CAPÍTULO 5

UM MÉTODO SINTÁTICO DE DEDUÇÃO NA LÓGICA PROPOSICIONAL

5.1 Introdução

Os métodos semânticos apresentados no Capítulo 4 resolvem bem o seguinte problema: dada uma fórmula H, eles decidem sobre suas propriedades semânticas. Isto é, tais métodos semânticos resolvem o problema de caracterizar as propriedades semânticas de uma fórmula da Lógica Proposicional. Por exemplo, para identificar as propriedades semânticas de uma fórmula H, basta construir uma tabela-verdade associada a H e analisar sua coluna. Desse ponto de vista, não haveria muito sentido em estudar outro tipo de método com objetivos semelhantes. Mas, mesmo assim, analisamos neste capítulo um método sintático, no qual um dos objetivos é caracterizar as tautologias. Mas, por que propor tal análise? Quando consideramos apenas a Lógica Proposicional, certamente não fica claro nossos objetivos futuros. Porém, vale a pena deixá-los claro. No contexto da Lógica de Predicados, que estudamos a partir do próximo capítulo, métodos semânticos e sintáticos são conceitualmente diferentes e não se equivalem. Isto significa que resultados obtidos, utilizando métodos semânticos, não correspondem a resultados obtidos, utilizando métodos sintáticos e vice-versa. A prova mais evidente disso é o famoso teorema de Gödel [Smith].

Neste capítulo é analisado um método sintático de dedução na Lógica Proposicional. Esse método define um sistema, também denominado sistema formal, que estabelece estruturas que permitem a dedução formal em um domínio sintático, no qual não é considerada a interpretação das fórmulas. Observe que isso difere dos métodos semânticos, nos quais sempre consideramos a interpretação dos símbolos das fórmulas. Além disso, é necessário enfatizar que o sistema formal proposto não é, certamente, o mais adequado para implementações em computadores. Em geral, os mais adequados utilizam métodos de dedução diferentes, como, por exemplo, *tableaux* semânticos. Entretanto, no sistema formal proposto, as noções de prova, derivação, consequência lógica e completude se apresentam de forma simples. E a definição das noções de prova e consequência lógica, por exemplo, é um tema central em Lógica. Por isso, existem excelentes referências que tratam de sistemas de dedução formal: [Andrews], [Mendelson], [Enderton], [Dalen], [Shoenfield] e [Fitting].

5.2 O sistema formal $\wp_a$

O sistema formal definido neste capítulo estabelece um método sintático de prova, que prova se uma fórmula H é uma tautologia[1]. E, para facilitar nossas referências ao sistema proposto, nós o denominamos pelo acrônimo $\wp_a$, que contém a letra $\wp$ e o índice "a", como uma simples referência, ou lembrança, à palavra "Proposicional". Para iniciar o estudo do sistema $\wp_a$, o primeiro passo é saber quais são seus elementos básicos.

Definição 5.1 (elementos básicos do sistema formal $\wp_a$) *O sistema formal $\wp_a$ da Lógica Proposicional é definido a partir de quatro conjuntos:*

1. *o alfabeto da Lógica Proposicional, Definição 3.14;*
2. *o conjunto das fórmulas da Lógica Proposicional;*
3. *um subconjunto das fórmulas especiais, que são denominadas axiomas;*
4. *um conjunto de regras de dedução.*

Portanto, a linguagem utilizada no sistema $\wp_a$ é a linguagem da Lógica Proposicional, com seu alfabeto e suas fórmulas. E para evitar dificuldades técnicas, admitimos no sistema $\wp_a$ denotar as fórmulas, utilizando todos os conectivos e não apenas $\neg$ e $\vee$, que pertencem ao alfabeto da Definição 3.14. Isto é, no contexto dos sistema $\wp_a$, podemos denotar as fórmulas conforme as correspondências.

Nota. N sistema $\wp_a$, consideramos as denotações:

1. $(H \wedge G)$ denota $\neg(\neg H \vee \neg G)$.
2. $H \rightarrow G$ denota $(\neg H \vee G)$.
3. $(H \leftrightarrow G)$ denota $(H \rightarrow G) \wedge (G \rightarrow H)$.

[1] O sistema estudado é o mesmo apresentado por P. B. Andrews em [Andrews].

Observe que não estamos falando em equivalências, como no Exemplo 3.22. Estamos relacionando diferentes denotações para a mesma fórmula. Isso deve ser feito dessa maneira, porque o sistema $\wp_a$ define um método sintático de derivação, no qual não se considera conceitos semânticos como o da equivalência. Um dos objetivos do estudo do sistema $\wp_a$ é o estudo formal da representação e dedução sintática de novas fórmulas. A dedução de uma nova fórmula é feita a partir de outras, dadas *a priori*. Essas fórmulas, fornecidas de antemão, são denominadas axiomas. Os axiomas são, portanto, fórmulas às quais atribuímos um status especial de verdade básica, ou *a priori*. O primeiro sistema axiomático foi organizado por Euclides, no qual ele estudou a geometria. Em seu sistema, um dos axiomas é a sentença:

"Dados dois pontos em um plano, é possível traçar uma linha reta passando pelos dois pontos."

Observe que, na geometria de Euclides, o conhecimento determinado por esse axioma é dado *a priori*, não sendo demonstrado. Esse axioma representa um conhecimento fundamental de sua geometria. Para Euclides, um axioma era uma proposição verdadeira, autoevidente, e não era preciso demonstrá-la. Portanto, no sistema $\wp_a$, um axioma é simplesmente uma fórmula que representa uma proposição aceita sem prova sintática.[2] Além disso, no sistema $\wp_a$, os axiomas são fórmulas definidas na linguagem formal da Lógica Proposicional. Já na geometria de Euclides, os axiomas são proposições construídas utilizando linguagem natural. Por isso, a geometria de Euclides é um sistema axiomático, e $\wp_a$ é um sistema formal axiomático. Definimos a seguir o conjunto especial de fórmulas, denominadas axiomas de $\wp_a$.

Definição 5.2 (axiomas do sistema $\wp_a$) *Os axiomas do sistema $\wp_a$ são fórmulas da Lógica Proposicional determinadas pelos esquemas indicados a seguir. Nesses esquemas E, G e H são fórmulas quaisquer da Lógica Proposicional.*

1. $Ax_l = (H \lor H) \rightarrow H$;
2. $Ax_2 = H \rightarrow (G \lor H)$;
3. $Ax_3 = (H \rightarrow G) \rightarrow ((E \lor H) \rightarrow (G \lor E))$.

Nota. Os axiomas do sistema formal $\wp_a$ são denominados axiomas lógicos. Isso porque há também os axiomas não lógicos utilizados no estudo de outros sistemas lógicos como é visto mais adiante.

Observe que na Definição 5.2 consideramos esquemas de fórmulas que denotam infinitas fórmulas. Isso significa que Ax_1, por exemplo, pode denotar qualquer uma das fórmulas: $(P \lor P) \rightarrow P$, $(Q \lor Q) \rightarrow Q$, $((P \rightarrow Q) \lor (P \rightarrow Q)) \rightarrow (P \rightarrow Q)$, e $((P \lor Q) \lor (P \lor Q)) \rightarrow (P \lor Q)$. Observe, ainda, que o axioma Ax_2 também pode ser denotado como: $Ax_2 = H \rightarrow (E \rightarrow H)$, que é obtido fazendo $G = \neg E$. No sistema $\wp_a$, os esquemas de fórmulas que definem os axiomas representam infinitas fórmulas. Por isso, o sistema $\wp_a$ possui infinitos axiomas. Entretanto, apesar de

[2] A definição de prova sintática será apresentada adiante.

possuir, rigorosamente, infinitos axiomas, os sistema $\wp_a$ possui um número finito de esquemas de axiomas: Ax_1, Ax_2 e Ax_3.

Nota. Sistemas formais que possuem um número finito de esquemas de axiomas são sistemas finitamente axiomatizáveis. Esse é o caso do sistema P_a.

Observe que o conjunto dos axiomas de $\wp_a$ é decidível, ou seja, temos um algoritmo que decide que fórmulas da Lógica Proposicional corresponde, ou não, a um axioma. Além disso, esse conjunto é enumerável. Isto é, há uma forma de enumerar todos os axiomas de $\wp_a$. Finalmente, observe que o sistema $\wp_a$ é definido a partir de uma linguagem que possui apenas símbolos proposicionais e não contém os símbolos de verdade $true$, ou $false$. Isso é importante porque alguns resultados apresentados neste capítulo[3] não são válidos, caso a linguagem de $\wp_a$ contenha algum símbolo de verdade. Os axiomas de um sistema axiomático representam o conhecimento dado a priori. No caso do sistema $\wp_a$, esse conhecimento é representado por axiomas que são tautologias. A demonstração de que os axiomas Ax_1 e Ax_2 são tautologias é imediata. Já a demonstração de Ax_3 pode ser feita utilizando o método da negação.

A regra de inferência. No sistema $\wp_a$, o mecanismo de inferência, que permite a dedução de novas fórmulas, se fundamenta em um esquema de regra de inferência denominado *modus ponens.*[4] O sistema $\wp_a$ possui apenas uma regra de inferência, determinada pelo esquema *modus ponens.*

Definição 5.3 (regra de inferência do sistema $\wp_a$, *modus ponens*) *Dadas as fórmulas H e G, a regra de inferência do sistema $\wp_a$, denominada modus ponens (MP), é definida pelo procedimento: tendo H e $(H \rightarrow G)$ deduza G.*

De forma análoga aos axiomas, a regra de inferência *modus ponens* é um postulado e, portanto, aceito sem demonstração. Logo, no sistema formal $\wp_a$, se H e $(H \rightarrow G)$ são fórmulas já deduzidas, então G também é deduzida. Além disso, observe também um fato importante. Dada uma interpretação I, se $I[H] = T$ e $I[(H \rightarrow G)] = T$, então $I[G] = T$. Isto é, as fórmulas H e $(H \rightarrow G)$ implicam G. Isso significa que a regra *modus ponens* é correta, ou seja, ela deduz uma fórmula verdadeira, a partir de outras verdadeiras. Em outras palavras, a regra de inferência nos permite inferir uma fórmula verdadeira, a partir de outras fórmulas verdadeiras e que já foram deduzidas anteriormente.

