2	$P \to \neg R$	Busco $Q \vee E$
3	P	$\&$E 1
4	$\neg R$	$\to$E 2, 3
5	$Q \vee R$	$\&$E 1
6	Q	$\vee$E 5, 4
7	$Q \vee E$	$\vee$I 6

4.

1	$(C \mathbin{\&} D) \vee E$	Busco $E \vee D$
2	$\quad \neg E$	Busco D
3	$\quad C \mathbin{\&} D$	$\vee$E 1, 2
4	$\quad D$	$\&$E 3
5	$\neg E \to D$	$\to$I 2–4
6	$E \vee D$	CM 5

5.

1	$\neg F \to G$	
2	$F \to H$	Busco $G \vee H$
3	$\quad \neg G$	Busco H
4	$\quad \neg\neg F$	MT 1, 3
5	$\quad F$	DN 4
6	$\quad H$	$\to$E 2, 5
7	$\neg G \to H$	$\to$I 3–6
8	$G \vee H$	CM 7

6.

1	$(X \,\&\, Y) \vee (X \,\&\, Z)$	
2	$\neg(X \,\&\, D)$	
3	$D \vee M$	Busco M
4	$\quad \neg X$	por redução
5	$\quad \neg X \vee \neg Y$	$\vee$I 4
6	$\quad \neg(X \,\&\, Y)$	DeM 5
7	$\quad X \,\&\, Z$	$\vee$E 1, 6
8	$\quad X$	&E 7
9	$\quad \neg X$	R 4
10	X	$\neg$E 4–9
11	$\quad \neg M$	por redução
12	$\quad D$	$\vee$E 3, 11
13	$\quad X \,\&\, D$	&I 10, 12
14	$\quad \neg(X \,\&\, D)$	R 2
15	M	$\neg$E 11–14

Capítulo 6 Parte H

1	$\forall x \exists y(Rxy \vee Ryx)$	
2	$\forall x \neg Rmx$	
3	$\exists y(Rmy \vee Rym)$	$\forall$E 1
4	$\quad Rma \vee Ram$	
5	$\quad \neg Rma$	$\forall$E 2
6	$\quad Ram$	$\vee$E 4, 5
7	$\quad \exists x Rxm$	$\exists$I 6
8	$\exists x Rxm$	$\exists$E 3, 4–7

1	$\forall x(\exists yLxy \to \forall zLzx)$	
2	Lab	
3	$\exists yLay \to \forall zLza$	$\forall$E 1
4	$\exists yLay$	$\exists$I 2
5	$\forall zLza$	$\to$E 3, 4
6	Lca	$\forall$E 5
7	$\exists yLcy \to \forall zLzc$	$\forall$E 1
8	$\exists yLcy$	$\exists$I 6
9	$\forall zLzc$	$\to$E 7, 8
10	Lcc	$\forall$E 9
11	$\forall xLxx$	$\forall$I 10

1	$\forall x(Jx \to Kx)$	
2	$\exists x\forall yLxy$	
3	$\forall xJx$	
4	$\quad \forall yLay$	
5	$\quad Ja$	$\forall$E 3
6	$\quad Ja \to Ka$	$\forall$E 1
7	$\quad Ka$	$\to$E 6, 5
8	$\quad Laa$	$\forall$E 4
9	$\quad Ka \,\&\, Laa$	&I 7, 8
10	$\quad \exists x(Kx \,\&\, Lxx)$	$\exists$I 9
11	$\exists x(Kx \,\&\, Lxx)$	$\exists$E 2, 4–10

1	$\quad \neg(\exists xMx \lor \forall x\neg Mx)$	
2	$\quad \neg\exists xMx \,\&\, \neg\forall x\neg Mx$	DeM 1
3	$\quad \neg\exists xMx$	&E 2
4	$\quad \forall x\neg Mx$	NQ 3
5	$\quad \neg\forall x\neg Mx$	&E 2
6	$\exists xMx \lor \forall x\neg Mx$	$\neg$E 1–5

Capítulo 6 Parte I

1.