Notação. Para representar o esquema de regra de inferência *modus ponens*, a notação a seguir é considerada:

$$MP = \frac{H, (H \rightarrow G)}{G}.$$

Nessa notação, o "numerador" da equação, $H, (H \rightarrow G)$, é denominado antecedente. O "denominador" é o consequente.

[3] O teorema da completude, por exemplo.

[4] Esta é uma forma reduzida da expressão, em latim, *"modus ponendo ponens",* que significa, literalmente, *"modo que, afirmando, afirma",* ou seja, o argumento que extrai uma conclusão afirmativa a partir de uma premissa condicional ou afirmativa.

Na regra de inferência *modus ponens*, as fórmulas H e G podem ser quaisquer. Isso significa que *modus ponens* denota um conjunto infinito de procedimentos de dedução. Como as fórmulas H e G são arbitrárias, a regra é, na verdade, um esquema de regra e determina, por exemplo, várias deduções, como as seguintes:

1. a partir de P e $(P \to Q)$, deduza Q;
2. a partir de $P \to Q$ e $((P \to Q) \to R)$, deduza R;
3. a partir de P e $(P \to (Q \vee R))$, deduza $(Q \vee R)$;
4. a partir de $(P \wedge Q)$ e $((P \wedge Q) \to (Q \to R))$, deduza $(Q \to R)$.

Suponha, por exemplo, que se tenha as representações P = "está chovendo" e Q = "a rua está molhada". Nesse caso, se P é deduzida, isto é, "está chovendo" e a implicação $(P \to Q)$ também é verdadeira, isto é, "se está chovendo então a rua está molhada", então, a partir da regra de inferência *modus ponens*, deduzimos Q, isto é, "a rua está molhada". Observe que somente é possível deduzir que "a rua está molhada" devido à conjunção de dois fatos: a dedução inicial de que "está chovendo" e também a dedução de "se está chovendo então a rua está molhada". Se apenas a implicação fosse deduzida, não se poderia concluir que "a rua está molhada". Assim, quando se deduz que $(P \to Q)$, nada se pode concluir Q. Até mosquito pensa dessa forma e tem essa capacidade de raciocínio.

5.3 Consequência lógica sintática no sistema $\wp_a$

O sistema axiomático $\wp_a$ define uma estrutura para representação e dedução sintática. Inicialmente, temos os axiomas e mais algumas fórmulas extras, denominadas hipóteses. A regra de inferência determina um mecanismo sintático de inferência que é aplicado aos axiomas e às hipóteses. Nesse sentido, fórmulas são deduzidas a partir da aplicação da regra de inferência ao que é dado *a priori*, no sistema. As fórmulas deduzidas formam, por sua vez, um conjunto denominado consequências lógicas sintáticas. As consequências lógicas sintáticas representam o que é provado, a partir dos axiomas e hipóteses, utilizando a regra de inferência. A prova de uma fórmula no sistema $\wp_a$ e a consequência lógica sintática são definidas a seguir.

Definição 5.4 (prova sintática no sistema $\wp_a$) *Sejam, na Lógica Proposicional, H uma fórmula e β um conjunto de fórmulas denominadas por hipóteses. Uma prova sintática de H a partir de β, no sistema axiomático $\wp_a$, é uma sequência de fórmulas $H_1, H_2, \ldots, H_n$, onde temos:*

1. $H = H_n$;

E para todo i tal que $1 \leq i \leq n$:

2. H_i *é um axioma; ou*
3. $H_i \in \beta$; *ou*

4. H_i é deduzida de H_j e H_k, utilizando a regra modus ponens, onde $1 \leq j < i$ e $1 \leq k < i$. Isto é:

$$MP = \frac{H_j \quad H_k}{H_i}$$

Observe que, neste caso, necessariamente, $H_k = H_j \to H_i$.

Exemplo 5.1 (prova no sistema $\wp_a$) Consideramos neste exemplo uma prova de $((Q \vee P) \vee \neg P)$ em $\wp_a$, na qual o conjunto de hipóteses β é dado pela fórmula $(\neg P \vee P)$. Conforme a Definição 5.4, esta prova é dada por uma sequência de fórmulas $H_1, H_2, \ldots, H_n$. O que fazemos a seguir é, simplesmente, determinar quem são essas fórmulas da sequência que define a prova.

A primeira fórmula da prova. A primeira fórmula da prova, H_1, é obtida do esquema que define o axioma Ax_3, fazendo: $H = P$, $G = (Q \vee P)$ e $E = \neg P$. Portanto, fazendo essas substituições, temos:

1. $H_1 = (P \to (Q \vee P)) \to ((\neg P \vee P) \to ((Q \vee P) \vee \neg P))$

A segunda fórmula da prova. A segunda fórmula, H_2, é obtida a partir do axioma Ax_2, fazendo: $H = P$, e $G = Q$. Portanto:

2. $H_2 = P \to (Q \vee P)$

A terceira fórmula da prova. Aplicando a regra *modus ponens* às fórmulas H_1 e H_2, obtemos a terceira fórmula da prova. Nesse caso, temos o resultado H_3, dado por:

3. $H_3 = (\neg P \vee P) \to ((Q \vee P) \vee \neg P)$

A quarta fórmula da prova. A quarta fórmula da prova é a hipótese do conjunto β. Isto é:

4. $H_4 = (\neg P \vee P)$

A quinta fórmula da prova. Aplicando a regra *modus ponens* às fórmulas H_3 e H_4 temos H_5, que é a quinta fórmula da prova.

5. $H_5 = (Q \vee P) \vee \neg P$

Conclusão: Temos uma prova de $(Q \vee P) \vee \neg P$ a partir dos axiomas de $\wp_a$ e da hipótese $\{\neg P \vee P\}$. Nesse caso, a sequência de fórmulas H_1, H_2, H_3, H_4, H_5 é a prova de $(Q \vee P) \vee \neg P$. ■

Apresentamos a seguir outro exemplo de prova. Mas, a linguagem utilizada é mais concisa e as fórmulas da sequência que define a prova são indicadas pelos subescritos dessas fórmulas.

Exemplo 5.2 (prova no sistema $\wp_a$) Considere o conjunto de hipóteses $\beta = \{G_1, \ldots, G_9\}$, tal que $G_1 = (P \wedge R) \to P$, $G_2 = Q \to P_4$, $G_3 = P_1 \to Q$, $G_4 = (P_1 \wedge P_2) \to Q$, $G_5 = (P_3 \wedge R) \to R$, $G_6 = P_4 \to P$, $G_7 = P_1$, $G_8 = P_3 \to P$, $G_9 = P_2$. A prova apresentada a seguir tem, portanto, como conhecimento a priori, nove fórmulas. E como o método de prova é sintático, não se considera o significado ou interpretação dessas fórmulas. A sequência de fórmulas $H_1, \ldots, H_9$ é uma prova de $(S \vee P)$ a partir de β no sistema axiomático $\wp_a$.

1. $H_1 = G_7$, ou seja: $H_1 = P_1$;
2. $H_2 = G_3$, ou seja: $H_2 = P_1 \rightarrow Q$;
3. $H_3 = Q$ (resultado de MP em H_1 e H_2);
4. $H_4 = G_2$, ou seja: $H_4 = Q \rightarrow P_4$;
5. $H_5 = P_4$ (resultado de MP em H_3 e H_4);
6. $H_6 = G_6$, ou seja: $H_6 = P_4 \rightarrow P$;
7. $H_7 = P$ (resultado de MP em H_5 e H_6);
8. $H_8 = Ax_2$, ou seja: $H_8 = P \rightarrow (S \vee P)$;
9. $H_9 = (S \vee P)$ (resultado de MP em H_7 e H_8).

Nessa sequência, as duas primeiras fórmulas correspondem a fórmulas do conjunto de hipóteses. Isto é, dado que são hipóteses, elas são admitidas na sequência da prova, pois nosso objetivo é provar $(S \vee P)$ a partir de β. Utilizando as duas primeiras fórmulas e a regra *modus ponens*, derivamos $H_3 = Q$. Nesse caso, a regra é utilizada na seguinte forma:

$$MP = \frac{P_1,\ (P_1 \rightarrow Q)}{Q}.$$

De forma análoga, as fórmulas H_5 e H_7 são deduzidas, utilizando *modus ponens*. A dedução de H_5 é a seguinte:

$$MP = \frac{P_1,\ (P_1 \rightarrow P_4)}{P_4}.$$

E a dedução de H_7 é a seguinte:

$$MP = \frac{P_4,\ (P_4 \rightarrow P)}{P}.$$

Além disso, $H_8 = Ax_2$, isto é:

$$H_8 = (P \rightarrow (S \vee P)).$$

Isso significa que obtemos que H_8 é um caso particular do axioma Ax_2. Nesse caso, H_8 é obtida a partir desse axioma, considerando $H = P$ e $G = S$. Logo:

$$H_8 = Ax_2 = (P \rightarrow (S \vee P)).$$

Finalmente, dado que $H_9 = (S \vee P)$, então temos uma prova de H_9 a partir das hipóteses em β e, é claro, dos axiomas. Uma notação ainda mais concisa dessa prova é aquelas na qual apresentamos apenas as aplicações da regra *modus ponens*. Nesse caso, a prova é representada conforme o esquema da Figura 5.1, a seguir.

$$
\begin{array}{l}
\underline{P_1, (P_1 \to Q)}\ldots MP_1 \\
\qquad\quad \underline{Q, \qquad (Q \to P_4)}\ldots MP_2 \\
\qquad\qquad\qquad\qquad\quad \underline{P_4, \qquad (P_4 \to P)}\ldots MP_3 \\
\qquad\qquad\qquad\qquad\qquad\qquad\qquad\quad \underline{P, \qquad (P \to (S \vee P))}\ldots MP_4 \\
\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad\quad (S \vee P)
\end{array}
$$

Figura 5.1: Prova de $(S \vee P)$.

Nesse esquema, a regra *modus ponens* é utilizada quatro vezes, o que é indicado por MP_1, MP_2, MP_3 e MP_4. Na aplicação de MP_1, as fórmulas do antecedente são P_1 e $(P_1 \to Q)$. Observe que tais fórmulas pertencem ao conjunto de hipóteses β. Na segunda aplicação da regra, o antecedente é formado por Q e $(Q \to P_4)$. Nesse caso, a fórmula Q é o consequente ou resultado da primeira aplicação da regra *modus ponens*, e a fórmula $(Q \to P_4)$ pertence ao conjunto β. As outras aplicações de *modus ponens* são análogas, exceto a última, em que o antecedente da regra é composto pelas fórmulas P e $(P \to (S \vee P))$. Nesse caso, P é o resultado da terceira aplicação da regra de inferência e a fórmula $(P \to (S \vee P))$ corresponde a um caso particular do axioma Ax_2, onde $H = P$ e $G = S$. ■

Dada uma fórmula H e um conjunto de hipóteses β, suponha que H tenha uma prova a partir de β. Isso significa que H é deduzida sintaticamente, utilizando *modus ponens*, a partir dos axiomas e das fórmulas do conjunto β. No caso em que o conjunto β é vazio, H é deduzida utilizando somente os axiomas e *modus ponens*.

Definição 5.5 (consequência lógica sintática no sistema $\wp_a$) *Dados, na Lógica Proposicional, uma fórmula H e um conjunto de hipóteses β*[5]*, então H é uma consequência lógica sintática de β em $\wp_a$, se existe uma prova de H a partir de β.*

Essa consequência lógica, expressa na Definição 5.5, é sintática porque todos os conceitos envolvidos são sintáticos. Não são considerados conceitos semânticos que tratam da interpretação das fórmulas. Entretanto, devemos lembrar que um dos principais objetivos da Lógica é relacionar conceitos lógicos sintáticos e semânticos. E nesse sentido, há extensos estudos que relacionam consequência lógica sintática e consequência lógica semântica. Por exemplo, os teoremas da correção e da completude no sistema $\wp_a$, vistos mais adiante, tratam desse tema.

Definição 5.6 (teorema no sistema $\wp_a$) *Uma fórmula H é um teorema em $\wp_a$, se existe uma prova de H, em $\wp_a$, que utiliza apenas os axiomas. Nesse caso, o conjunto de hipóteses é vazio.*

[5] O conjunto de hipóteses β pode ser finito ou não. Mas, se ele é infinito, utilizamos em geral conjuntos, pelo menos, enumeráveis, ou decidíveis. Mas, não se preocupe com tais detalhe técnicos. Este livro é apenas introdutório.

Um teorema é uma fórmula derivável no sistema axiomático $\wp_a$, somente a partir de seus axiomas. Isto é, um teorema é uma fórmula H derivável a partir de um conjunto vazio de hipóteses.

Notação. Dada uma fórmula H, se H é consequência lógica sintática de um conjunto de hipóteses β, tal que $\beta = \{H_1, H_2, \ldots, H_n, \ldots\}$, então esse fato é denotado por $\beta \vdash H$ ou $\{H_1, H_2, \ldots, H_n, \ldots\} \vdash H$. Se H não é consequência lógica sintática de β, utilizamos a notação $\beta \nvdash H$. No caso em que H é um teorema, ou seja, β é vazio, então utilizamos a notação $\vdash H$. E, se H não é um teorema, denotamos isso por $\nvdash H$.

É importante enfatizar que o símbolo $\vdash$ é utilizado para denotar consequência lógica sintática, que não considera a interpretação das fórmulas.

Notação. Para distinguir as consequências lógicas sintática e semântica, utilizamos a seguinte notação para consequência lógica semântica. Dada uma fórmula H e um conjunto de fórmulas β, tal que $\beta = \{H_1, H_2, \ldots, H_n, \ldots\}$, se β implica semanticamente H, então H é consequência lógica semântica de β e este fato é denotado por $\beta \vDash H$ ou $\{H_1, H_2, \ldots, H_n, \ldots\} \vDash H$. Se H não é consequência lógica semântica de β, utilizamos a notação $\beta \nvDash H$. No caso em que H é uma tautologia, ou seja, β é vazio, então utilizamos a notação $\vDash H$. E se H não é uma tautologia, denotamos isso por $\nvDash H$.

Observe que o símbolo $\vDash$ é utilizado para denotar consequência lógica semântica, que considera a interpretação das fórmulas. Portanto, apesar da ligeira diferença, os símbolos $\vDash$ e $\vdash$ denotam conceitos bastante distintos. Uma prova sintática é um procedimento diferente de uma prova semântica, como aquela dos *tableaux* semânticos ou tabela-verdade, que utilizam a interpretação dos símbolos das fórmulas. Na prova sintática não levamos em conta a interpretação desses símbolos. Além disso, ela é um procedimento mecânico, não necessariamente determinístico.[6] A prova sintática é determinada por um conjunto de instruções que definem tal procedimento e que até podem ser executadas por um computador, não exigindo qualquer criatividade ou engenhosidade para a sua execução. Entretanto, no caso de uma prova, conforme Definição 5.4, o mais difícil é escolher corretamente os axiomas e hipóteses a serem utilizados para determinar a sequência de fórmulas $H_1, H_2, \ldots, H_n$. Nem sempre essa escolha é imediata, pois os axiomas e hipóteses denotam conjuntos infinitos de fórmulas. Por isso, dada uma fórmula H e um conjunto de hipóteses β, nem sempre é fácil saber se $\beta \vdash H$. Ao tentar responder essa questão, a execução do procedimento, determinado pela definição de prova, pode não determinar nenhuma resposta. Nesse caso, o procedimento computacional não termina após um número finito de passos. Logo, nesse caso, ele não é um procedimento efetivo. E se isso ocorre, não podemos responder se $\beta \vdash H$ e nem concluir o contrário, que $\beta \nvdash H$. Observe a importante diferença. No método dos *tableaux* semânticos, se não podemos responder que $\beta \vDash H$, então concluímos o contrário, que $\beta \nvDash H$. Por outro lado, no método de prova sintático, determinado pelo sistema $\wp_a$, se não podemos responder que $\beta \vdash H$, então não necessariamente concluímos o contrário, que $\beta \nvdash H$. Tais problemas

[6] Pesquise os significados dos termos: "procedimento" e "determinístico".

não ocorreriam se fosse possível encontrar um procedimento efetivo, decidível, que determinasse se $\beta \vdash H$, sem entrar em loop, ou entrar por um conjunto sem fim de operações.[7] Mas, infelizmente, não temos procedimentos efetivos que decidem e respondem se $\beta \vdash H$, ou $\beta \nvdash H$. Considere agora uma outra importante questão: Qual a razão para escolher o conjunto de axiomas da Definição 5.2 e a regra *modus ponens*? Poderiam ser escolhidos outros axiomas, como também outro tipo de regra de inferência. Espere um pouco. A análise dessa questão é considerada mais adiante, após a demonstração do teorema da completude no sistema $\wp_a$.

5.4 Proposições no sistema $\wp_a$

Apresentamos, a seguir, um conjunto de proposições cujos resultados são utilizados na demonstração do teorema da completude, que é o principal resultado deste capítulo. Em uma primeira leitura, não se preocupe com os detalhes das demonstrações. Basta entender os resultados e isso não compromete o entendimento do resto deste capítulo.

Proposição 5.1 *Sejam β um conjunto de fórmulas, e A, B e C três fórmulas da Lógica Proposicional. Temos que: se* $\{\beta \vdash (A \rightarrow B)$ e $\beta \vdash (C \vee A)\}$, *então* $\{\beta \vdash (B \vee C)\}$.

Demonstração. A sequência de fórmulas H_1, H_2 e H_3 demonstra esta proposição.

A primeira fórmula da prova. A primeira fórmula da prova, H_1, é obtida a partir do axioma Ax_3 fazendo $H = A,\ G = B$ e $E = C$. Logo:

1. $H_1 = (A \rightarrow B) \rightarrow ((C \vee A) \rightarrow (B \vee C))$

A segunda fórmula da prova. Por hipótese, temos $\beta \vdash (A \rightarrow B)$, então fazemos:

2. $H_2 = (A \rightarrow B)$

A terceira fórmula da prova. Aplicando *modus ponens* em H_1 e H_2, temos:

$$MP = \frac{(A \rightarrow B),\ ((A \rightarrow B) \rightarrow ((C \vee A) \rightarrow (B \vee C)))}{((C \vee A) \rightarrow (B \vee C))}$$

isto é, obtemos a prova da terceira fórmula.

3. $H_3 = ((C \vee A) \rightarrow (B \rightarrow C))$

A quarta fórmula da prova. Por hipótese, temos $\beta \vdash (C \vee A)$, então fazemos:

4. $H_4 = C \vee A$

[7]Esse tipo de estudo é importante na Lógica e em Teoria da Computação [Cooper], [Sipser] e [Smith].