1	$\quad \neg(\forall xFx \lor \neg\forall xFx)$	por redução
2	$\quad \neg\forall xFx \,\&\, \neg\neg\forall xFx$	DeM 1
3	$\quad \neg\forall xFx$	&E 2
4	$\quad \neg\neg\forall xFx$	&E 2
5	$\forall xFx \lor \neg\forall xFx$	$\neg$E 1–4

2.

1	$\forall x(Mx \leftrightarrow Nx)$	
2	$Ma \,\&\, \exists x Rxa$	busca-se $\exists x Nx$
3	$Ma \leftrightarrow Na$	$\forall$E 1
4	Ma	& E 2
5	Na	$\leftrightarrow$E 3, 4
6	$\exists x Nx$	$\exists$I 5

3.

1	$\forall x(\neg Mx \vee Ljx)$	
2	$\forall x(Bx \rightarrow Ljx)$	
3	$\forall x(Mx \vee Bx)$	busca-se $\forall x Ljx$
4	$\neg Ma \vee Lja$	$\forall$E 1
5	$Ma \rightarrow Lja$	CM 4
6	$Ba \rightarrow Lja$	$\forall$E 2
7	$Ma \vee Ba$	$\forall$E 3
8	Lja	DIL 7, 5, 6
9	$\forall x Ljx$	$\forall$I 8

4.

1	$\forall x(Cx \,\&\, Dt)$	Busco $\forall x Cx \,\&\, Dt$
2	$Ca \,\&\, Dt$	$\forall$E 1
3	Ca	& E 2
4	$\forall x Cx$	$\forall$I 3
5	Dt	& E 2
6	$\forall x Cx \,\&\, Dt$	& I 4, 5

5.

1	$\exists x(Cx \vee Dt)$	Busco $\exists xCx \vee Dt$
2	$Ca \vee Dt$	por $\exists$E
3	$\neg(\exists xCx \vee Dt)$	por redução
4	$\neg\exists xCx \,\&\, \neg Dt$	DeM 3
5	$\neg Dt$	&E 4
6	Ca	$\vee$E 2, 5
7	$\exists xCx$	$\exists$I 6
8	$\neg\exists xCx$	&E 4
9	$\exists xCx \vee Dt$	$\neg$E 3–8
10	$\exists xCx \vee Dt$	$\exists$E 1, 2–9

Capítulo 6 Parte P Quanto à tradução deste argumento, *cf.* p. 81.

1	$\exists x\forall y[\forall z(Lxz \to Lyz) \to Lxy]$	
2	$\forall y[\forall z(Laz \to Lyz) \to Lay]$	
3	$\forall z(Laz \to Laz) \to Laa$	$\forall$E 2
4	$\neg\exists xLxx$	por redução
5	$\forall x\neg Lxx$	NQ 4
6	$\neg Laa$	$\forall$E 5
7	$\neg\forall z(Laz \to Laz)$	MT 5, 6
8	Lab	
9	Lab	R 8
10	$Lab \to Lab$	$\to$I 8—9
11	$\forall z(Laz \to Laz)$	$\forall$I 10
12	$\neg\forall z(Laz \to Laz)$	R 7
13	$\exists xLxx$	$\neg$E 4—12
14	$\exists xLxx$	$\exists$E 1, 2—13

Capítulo 6 Parte S 2, 3, e 5 são logicamente equivalentes.

Capítulo 6 Parte T 2, 4, 5, 7, e 10 são válidos. Seguem as respostas completas para alguns deles:

1. $$\text{UD} = \{\text{mocha, freddo}\}$$
   $$\text{extensão}(R) = \{<\text{mocha, freddo}>, <\text{freddo, mocha}>\}$$

2.