A quinta fórmula da prova. Aplicando *modus ponens*, em H_3 e H_4 temos:

$$MP = \frac{(C \vee A),\ ((C \vee A) \rightarrow (B \vee C))}{(B \vee C)}$$

isto é, obtemos a prova da quinta fórmula.

5. $H_5 = (B \vee C)$

Portanto, concluímos que $\beta \vdash (B \vee C)$. Observe que não é o caso que $\vdash (B \vee C)$. Isso ocorre porque na prova acima, as fórmulas H_2 e H_4 são consideradas. Tais fórmulas são consequências sintáticas de β. Isto é, assumimos como ponto de partida que $\beta \vdash (A \rightarrow B)$ e $\beta \vdash (C \vee A)$; e não que $\vdash (A \rightarrow B)$ e $\vdash (C \vee A)$. **cqd** ■

Proposição 5.2 *Temos que* $\vdash (P \vee \neg P)$.

Demonstração. Nesta demonstração, utilizamos uma notação mais simples:

1. $A = ((P \vee P) \rightarrow P) \rightarrow ((\neg P \vee (P \vee P)) \rightarrow (P \vee \neg P))$.
   Fórmula obtida de Ax_3 fazendo $H = (P \vee P)$, $G = P$ e $E = \neg P$;
2. $B = (P \vee P) \rightarrow P$.
   Fórmula obtida de Ax_1 fazendo $H = P$;
3. $C = (\neg P \vee (P \vee P)) \rightarrow (P \vee \neg P)$.
   Resultado de MP em A e B.
   Em C, a subfórmula $\neg P \vee (P \vee P)$ pode ser denotada como $P \rightarrow (P \vee P)$. Logo:
   $C = (P \rightarrow (P \vee P)) \rightarrow (P \vee \neg P)$.
4. $D = P \rightarrow (P \vee P)$.
   Fórmula obtida a partir do axioma Ax_2, fazendo $H = P$ e $G = P$.
5. $H = P \vee \neg P$.
   Resultado de MP em C e D.

Portanto, $\vdash (P \vee \neg P)$. **cqd** ■

Notação. Para denotar que um símbolo proposicional P ocorre em uma fórmula H, escrevemos: $H_{[P]}$. Da mesma forma, se os símbolos proposicionais $P_1, \ldots, P_n$ ocorrem em uma fórmula H, escrevemos: $H_{[P_1,\ldots,P_n]}$. Suponha, por exemplo, $H = (P \leftrightarrow Q) \vee (R \rightarrow S)$. Então, podemos denotar H por $H_{[P,Q,R,S]}$. Observe que essa notação somente dá ênfase ao fato de P, Q, R e S ocorrerem em H, pois continuamos tendo $H_{[P,Q,R,S]} = (P \leftrightarrow Q) \vee (R \rightarrow S)$.

Proposição 5.3 (regra da substituição) *Sejam β um conjunto de fórmulas e $H_{[P_1,\ldots,P_n]}$ uma fórmula da Lógica Proposicional tais que $\beta \vdash H_{[P_1,\ldots,P_n]}$. Considere $\{P_1, \ldots, P_n\}$ um conjunto de símbolos proposicionais que ocorrem em H, mas não ocorrem nas fórmulas de β. Seja $G_{[E_1,\ldots,E_n]}$ a fórmula obtida de $H_{[P_1,\ldots,P_n]}$, substituindo os símbolos proposicionais $P_1, \ldots, P_n$ pelas fórmulas $E_1, \ldots, E_n$, respectivamente. Temos que $\beta \vdash G_{[E_1,\ldots,E_n]}$.*

Nota. A demonstração da regra de substituição, vista na Proposição 5.3, utiliza o princípio da indução finita e não é considerada neste livro.

Entretanto, podemos fazer um breve comentário. Para obter a prova de G a partir de β, basta substituir os símbolos proposicionais $P_1, \ldots, P_n$ pelas fórmulas $E_1, \ldots, E_n$, respectivamente, em todos os passos da prova de H a partir de β. Observe que nesse argumento é essencial que os símbolos proposicionais $P_1, \ldots, P_n$ não ocorram nas fórmulas do conjunto β. Observe, portanto, que as substituições são aplicadas apenas à fórmula H e não às fórmulas do conjunto de hipóteses β. Esta é a razão para considerarmos que $\{P_1, \ldots, P_n\}$ é um conjunto de símbolos proposicionais que ocorrem em H, mas não nas fórmulas de β .

Proposição 5.4 *Temos que* $\vdash (P \rightarrow \neg\neg P)$.

Demonstração. Em $\wp_a$, a fórmula $(P \rightarrow \neg\neg P)$ é uma denotação da fórmula $(\neg P \vee \neg\neg P)$. Portanto, basta demonstrar que $\vdash (\neg P \vee \neg\neg P)$. Conforme a Proposição 5.2, temos que $\vdash (P \vee \neg P)$. Utilizando a regra de substituição, de acordo com a Proposição 5.3, e substituindo P por $\neg P$ em $\vdash (P \vee \neg P)$, obtemos a prova de $\vdash (\neg P \vee \neg\neg P)$. **cqd** ■

Exemplo 5.3 (regra de substituição) Este exemplo considera uma forma incorreta de utilização da regra de substituição. Considere $\beta = \{P, P \rightarrow Q\}$. Aplicando *modus ponens* às fórmulas de β, o símbolo proposicional Q é deduzido de P e $P \rightarrow Q$. Logo, $\beta \vdash Q$. Aplicando de forma incorreta a regra de substituição, Proposição 5.3, nesse resultado, e substituindo Q por R, na fórmula Q, concluímos $\beta \vdash R$. Mas é falso que $\{P, P \rightarrow Q\} \vdash R$. Nesse caso, a regra de substituição foi aplicada de forma incorreta, pois Q ocorre em uma fórmula de $\{P, P \rightarrow Q\}$. Logo, Q não pode ser substituído. A questão importante a ser observada é que a substituição considerada na Proposição 5.3 não é global, pois não considera as fórmulas do conjunto de hipóteses. ■

Na demonstração da Proposição 5.5, a seguir, é utilizada uma notação diferente daquela indicada anteriormente neste capítulo. Essa nova notação é frequentemente utilizada em demonstrações na lógica, [Andrews], e muitas vezes simplifica a indicação da sequência de passos.

Proposição 5.5 *Temos que* $\vdash (P \rightarrow P)$.

Demonstração. Na notação utilizada nesta demonstração, cada linha é enumerada à esquerda. Além disso, à direita de cada linha são colocadas as justificativas e a partir delas é possível deduzir a presente linha. Tais justificativas são escritas de forma abreviada. Uma justificativa do tipo "$MP, pr5.4, Ax_3$", por exemplo, significa que a regra *modus ponens* foi aplicada ao resultado da Proposição 5.4 e ao axioma Ax_3 com as devidas substituições. Observe que as substituições no axioma ficam implícitas e não são indicadas. Além disso, $ex6$, por exemplo, indica a utilização do

Exercício 6 deste capítulo. O leitor é convidado a detalhar todas as justificativas dessa prova e das subsequentes.

1.	$\vdash (P \vee P) \to (\neg\neg P \vee P)$	$MP, pr5.4, Ax_3$
2.	$\vdash (\neg\neg P \vee P) \to (\neg\neg P \vee \neg\neg P)$	$MP, pr5.4, Ax_3$
3.	$\vdash \neg P \vee (P \vee P)$	Ax_2
4.	$\vdash (\neg\neg P \vee P) \vee \neg P$	$pr5.1, 1., 3.$
5.	$\vdash \neg\neg\neg P \vee (\neg\neg P \vee P)$	$pr5.3, pr5.4, pr5.1, 4.$
6.	$\vdash (\neg\neg P \vee \neg\neg P) \vee \neg\neg\neg P$	$pr5.1, 2., 5.$
7.	$\vdash \neg P \vee (\neg\neg P \vee \neg\neg P)$	$pr5.3, ex6, pr5.1, 6.$
8.	$\vdash \neg\neg P \vee \neg P$	$pr5.1, Ax_1, 7.$
9.	$\vdash P \vee \neg\neg\neg P$	$pr5.3, 8., pr5.1, ex5$
10.	$\vdash \neg P \vee P$	$pr5.3, ex6, pr5.1, 9.$

Conclusão: temos que $\vdash (P \to P)$. **cqd** ■

Nas proposições que se seguem, β é um conjunto de hipóteses e A, B e C são fórmulas. Evidentemente, após o enunciado de cada proposição, é apresentada sua demonstração.

Proposição 5.6 *Temos que* $\vdash (A \vee B) \to (B \vee A)$.

1.	$\vdash (P \to P)$	$pr5.5$
2.	$\vdash (B \to B)$	$pr5.3, 1.$
3.	$\vdash (B \to B) \to ((A \vee B) \to (B \vee A))$	Ax_3
4.	$\vdash (A \vee B) \to (B \vee A)$	$MP, 2., 3.$ **cqd**■

Proposição 5.7 (transitividade) *Se* $\beta \vdash (A_1 \to A_2)$ *e* $\beta \vdash (A_2 \to A_3)$, *então* $\beta \vdash (A_1 \to A_3)$.

1.	$\beta \vdash (\neg A_1 \vee A_2)$	*hip*
2.	$\beta \vdash (A_2 \to A_3)$	*hip*
3.	$\beta \vdash (A_3 \vee \neg A_1)$	$pr5.1, 1., 2.$
4.	$\beta \vdash (A_3 \vee \neg A_1) \to (\neg A_1 \vee A_3)$	$pr5.6$
5.	$\beta \vdash (\neg A_1 \vee A_3)$	$MP, 3., 4.$
5.	$\beta \vdash (A_1 \to A_3)$	reescrita de 5. **cqd**■

Proposição 5.8 *Se* $\beta \vdash (A \to C)$ *e* $\beta \vdash (B \to C)$, *então* $\beta \vdash ((A \vee B) \to C)$.

1.	$\beta \vdash (B \to C)$	*hip*
2.	$\beta \vdash (B \to C) \to ((A \vee B) \to (C \vee A))$	Ax_3
3.	$\beta \vdash (A \vee B) \to (C \vee A)$	$MP, 1., 2.$
4.	$\beta \vdash (A \to C)$	*hip*
5.	$\beta \vdash (A \to C) \to ((C \vee A) \to (C \vee C))$	Ax_3
6.	$\beta \vdash (C \vee A) \to (C \vee C)$	$MP, 4., 5.$
7.	$\beta \vdash (A \vee B) \to (C \vee C)$	$pr5.7, 3., 6.$
8.	$\beta \vdash (C \vee C) \to C$	Ax_1
9.	$\beta \vdash (A \vee B) \to C$	$pr5.7, 7., 8.$ **cqd**■

Proposição 5.9 *Se* $\beta \vdash (A \to C)$ *e* $\beta \vdash (\neg A \to C)$*, então* $\beta \vdash C$.

1.	$\beta \vdash (A \to C)$	*hip*
2.	$\beta \vdash (\neg A \to C)$	*hip*
3.	$\beta \vdash (A \vee \neg A) \to C$	$pr5.8, 1., 2.$
4.	$\beta \vdash (A \vee \neg A)$	$pr5.2$
5.	$\beta \vdash C$	$MP, 3., 4.$**cqd**■

Proposição 5.10 *Se* $\beta \vdash (A \to B)$*, então* $\beta \vdash (A \to (C \vee B))$ *e* $\beta \vdash (A \to (B \vee C))$.

1.	$\beta \vdash (A \to B)$	*hip*
2.	$\beta \vdash B \to (C \vee B)$	Ax_2
3.	$\beta \vdash A \to (C \vee B)$	$pr5.7, 1., 2.$
4.	$\beta \vdash (C \vee B) \to (B \vee C)$	$pr5.3, pr5.6$
5.	$\beta \vdash A \to (B \vee C)$	$pr5.7, 3., 4.$**cqd**■

Proposição 5.11 (associatividade) *Temos que* $\vdash ((A \vee B) \vee C) \to (A \vee (B \vee C))$.

1.	$\vdash (P \to P)$	$pr5.5$
2.	$\vdash A \to (A \vee (B \vee C))$	$pr5.3, 1., pr5.10$
3.	$\vdash B \to (B \vee C)$	$pr5.3, 1., pr5.10$
4.	$\vdash B \to (A \vee (B \vee C))$	$pr5.10, 3.$
5.	$\vdash (A \vee B) \to (A \vee (B \vee C))$	$pr5.8, 2., 4.$
6.	$\vdash C \to (B \vee C)$	$pr5.3, 1., pr5.10$
7.	$\vdash C \to (A \vee (B \vee C))$	$pr5.10, 6.$
8.	$\vdash ((A \vee B) \vee C) \to (A \vee (B \vee C))$	$pr5.8, 5., 7.$**cqd**■

Proposição 5.12 *Se* $\beta \vdash ((A \vee B) \vee C)$*, então* $\beta \vdash (A \vee (B \vee C))$.

1.	$\beta \vdash (A \vee B) \vee C$	*hip*
2.	$\beta \vdash ((A \vee B) \vee C) \to (A \vee (B \vee C))$	$pr5.11$
3.	$\beta \vdash (A \vee (B \vee C))$	$MP, 1., 2.$**cqd**■

Proposição 5.13 *Se* $\beta \vdash (A \to B)$ *e* $\beta \vdash (A \to (B \to C))$*, então* $\beta \vdash (A \to C)$.

1.	$\beta \vdash (A \to B)$	*hip*
2.	$\beta \vdash (\neg A \vee (\neg B \vee C))$	*hip*
3.	$\beta \vdash ((\neg B \vee C) \vee \neg A)$	$pr5.6, 2.$
4.	$\beta \vdash (\neg B \vee (C \vee \neg A))$	$pr5.12, 3.$
4.	$\beta \vdash (B \to (C \vee \neg A))$	reescrita
5.	$\beta \vdash (A \to (C \vee \neg A))$	$pr5.7, 1., 4.$
5.	$\beta \vdash (\neg A \vee (C \vee \neg A))$	reescrita
6.	$\beta \vdash ((C \vee \neg A) \vee \neg A)$	$pr5.6, 5.$
7.	$\beta \vdash (C \vee (\neg A \vee \neg A))$	$pr5.12, 6.$
8.	$\beta \vdash (\neg A \vee C)$	$pr5.1, Ax_1, 7.$
9.	$\beta \vdash (A \to c)$	reescrita **cqd**■

Consideramos, a seguir, o teorema da dedução na sua versão sintática, cujo resultado é utilizado na demonstração do teorema da completude. Na demonstração desse teorema, utilizamos o princípio da indução finita e o lema a seguir.

Lema 5.1 *Suponha que $\beta \cup \{A\} \vdash B$ e que $B \in \beta$, ou $B = A$, ou B é axioma. Temos, então, que $\beta \vdash (A \rightarrow B)$.*

Demonstração. Essa demonstração é dividida em dois casos:

Caso 1: Suponha que $B \in \beta$, ou B é um axioma. Logo, $\beta \vdash B$. Utilizando $Ax_2 = H \rightarrow (G \vee H)$, com $H = B$, $G = \neg A$, temos a prova $\beta \vdash B \rightarrow (A \rightarrow B)$. Aplicando a regra *modus ponens* aos resultados $\beta \vdash B$ e $\beta \vdash B \rightarrow (A \rightarrow B)$, então obtemos: $\beta \vdash (A \rightarrow B)$.

Caso 2: Suponha que $B = A$. Utilizando a Regra de Substituição, Proposição 5.3, no resultado $\vdash (P \rightarrow P)$, que é dado pela Proposição 5.5, obtemos $\vdash (A \rightarrow A)$. Logo $\beta \vdash (A \rightarrow B)$. **cqd**■

Teorema 5.1 (teorema da dedução - forma sintática) *Se $\beta \cup \{A\} \vdash B$, então $\beta \vdash (A \rightarrow B)$.*

Demonstração. Temos que $\beta \cup \{A\} \vdash B$ e queremos demonstrar que $\beta \vdash (A \rightarrow B)$. Como $\beta \cup \{A\} \vdash B$, então existe uma sequência de fórmulas $C_1, \ldots, C_n$, que é uma prova de B a partir de $\beta \cup \{A\}$. Nesse caso, temos $B = C_n$. É demonstrado a seguir, utilizando o princípio da indução, que $\beta \vdash (A \rightarrow C_i)$ para todo inteiro i, $1 \leq i \leq n$. A partir desse fato, fazendo $i = n$, concluímos que $\beta \vdash (A \rightarrow B)$, pois $B = C_n$. A demonstração de $\beta \vdash (A \rightarrow C_i)$ utiliza o princípio da indução no inteiro i. Considere a asserção:

$$A(i) \equiv \text{ se } \beta \cup \{A\} \vdash C_i, \text{ então } \beta \vdash (A \rightarrow C_i).$$

Base da indução. A base da indução corresponde ao caso $i = 1$. Fazendo $i = 1$ na asserção $A(i)$ obtemos:

$$A(1) \equiv \text{ se } \beta \cup \{A\} \vdash C_1, \text{ então } \beta \vdash (A \rightarrow C_1).$$

Portanto, devemos demonstrar que $\beta \vdash (A \rightarrow C_1)$, tendo que $\beta \cup \{A\} \vdash C_1$. Observe que C_1 é a primeira fórmula da prova a partir de $\beta \cup \{A\}$, logo $C_1 \in \beta \cup \{A\}$ ou C_1 é um axioma. Nesse caso, utilizando o lema 5.1, temos que $\beta \vdash (A \rightarrow C_1)$

Passo da indução. Esta demonstração considera a segunda forma do princípio da indução na Lógica. Nesse caso, para cada inteiro i, $1 \leq i \leq n$, é necessário demonstrar que se $A(k)$ é verdadeira para todo k, $k < i$, então $A(i)$ também é verdadeira. Em outras palavras, se a asserção:

$$A(k) \equiv \text{ se } \beta \cup \{A\} \vdash C_k, \text{ então } \beta \vdash (A \rightarrow C_k).$$

é verdadeira para todo k, $k < i$, então também é verdadeira a asserção:

$$A(i) \equiv \text{ se } \beta \cup \{A\} \vdash C_i, \text{ então } \beta \vdash (A \rightarrow C_i).$$

Para demonstrar $A(i)$, suponha $\beta \cup \{A\} \vdash C_i$. O resultado $\beta \vdash (A \rightarrow C_i)$ é demonstrado a seguir. Nos casos em que C_i é axioma, hipótese, ou é igual a A, então, utilizando o Lema 5.1, concluímos que $\beta \vdash (A \rightarrow C_i)$. No caso em que C_i não é axioma, nem hipótese, e é diferente de A, não podemos utilizar o Lema 5.1. Nesse caso, na sequência da prova de C_i, existem as fórmulas C_j e $(C_j \rightarrow C_i)$, a partir das quais, C_i é deduzida, utilizando *modus ponens*. Portanto, devemos ter: $C_s = (C_j \rightarrow C_i)$, onde $j < i$, $s < i$. Isto é, as fórmulas C_j e $(C_j \rightarrow C_i)$ pertencem à prova de C_i. Observe que, nesse caso, C_i é deduzida de C_j e $(C_j \rightarrow C_i)$, utilizando *modus ponens*. Como C_j e C_s pertencem à sequência de fórmulas da prova de C_i, temos: $\beta \cup \{A\} \vdash C_j$ e $\beta \cup \{A\} \vdash C_k$. Então, utilizando que:

$$A(k) \equiv \text{ se } \beta \cup \{A\} \vdash C_k, \text{ então } \beta \vdash (A \rightarrow C_k)$$