1	$\exists y \forall x Rxy$	Busco $\forall x \exists y Rxy$
2	$\quad \forall x Rxa$	
3	$\quad Rba$	$\forall$E 2
4	$\quad \exists y Rby$	$\exists$I 3
5	$\quad \forall x \exists y Rxy$	$\forall$I 4
6	$\forall x \exists y Rxy$	$\exists$E 1, 2–5

Apêndice C

Referência rápida

Tabelas de verdade

$\mathcal{A}$	$\neg\mathcal{A}$
V	F
F	V

$\mathcal{A}$	$\mathcal{B}$	$\mathcal{A}\&\mathcal{B}$	$\mathcal{A}\vee\mathcal{B}$	$\mathcal{A}\to\mathcal{B}$	$\mathcal{A}\leftrightarrow\mathcal{B}$
V	V	V	V	V	V
V	F	F	V	F	F
F	V	F	V	V	F
F	F	F	F	V	V

Simbolização

CONECTIVOS SENTENCIAS (capítulo 2)

Não é o caso que A.	$\neg P$
P ou Q	$(P \vee Q)$
Nem P, nem Q.	$\neg(P \vee Q)$ ou $(\neg P \,\&\, \neg Q)$
P e Q.	$(P \,\&\, Q)$
Se P, então Q.	$(P \rightarrow Q)$
P somente se Q.	$(P \rightarrow Q)$
P se e somente se Q.	$(P \leftrightarrow Q)$
A menos que P, Q. P a menos que Q.	$(P \vee Q)$

PREDICADOS (capítulo 4)

Todos Fs são Gs.	$\forall x(Fx \rightarrow Gx)$
Alguns Fs são Gs.	$\exists x(Fx \,\&\, Gx)$
Nem todos Fs são Gs.	$\neg\forall x(Fx \rightarrow Gx)$ ou $\exists x(Fx \,\&\, \neg Gx)$
Nenhum F é G.	$\forall x(Fx \rightarrow \neg Gx)$ ou $\neg\exists x(Fx \,\&\, Gx)$

IDENTIDADE (seção 4.6)

Somente j é G.	$\forall x(Gx \leftrightarrow x = j)$
Tudo menos j é G.	$\forall x(x \neq j \rightarrow Gx)$
O F é G.	$\exists x(Fx \,\&\, \forall y(Fy \rightarrow x = y) \,\&\, Gx)$
'O F não é G' pode ser traduzido de duas maneiras:	
Não é o caso que F é G. (negação externa)	$\neg\exists x(Fx \,\&\, \forall y(Fy \rightarrow x = y) \,\&\, Gx)$
O F é não-G. (negação interna)	$\exists x(Fx \,\&\, \forall y(Fy \rightarrow x = y) \,\&\, \neg Gx)$

Usando a identidade para simbolizar quantidades

Existe(m) pelo menos _____ F(s).

um $\exists x Fx$

dois $\exists x_1 \exists x_2(Fx_1 \,\&\, Fx_2 \,\&\, x_1 \neq x_2)$

três $\exists x_1 \exists x_2 \exists x_3(Fx_1 \,\&\, Fx_2 \,\&\, Fx_3 \,\&\, x_1 \neq x_2 \,\&\, x_1 \neq x_3 \,\&\, x_2 \neq x_3)$

quatro $\exists x_1 \exists x_2 \exists x_3 \exists x_4 (Fx_1 \,\&\, Fx_2 \,\&\, Fx_3 \,\&\, Fx_4 \,\&\, x_1 \neq x_2 \,\&\, x_1 \neq x_3 \,\&\, x_1 \neq x_4 \,\&\, x_2 \neq x_3 \,\&\, x_2 \neq x_4 \,\&\, x_3 \neq x_4)$

n $\exists x_1 \cdots \exists x_n (Fx_1 \,\&\, \cdots \,\&\, Fx_n \,\&\, x_1 \neq x_2 \,\&\, \cdots \,\&\, x_{n-1} \neq x_n)$

Existe(m) no máximo _____ F(s).