é verdadeira para todo k, $k < i$, concluímos: $\beta \vdash (A \rightarrow C_j)$ e $\beta \vdash (A \rightarrow C_s)$. Substituindo a fórmula C_s, temos as duas provas: $\beta \vdash (A \rightarrow C_j)$ e $\beta \vdash (A \rightarrow (C_j \rightarrow C_i))$. Utilizando a Proposição 5.13, concluímos o resultado $\beta \vdash (A \rightarrow C_i)$. **cqd**■

Exemplo 5.4 (utilização do teorema da dedução) Este exemplo demonstra, utilizando o teorema da dedução, Teorema 5.1, que:

$$\vdash Q \rightarrow (P \rightarrow (P_2 \rightarrow ((P_2 \rightarrow (\neg P \vee \neg Q)) \rightarrow P_{100}))). \tag{5.1}$$

Se for demonstrado que:

$$\{Q\} \vdash P \rightarrow (P_2 \rightarrow ((P_2 \rightarrow (\neg P \vee \neg Q)) \rightarrow P_{100})), \tag{5.2}$$

então, podemos utilizar o teorema da dedução e obter o Resultado 5.1. Da mesma forma, se for demonstrado que:

$$\{Q, P\} \vdash P_2 \rightarrow ((P_2 \rightarrow (\neg P \vee \neg Q)) \rightarrow P_{100}), \tag{5.3}$$

então novamente podemos utilizar o teorema da dedução e obter o Resultado 5.2 e, portanto, o Resultado 5.1. Seguindo esse raciocínio, para demonstrar o Resultado 5.1 basta demonstrar:

$$\{Q, P, P_2, (P_2 \rightarrow (\neg P \vee \neg Q))\} \vdash P_{100}, \tag{5.4}$$

e utilizar o teorema da dedução, repetidas vezes, para se obter o Resultado 5.1. A questão agora é como demonstrar a dedução 5.4? Observe inicialmente que para deduzir P_{100} é necessário utilizar a regra MP. Isso ocorre porque P_{100} não é um axioma e nem uma hipótese. Logo, na sequência de fórmulas que constituem a prova de P_{100}, a última fórmula é igual a P_{100} e existem fórmulas anteriores iguais a $H \rightarrow P_{100}$ e H. Mas quem é a fórmula H? Faça alguma tentativa. Nesse caso, consideramos $H = Q$, dado que Q é uma hipótese, o que parece ser uma boa escolha. Portanto, no final da prova de P_{100}, temos a sequência:

$H_{n-2} = Q$	hipótese
$H_{n-1} = Q \rightarrow P_{100}$	a ser provada
$H_n = P_{100}$	$MP,\ H_{n-2},\ H_{n-1}$

Assim, para provar P_{100}, é necessário provar $Q \rightarrow P_{100}$. Observe que Q é uma hipótese, não sendo necessário prová-la. Uma boa ideia para provar $Q \rightarrow P_{100}$ é utilizar o axioma 3, considerando:

$$H_1 = Ax_3 = (P \rightarrow \neg Q) \rightarrow ((P_{100} \vee P) \rightarrow (Q \rightarrow P_{100}))$$

Observe que a última implicação em H_1 é exatamente igual a H_{n-1}, a fórmula a ser provada. Nesse caso, para provar H_{n-1}, utilizando H_1, é necessário aplicar a regra *modus ponens* duas vezes para eliminar os antecedentes $(P \rightarrow \neg Q)$ e $(P_{100} \vee P)$. Essas eliminações são obtidas pela sequência:

$H_1 =$	
$(P \rightarrow \neg Q) \rightarrow ((P_{100} \vee P) \rightarrow (Q \rightarrow P_{100}))$	axioma Ax_3
$H_{n-5} = (P \rightarrow \neg Q)$	a ser provada
$H_{n-4} = (P_{100} \vee P)$	a ser provada
$H_{n-3} = (P_{100} \vee P) \rightarrow (Q \rightarrow P_{100}))$	$MP,\ H_1,\ H_{n-5}$
$H_{n-2} = Q$	hipótese
$H_{n-1} = Q \rightarrow P_{100}$	$MP,\ H_{n-4},\ H_{n-3}$
$H_n = P_{100}$	$MP,\ H_{n-2},\ H_{n-1}$

Nesse ponto devemos provar que $(P \rightarrow \neg Q)$ e $(P_{100} \vee P)$. No caso de $(P \rightarrow \neg Q)$, observe inicialmente que, em $\wp_a$, essa fórmula é uma denotação de $(\neg P \vee \neg Q)$. A prova dessa última fórmula, utilizando as hipóteses, é dada pela sequência:

$H_1 =$	
$(P \rightarrow \neg Q) \rightarrow ((P_{100} \vee P) \rightarrow (Q \rightarrow P_{100}))$	axioma Ax_3
$H_{n-7} = P_2$	hipótese
$H_{n-6} = P_2 \rightarrow (\neg P \vee \neg Q)$	hipótese
$H_{n-5} = (\neg P \vee \neg Q)$	$MP,\ H_{n-7},\ H_{n-6}$
$H_{n-4} = (P_{100} \vee P)$	a ser provada
$H_{n-3} = (P_{100} \vee P) \rightarrow (Q \rightarrow P_{100}))$	$MP,\ H_1,\ H_{n-5}$
$H_{n-2} = Q$	hipótese
$H_{n-1} = Q \rightarrow P_{100}$	$MP,\ H_{n-4},\ H_{n-3}$
$H_n = P_{100}$	$MP,\ H_{n-2},\ H_{n-1}$

Para provar $(P_{100} \vee P)$ é necessário utilizar o axioma 2, na forma $Ax_2 = P \rightarrow (P_{100} \vee P)$ e a hipótese P, obtendo a sequência:

$H_1 =$	
$(P \to \neg Q) \to ((P_{100} \vee P) \to (Q \to P_{100}))$	axioma Ax_3
$H_{n-9} = P$	hipótese
$H_{n-8} = P \to (P_{100} \vee P)$	axioma Ax_2
$H_{n-7} = P_2$	hipótese
$H_{n-6} = P_2 \to (\neg P \vee \neg Q)$	hipótese
$H_{n-5} = (\neg P \vee \neg Q)$	$MP,\ H_{n-7},\ H_{n-6}$
$H_{n-4} = (P_{100} \vee P)$	$MP,\ H_{n-9},\ H_{n-8}$
$H_{n-3} = (P_{100} \vee P) \to (Q \to P_{100}))$	$MP,\ H_1,\ H_{n-5}$
$H_{n-2} = Q$	hipótese
$H_{n-1} = Q \to P_{100}$	$MP,\ H_{n-4},\ H_{n-3}$
$H_n = P_{100}$	$MP,\ H_{n-2},\ H_{n-1}$

Para obter a prova final, basta ajustar o índice n, fazendo $n = 11$. ■

Os resultados das proposições a seguir são utilizados na próxima seção.

Proposição 5.14 *Temos que* $\vdash (\neg A \to (\neg B \to \neg(A \vee B)))$.

1.	$\vdash A \to \neg\neg A$	$pr5.3, pr5.4$
2.	$\vdash B \to \neg\neg B$	$pr5.3, pr5.4$
3.	$\vdash A \to (\neg\neg A \vee \neg\neg B)$	$pr5.10, 1.$
4.	$\vdash B \to (\neg\neg A \vee \neg\neg B)$	$pr5.10, 2.$
5.	$\vdash (A \vee B) \to (\neg\neg A \vee \neg\neg B)$	$pr5.8, 3., 4.$
6.	$\vdash (\neg\neg A \vee \neg\neg B) \vee \neg(A \vee B)$	$pr5.6, 5.$
7.	$\vdash \neg\neg A \vee (\neg\neg B \vee \neg(A \vee B))$	$pr5.12, 6.$
7.	$\vdash \neg A \to (\neg B \to \neg(A \vee B))$	reescrita **cqd**■

Proposição 5.15 *Temos que* $\vdash A \to (A \vee B)$ *e* $\vdash \neg A \to (\neg A \vee B)$.

Demonstração.

Prova de $\vdash A \to (A \vee B)$.

1.	$\vdash A \to A$	$pr5.3, pr5.5$
2.	$\vdash A \to (A \vee B)$	$pr5.10, 1.$

Prova de $\vdash \neg A \to (\neg A \vee B)$.

1.	$\vdash \neg A \vee \neg\neg A$	$pr5.3, pr5.4$
2.	$\vdash (\neg A \vee \neg\neg A) \to (\neg\neg A \vee \neg A)$	$pr5.3, pr5.6$
3.	$\vdash (\neg\neg A \vee \neg A)$	$MP, 1., 2.$
3.	$\vdash \neg A \to \neg A$	reescrita
4.	$\vdash \neg A \to (\neg A \vee B)$	$pr5.10, 3.$ **cqd**■

5.5 Correção e completude do sistema $\wp_a$

Na definição do sistema axiomático $\wp_a$ é escolhido um conjunto de axiomas e a regra de inferência *modus ponens*. Uma questão importante é a seguinte: por que foram escolhidos os axiomas Ax_1, Ax_2, Ax_3 e a regra de inferência *modus ponens*? Outros axiomas e regras também poderiam ser escolhidos. Há pelo menos duas razões para isso.