Uma maneira de dizer 'no máximo n coisas são F' é colocando um sinal de negação em frente de uma das simbolizações acima e dizer ¬'pelo menos n+1 coisas são Fs'. Equivalentemente:

um $\forall x_1 \forall x_2 \big[(Fx_1 \,\&\, Fx_2) \rightarrow x_1 = x_2\big]$

dois $\forall x_1 \forall x_2 \forall x_3 \big[(Fx_1 \,\&\, Fx_2 \,\&\, Fx_3) \rightarrow (x_1 = x_2 \vee x_1 = x_3 \vee x_2 = x_3)\big]$

três $\forall x_1 \forall x_2 \forall x_3 \forall x_4 \big[(Fx_1 \,\&\, Fx_2 \,\&\, Fx_3 \,\&\, Fx_4) \rightarrow (x_1 = x_2 \vee x_1 = x_3 \vee x_1 = x_4 \vee x_2 = x_3 \vee x_2 = x_4 \vee x_3 = x_4)\big]$

n $\forall x_1 \cdots \forall x_{n+1} \big[(Fx_1 \,\&\, \cdots \,\&\, Fx_{n+1}) \rightarrow (x_1 = x_2 \vee \cdots \vee x_n = x_{n+1})\big]$

Existe(m) exatamente _____ F(s).

Uma maneira de dizer 'exatamente n coisas são F' é conjuntando duas das simbolizações acima e dizer 'pelo menos n coisas são F' & 'no máximo n coisas são F'. As seguintes fórmulas equivalentes são mais curtas:

zero $\forall x \neg Fx$

um $\exists x \big[Fx \,\&\, \neg \exists y (Fy \,\&\, x \neq y)\big]$

dois $\exists x_1 \exists x_2 \big[Fx_1 \,\&\, Fx_2 \,\&\, x_1 \neq x_2 \,\&\, \neg \exists y \big(Fy \,\&\, y \neq x_1 \,\&\, y \neq x_2\big)\big]$

três $\exists x_1 \exists x_2 \exists x_3 \big[Fx_1 \,\&\, Fx_2 \,\&\, Fx_3 \,\&\, x_1 \neq x_2 \,\&\, x_1 \neq x_3 \,\&\, x_2 \neq x_3 \,\&\, \neg \exists y (Fy \,\&\, y \neq x_1 \,\&\, y \neq x_2 \,\&\, y \neq x_3)\big]$

n $\exists x_1 \cdots \exists x_n \big[Fx_1 \,\&\, \cdots \,\&\, Fx_n \,\&\, x_1 \neq x_2 \,\&\, \cdots \,\&\, x_{n-1} \neq x_n \,\&\, \neg \exists y (Fy \,\&\, y \neq x_1 \,\&\, \cdots \,\&\, y \neq x_n)\big]$

Especificando o tamanho do universo de discurso

Remover F das simbolizações acima produz sentenças que falam sobre o tamanho do UD. Por exemplo, 'existem pelo menos duas coisas (no UD)' pode ser simbolizada como $\exists x \exists y (x \neq y)$.

Regras básicas de prova

Reiteração

$$\begin{array}{r|ll} m & \mathcal{A} & \\ & \mathcal{A} & \text{R } m \end{array}$$

Introdução da conjunção

$$\begin{array}{r|ll} m & \mathcal{A} & \\ n & \mathcal{B} & \\ & \mathcal{A} \mathbin{\&} \mathcal{B} & \mathbin{\&}\text{I } m, n \end{array}$$

Eliminação da conjunção

$$\begin{array}{r|ll} m & \mathcal{A} \mathbin{\&} \mathcal{B} & \\ & \mathcal{A} & \mathbin{\&}\text{E } m \end{array}$$

$$\begin{array}{r|ll} m & \mathcal{A} \mathbin{\&} \mathcal{B} & \\ & \mathcal{B} & \mathbin{\&}\text{E } m \end{array}$$