1. O sistema axiomático deve ser correto, ou seja, todo teorema deve ser uma tautologia. Isso significa que o conhecimento deduzido a partir dos axiomas, utilizando as regras de inferência, deve ser válido. Observe que um sistema axiomático de dedução não correto não poderia ter aplicações. Nesse caso, haveria a dedução de conhecimento não válido a partir dos axiomas. Portanto, a correção é uma propriedade desejável em um sistema axiomático.
2. O sistema axiomático deve ser completo, ou seja, toda tautologia deve ser um teorema. Dada uma tautologia qualquer, se o sistema axiomático é completo, então a tautologia é um teorema. Isso significa que existe, no sistema axiomático, uma prova dessa tautologia a partir dos axiomas. A denominação "sistema completo" se deve a esse fato. Toda tautologia pode ser derivada sintaticamente no sistema axiomático. Observe que muitas vezes pode ser fácil identificar se uma fórmula é uma tautologia. Entretanto, a partir da tautologia, determinar a sua prova utilizando os axiomas do sistema axiomático pode não ser tão fácil.

Os teoremas a seguir formalizam tais questões:

Teorema 5.2 (teorema da correção) *Seja H uma fórmula da Lógica Proposicional, se H é um teorema, então H é uma tautologia.*

Esquema de demonstração. Observe inicialmente que os axiomas Ax_1, Ax_2 e Ax_3 são tautologias, como já foi demonstrado anteriormente. Na regra *modus ponens*, a fórmula G é deduzida de duas outras fórmulas: H e $(H \rightarrow G)$. Isto é:

$$\frac{H, (H \rightarrow G)}{G}$$

Mas, se $\vDash H$ e $\vDash (H \rightarrow G)$, então $\vDash G$. Ou seja, a partir de tautologias, utilizando *modus ponens*, são deduzidas apenas tautologias (ver exercícios). Se $\vdash H$, então existe uma sequência de fórmulas $H_1, \ldots, H_n$ tal que $H = H_n$. Além disso, dada uma fórmula H_i dessa sequência, então há duas possibilidades:

a) H_i é um axioma e, portanto, uma tautologia;

b) H_i é deduzida, utilizando *modus ponens*, de fórmulas anteriores na sequência. Considerando que a regra *modus ponens* mantém a validade, logo, para todo i, H_i também é tautologia. Portanto, H é uma tautologia.

Essa demonstração pode ser feita, de forma mais rigorosa, utilizando indução no comprimento da prova (ver exercícios). **cqd**■

Teorema 5.3 (teorema da completude) *Seja H uma fórmula da Lógica Proposicional, se H é uma tautologia, então H é um teorema.*

A demonstração desse teorema utiliza o teorema da dedução, o princípio da indução finita e os conceitos considerados a seguir. Além disso, essa demonstração somente é válida se H não contém símbolos de verdade. Essa é uma das razões pelas quais o alfabeto do sistema $\wp_a$ não contém símbolos de verdade.

Definição 5.7 (base associada a uma fórmula.) *Seja H uma fórmula e $P_1, \ldots,$ P_n os símbolos proposicionais contidos em H. Dada uma interpretação I, então a base associada a H conforme I, denotada por $B[H, I]$, é um conjunto de literais, definidos a partir de $P_1, \ldots, P_n$ como se segue:*

- *se $I[P_i] = T$, então $P_i \in B[H, I]$;*
- *se $I[P_i] = F$, então $\neg P_i \in B[H, I]$.*

Exemplo 5.5 (base associada a uma fórmula) Considere a fórmula:
$H = \neg(P \rightarrow Q) \rightarrow (Q \rightarrow P)$. Seja também I uma interpretação tal que $I[P] = T$ e $I[Q] = F$. Nesse caso, $B[H, I] = \{P, \neg Q\}$. ■

Lema 5.2 *Seja H uma fórmula e $P_1, ..., P_n$ os símbolos proposicionais contidos em H. Dada uma interpretação I, então:*

a) $I[H] = T \Rightarrow B[H, I] \vdash H$.

b) $I[H] = F \Rightarrow B[H, I] \vdash \neg H$.

Antes de apresentar a demonstração do Lema 5.2, considere o exemplo a seguir:

Exemplo 5.6 Sejam H e I, respectivamente, a fórmula e a interpretação do Exemplo 5.5. Nesse caso, $I[H] = T$. Logo, a partir do resultado determinado pelo Lema 5.2, temos:

$$\{P, \neg Q\} \vdash \neg(P \rightarrow Q) \rightarrow (Q \rightarrow P)$$

■

Demonstração (Lema 5.2). Inicialmente, lembre-se de que o sistema $\wp_a$ utiliza o alfabeto simplificado da Lógica Proposicional. Por isso, nessa demonstração, as fórmulas são escritas utilizando apenas os conectivos $\neg$ e $\vee$. Esta demonstração utiliza indução no número de conectivos $\neg$ e $\vee$ que ocorrem em H. Considere, então, a notação:

$$Nc(H) = \text{número de conectivos } \neg \text{ e } \vee \text{ em } H.$$

Seja $A(n)$ a asserção:

$$A(n) \equiv \text{ Se } Nc(H) = n, \text{ então } \quad \begin{array}{l} a)\ I[H] = T \Rightarrow B[H, I] \vdash H, \\ b)\ I[H] = F \Rightarrow B[H, I] \vdash \neg H. \end{array}$$

Para demonstrar este lema, vamos provar, utilizando o princípio da indução finita, que $A(n)$ é válida para todo n.

Base da indução. Nesse caso, devemos demonstrar que $A(0)$ é verdadeira.

$A(0) \equiv$ Se $Nc(H) = 0$, então $\quad a)\ I[H] = T \Rightarrow B[H, I] \vdash H,$
$\qquad b)\ I[H] = F \Rightarrow B[H, I] \vdash \neg H.$

Como $Nc(H) = 0$, então H é igual a um símbolo proposicional qualquer $\check{P}$, isto é $H = \check{P}$. Nesse caso devemos demonstrar que:

a) se $I[H] = T$ então $B[H, I] \vdash \check{P}$;

b) se $I[H] = F$ então $B[H, I] \vdash \neg\check{P}$.

Como $\{\check{P}\} \vdash \check{P}$, $\{\neg\check{P}\} \vdash \neg\check{P}$ e $H = \check{P}$, temos que:

a) se $I[H] = T$ então $B[H, I] = \{\check{P}\}$. Logo, $B[H, I] \vdash \check{P}$. Isto é, $B[H, I] \vdash H$;

b) se $I[H] = F$ então $B[H, I] = \{\neg\check{P}\}$. Logo, $B[H, I] \vdash \neg\check{P}$. Isto é, $B[H, I] \vdash \neg H$.

Portanto, $A(0)$ é verdadeira.

Passo da indução. Nessa demonstração, consideramos a segunda forma do princípio da indução, na qual o passo da indução é dado por:

Se $A(j)$ é verdadeira para todo j; $1 \leq j < k$, então $A(k)$ é verdadeira.

Há dois casos a considerar:

Caso 1. Suponha que $H = \neg G$, tal que $Nc(H) = k$. Logo, $Nc(G) < k$. Como $A(j)$ é verdadeira para todo j, $j < k$, então:

a) $I[G] = T \Rightarrow B[G, I] \vdash G$,

b) $I[G] = F \Rightarrow B[G, I] \vdash \neg G$.

Como $B[G, I] = B[H, I]$, então temos:

$$\begin{aligned} I[H] = T \quad &\Rightarrow I[G] = F \\ &\Rightarrow B[G, I] \vdash \neg G \\ &\Rightarrow B[H, I] \vdash \neg G \\ &\Rightarrow B[H, I] \vdash H \end{aligned}$$

Portanto, se $I[H] = T$, então $B[H, I] \vdash H$. Se $I[H] = F$, o raciocínio é análogo.

$$\begin{aligned} I[H] = F \quad &\Rightarrow I[G] = T \\ &\Rightarrow B[G, I] \vdash G \\ &\Rightarrow B[H, I] \vdash G \end{aligned}$$

Utilizando as Proposições 5.3 e 5.4 temos:

$$\vdash G \rightarrow \neg\neg G. \tag{5.5}$$

Aplicando *modus ponens* ao Resultado 5.5 e ao resultado $B[H, I] \vdash G$, concluímos $B[H, I] \vdash \neg\neg G$, ou seja, $B[H, I] \vdash \neg H$. Portanto, se $I[H] = F$, então $B[H, I] \vdash \neg H$.

Caso 2. Suponha que $H = (E \vee G)$ tal que $Nc(H) = k$. Logo, $Nc(E) < k$ e $Nc(G) < k$. Como $A(j)$ é verdadeira para todo $j, j < k$, então:

1. $I[E] = T \Rightarrow B[E, I] \vdash E$.
2. $I[E] = F \Rightarrow B[E, I] \vdash \neg E$.

3. $I[G] = T \Rightarrow B[G, I] \vdash G$.

4. $I[G] = F \Rightarrow B[G, I] \vdash \neg G$.

Como $B[H, I] = \{B[E, I] \cup B[G, I]\}$, então:

$a)I[H] = F \quad \Rightarrow I[E] = F$ e $I[G] = F$

$\Rightarrow B[E, I] \vdash \neg E$ e $B[G, I] \vdash \neg G$

$\Rightarrow B[H, I] \vdash \neg E$ e $B[H, I] \vdash \neg G$

Considerando as Proposições 5.3 e 5.14, obtemos $\vdash (\neg E \rightarrow (\neg G \rightarrow \neg(E \vee G)))$. Aplicando *modus ponens* duas vezes, concluímos que $B[H, I] \vdash \neg(E \vee G)$, isto é, $B[H, I] \vdash \neg H$.

$b)I[H] = T \quad \Rightarrow I[E] = T$ ou $I[G] = T$

Portanto, há dois casos a considerar: $I[E] = T$ ou $I[G] = T$.