Introdução da disjunção

$$\begin{array}{r|ll} m & \mathcal{A} & \\ & \mathcal{A} \vee \mathcal{B} & \vee\text{I } m \end{array}$$

$$\begin{array}{r|ll} m & \mathcal{A} & \\ & \mathcal{B} \vee \mathcal{A} & \vee\text{I } m \end{array}$$

Eliminação da disjunção

$$\begin{array}{r|ll} m & \mathcal{A} \vee \mathcal{B} & \\ n & \neg\mathcal{B} & \\ & \mathcal{A} & \vee\text{E } m, n \end{array}$$

$$\begin{array}{r|ll} m & \mathcal{A} \vee \mathcal{B} & \\ n & \neg\mathcal{A} & \\ & \mathcal{B} & \vee\text{E } m, n \end{array}$$

Introdução do condicional

$$\begin{array}{r|ll} \begin{array}{r} m \\ n \end{array} & \begin{array}{|l} \mathcal{A} \\ \hline \mathcal{B} \end{array} & \begin{array}{l} \text{busco } \mathcal{B} \\ \ \end{array} \\ & \mathcal{A} \rightarrow \mathcal{B} & \rightarrow\text{I } m\text{–}n \end{array}$$

Eliminação do condicional

$$\begin{array}{r|ll} m & \mathcal{A} \rightarrow \mathcal{B} & \\ n & \mathcal{A} & \\ & \mathcal{B} & \rightarrow\text{E } m, n \end{array}$$

Introdução do bicondicional

$$\begin{array}{r|ll} \begin{array}{r} m \\ n \end{array} & \begin{array}{|l} \mathcal{A} \\ \hline \mathcal{B} \end{array} & \begin{array}{l} \text{busco } \mathcal{B} \\ \ \end{array} \\ \begin{array}{r} p \\ q \end{array} & \begin{array}{|l} \mathcal{B} \\ \hline \mathcal{A} \end{array} & \begin{array}{l} \text{busco } \mathcal{A} \\ \ \end{array} \\ & \mathcal{A} \leftrightarrow \mathcal{B} & \leftrightarrow\text{I } m\text{–}n,\ p\text{–}q \end{array}$$

Eliminação do bicondicional

$$\begin{array}{r|ll} m & \mathcal{A} \leftrightarrow \mathcal{B} & \\ n & \mathcal{B} & \\ & \mathcal{A} & \leftrightarrow\text{E } m, n \end{array}$$

$$\begin{array}{r|ll} m & \mathcal{A} \leftrightarrow \mathcal{B} & \\ n & \mathcal{A} & \\ & \mathcal{B} & \leftrightarrow\text{E } m, n \end{array}$$

Introdução da negação

$$
\begin{array}{l|l|ll}
m & & \mathcal{A} & \text{por redução} \\ \cline{3-3}
n-1 & & \mathcal{B} & \\
n & & \neg\mathcal{B} & \\
 & \neg\mathcal{A} & & \neg\text{I } m\text{–}n
\end{array}
$$

Eliminação da negação

$$
\begin{array}{l|l|ll}
m & & \neg\mathcal{A} & \text{redução} \\ \cline{3-3}
n-1 & & \mathcal{B} & \\
n & & \neg\mathcal{B} & \\
 & \mathcal{A} & & \neg\text{E } m\text{–}n
\end{array}
$$

Regras dos quantificadores

Introdução do existencial

$$
\begin{array}{l|ll}
m & \mathcal{A}c & \\
 & \exists\chi\mathcal{A}\ \boxed{\boxed{\chi\Rightarrow c}} & \exists\text{I } m
\end{array}
$$

Note que χ pode substituir algumas ou todas ocorrências de c em $\mathcal{A}$.

Eliminação existencial

$$
\begin{array}{l|l|ll}
m & \exists\chi\mathcal{A}\chi & & \\
n & & \mathcal{A}\ \boxed{c* \Rightarrow\chi} & \\ \cline{3-3}
p & & \mathcal{B} & \\
 & \mathcal{B} & & \exists\text{E } m,\ n\text{–}p
\end{array}
$$

* c não deve ocorrer fora da subprova.

Introdução universal

$$
\begin{array}{l|ll}
m & \mathcal{A} & \\
 & \forall\chi\mathcal{A}\ \boxed{c* \Rightarrow\chi} & \forall\text{I } m
\end{array}
$$

* c não deve ocorrer em quaisquer hipóteses não descartadas.

Eliminação universal

$$
\begin{array}{l|ll}
m & \forall\chi\mathcal{A}\chi & \\
 & \mathcal{A}\ \boxed{\chi\Rightarrow c} & \forall\text{E } m
\end{array}
$$

Regras de identidade

$$
\begin{array}{|ll}
c = c & =\text{I}
\end{array}
$$

$$
\begin{array}{l|ll}
m & c = d & \\
n & \mathcal{A} & \\
 & \mathcal{A}\ \boxed{\boxed{c\Rightarrow d}} & =\text{E } m,\ n \\
 & \mathcal{A}\ \boxed{\boxed{d\Rightarrow c}} & =\text{E } m,\ n
\end{array}
$$

Uma constante pode substituir algumas ou todas as ocorrências da outra.

Regras derivadas

Dilema

m	$\mathcal{A} \vee \mathcal{B}$	
n	$\mathcal{A} \to \mathcal{C}$	
p	$\mathcal{B} \to \mathcal{C}$	
	$\mathcal{C}$	DIL m, n, p

Modus Tollens

m	$\mathcal{A} \to \mathcal{B}$	
n	$\neg\mathcal{B}$	
	$\neg\mathcal{A}$	MT m, n

Silogismo hipotético

m	$\mathcal{A} \to \mathcal{B}$	
n	$\mathcal{B} \to \mathcal{C}$	
	$\mathcal{A} \to \mathcal{C}$	SH m, n

Regras de substituição

Comutatividade (Com)

$$(\mathcal{A} \,\&\, \mathcal{B}) \Longleftrightarrow (\mathcal{B} \,\&\, \mathcal{A})$$
$$(\mathcal{A} \vee \mathcal{B}) \Longleftrightarrow (\mathcal{B} \vee \mathcal{A})$$
$$(\mathcal{A} \leftrightarrow \mathcal{B}) \Longleftrightarrow (\mathcal{B} \leftrightarrow \mathcal{A})$$

DeMorgan (DeM)

$$\neg(\mathcal{A} \vee \mathcal{B}) \Longleftrightarrow (\neg\mathcal{A} \,\&\, \neg\mathcal{B})$$
$$\neg(\mathcal{A} \,\&\, \mathcal{B}) \Longleftrightarrow (\neg\mathcal{A} \vee \neg\mathcal{B})$$

Dupla negação (DN)

$$\neg\neg\mathcal{A} \Longleftrightarrow \mathcal{A}$$

Condicional material (CM)

$$(\mathcal{A} \to \mathcal{B}) \Longleftrightarrow (\neg\mathcal{A} \vee \mathcal{B})$$
$$(\mathcal{A} \vee \mathcal{B}) \Longleftrightarrow (\neg\mathcal{A} \to \mathcal{B})$$

Troca de bicondicional($\leftrightarrow$c)

$$[(\mathcal{A} \to \mathcal{B}) \,\&\, (\mathcal{B} \to \mathcal{A})] \Longleftrightarrow (\mathcal{A} \leftrightarrow \mathcal{B})$$

Negação de quantificador (NQ)

$$\neg\forall\chi\mathcal{A} \Longleftrightarrow \exists\chi\neg\mathcal{A}$$
$$\neg\exists\chi\mathcal{A} \Longleftrightarrow \forall\chi\neg\mathcal{A}$$