$I[E] = T \quad \Rightarrow B[E, I] \vdash E$

$\Rightarrow B[H, I] \vdash E$

Utilizando as Proposições 5.3 e 5.15, obtemos $\vdash (E \rightarrow (E \vee G))$. Em seguida, aplicando *modus ponens*, concluímos que $B[H, I] \vdash (E \vee G)$. Isto é, $B[H, I] \vdash H$. Se $I[G] = T$ a demonstração é análoga. Nesse caso, basta usar a Proposição 5.6. Portanto, como a base e o passo da indução são válidos, utilizando o princípio da indução, a demonstração está concluída. **cqd**■

A demonstração do teorema da completude utiliza o resultado do Lema 5.2. Mas, antes de considerar essa demonstração, seus passos são analisados em um caso mais simples. Nesse caso mais simples, consideramos um tipo de tautologia particular H, que contém apenas os símbolos proposicionais P_1, P_2 e P_3. A fórmula H poderia ser, por exemplo, $H = (P_1 \wedge P_2) \rightarrow (P_3 \vee \neg P_3)$. A tabela-verdade, Tabela 5.1, indica todas as possibilidades de interpretação desses símbolos.

Interpretação	P_1	P_2	P_3
I_1	T	T	T
I_2	T	T	F
I_3	T	F	T
I_4	T	F	F
I_5	F	T	T
I_6	F	T	F
I_7	F	F	T
I_8	F	F	F

Tabela 5.1: Tabela-verdade associada aos símbolos P_1, P_2, P_3.

Conforme a Tabela 5.1, as seguintes bases para a fórmula H são obtidas: $B[H, I_1] = \{P_1, P_2, P_3\}$, $B[H, I_2] = \{P_1, P_2, \neg P_3\}$, $B[H, I_3] = \{P_1, \neg P_2, P_3\}$, $B[H, I_4] = \{P_1, \neg P_2, \neg P_3\}$, $B[H, I_5] = \{\neg P_1, P_2, P_3\}$, $B[H, I_6] = \{\neg P_1, P_2, \neg P_3\}$,

$B[H, I_7] = \{\neg P_1, \neg P_2, P_3\}$, $B[H, I_8] = \{\neg P_1, \neg P_2, \neg P_3\}$,
Como H é uma tautologia, então para qualquer interpretação I_i temos $I_i[H] = T$. Logo, utilizando o Lema 5.2, concluímos que $B[H, I_i] \vdash H$ para toda interpretação I_i. Temos, por exemplo, que:
$B[H, I_1] \vdash H$ ou seja, $\{P_1, P_2, P_3\} \vdash H$
$B[H, I_2] \vdash H$ ou seja, $\{P_1, P_2, \neg P_3\} \vdash H$
As interpretações I_1 e I_2 se diferem apenas no símbolo P_3 e coincidem nos símbolos P_1 e P_2. Quando isso ocorre, tais interpretações são denominadas interpretações complementares em P_3. Demonstramos, a seguir, o resultado: dado um conjunto de hipóteses β, se $\beta \cup \{P\} \vdash H$ e $\beta \cup \{\neg P\} \vdash H$, então $\beta \vdash H$. Observe que os literais complementares P e $\neg P$ são eliminados dos conjuntos de hipóteses iniciais. Portanto, como $\{P_1, P_2, P_3\} \vdash H$ e $\{P_1, P_2, \neg P_3\} \vdash H$, concluímos que $\{P_1, P_2\} \vdash H$. Observe que os literais complementares P_3 e $\neg P_3$ são eliminados dos conjuntos de hipóteses iniciais. Analogamente, considerando outras duas interpretações complementares, I_3 e I_4, temos:
$B[H, I_3] \vdash H$ ou seja, $\{P_1, \neg P_2, P_3\} \vdash H$
$B[H, I_4] \vdash H$ ou seja, $\{P_1, \neg P_2, \neg P_3\} \vdash H$.
Os literais complementares P_3 e $\neg P_3$ também podem ser eliminados dos conjuntos de hipóteses iniciais e obtemos: $\{P_1, \neg P_2\} \vdash H$. Utilizando os resultados $\{P_1, P_2\} \vdash H$ e $\{P_1, \neg P_2\} \vdash H$, e a eliminação dos literais complementares P_2 e $\neg P_2$, temos $\{P_1\} \vdash H$.

A consequência sintática $\{P_1\} \vdash H$ é obtida considerando as bases $B[H, I_1]$, $B[H, I_2]$, $B[H, I_3]$ e $B[H, I_4]$. Repetindo, de forma inteiramente análoga, utilizando as bases $B[H, I_5], B[H, I_6], B[H, I_7]$ e $B[H, I_8]$, concluímos que $\{\neg P_1\} \vdash H$, Utilizando os resultados: $\{P_1\} \vdash H$, e $\{\neg P_1\} \vdash H$, como também a eliminação dos literais complementares P_1 e $\neg P_1$, temos o resultado final: $\vdash H$, ou seja, $\vdash H$. Portanto, considerando que H é uma tautologia, concluímos que $\vdash H$, ou seja, H é um teorema. A demonstração do teorema da completude repete esse processo, considerando uma fórmula H qualquer. Inicialmente, identificamos os pares de interpretações complementares e depois eliminamos literais complementares, até obter um conjunto vazio de hipóteses, concluindo que $\vdash H$.

Demonstração (teorema da completude). Seja H uma tautologia. Demonstramos, a seguir, que H é um teorema, isto é $\vdash H$. Sejam $P_1, \ldots, P_n$ os símbolos proposicionais de H e a Tabela 5.2, a seguir.

Os símbolos proposicionais de H são apresentados na primeira linha da tabela, e nas outras, todas as combinações possíveis de interpretações desses símbolos. Cada linha define uma interpretação diferente para $P_1, \ldots, P_n$, o que totaliza 2^n interpretações diferentes para esses símbolos.[8] Nesse caso, considerando que as linhas da Tabela 5.2 são preenchidas de forma análoga à Tabela 5.1, então na primeira linha da Tabela 5.2 temos apenas o símbolo T e na última, o símbolo F. Isso significa que

[8]A rigor, cada linha da tabela-verdade corresponde a um conjunto infinito de interpretações que coincidem nos símbolos $P_1, \ldots, P_n$. Para simplificar a notação, esse conjunto de interpretações é simplesmente referenciado por um representante.

se I_1 é uma interpretação definida pela primeira linha e I_{2^n} definida pela última linha, então:

$$I_1[P_1] = I_1[P_2] = \ldots = I_1[P_n] = T$$

e

$$I_{2^n}[P_1] = I_{2^n}[P_2] = \ldots = I_{2^n}[P_n] = F.$$

P_1	P_2	....	P_{n-1}	P_n
T	T	....	T	T
T	T	....	T	F
.....	.....	....	.....	.....
F	F	....	F	T
F	F	....	F	F

Tabela 5.2: Tabela-verdade associada aos símbolos $P_1, \ldots, P_n$.

Além disso, os símbolos T e F são escritos na Tabela 5.2, seguindo estas regras. Na coluna de P_1 há 2^{n-1} símbolos T seguidos de 2^{n-1} símbolos F. Na coluna de P_2 há 2^{n-2} símbolos T, seguidos de 2^{n-2} símbolos F, seguidos de 2^{n-2} símbolos T e por fim, seguidos de 2^{n-2} símbolos F. Seguindo tais regras, na última coluna, que corresponde a P_n, os símbolos T e F se alternam, iniciando-se com T. Dado um símbolo P_i, existem vários pares de interpretações I_{c_i} e J_{c_i} tais que:

$$\text{se } j \neq i, \text{ então } I_{c_i}[P_j] = J_{c_i}[P_j] \text{ e}$$
$$\text{se } j = i, \text{ então } I_{c_i}[P_i] = T \text{ e } J_{c_i}[P_i] = F.$$

Considere, por exemplo, o símbolo P_1. Na coluna de P_1 há 2^{n-1} símbolos T seguida de 2^{n-1} símbolos F. Nesse caso, as interpretações I_1 e $I_{2^{n-1}+1}$ são tais que $I_1[P_1] = T$, $I_{2^{n-1}+1}[P_1] = F$ e $I_1[P_j] = I_{2^{n-1}+1}[P_j]$ para todo j tal que $j \neq 1$. Nesse caso, as interpretações I_1 e $I_{2^{n-1}+1}$ são denominadas interpretações complementares de P_1. As interpretações I_{c_i} e J_{c_i} são também chamadas de interpretações complementares de P_i. Observe que, para qualquer símbolo P_i, a metade das interpretações definidas pela Tabela 5.2 interpreta P_i como T e a outra metade como F. Portanto, para qualquer símbolo P_i há 2^{n-1} pares de interpretações complementares. Isso significa que o número de pares de interpretações complementares depende exponencialmente do número de símbolos da fórmula H. Considere I_{c_n} e J_{c_n} duas interpretações complementares de P_n. Logo:

$$B[H, I_{c_n}] = \{P_1', \ldots, P_{n-1}', P_n\}$$
$$B[H, J_{c_n}] = \{P_1', \ldots, P_{n-1}', \neg P_n\}$$

e, para $j < n$:

$$P'_j = P_j \text{ se } I_{c_n}[P_j] = J_{c_n}[P_j] = T$$
$$P'_j = \neg P_j \text{ se } I_{c_n}[P_j] = J_{c_n}[P_j] = F$$

Como $\vDash H$, então $I_{c_n}[H] = J_{c_n}[H] = T$. Utilizando o Lema 5.2, $B[H, I_{c_n}] \vdash H$ e $B[H, J_{c_n}] \vdash H$, isto é:

$$\{P'_1, \ldots, P'_{n-1}, P_n\} \vdash H$$
$$\{P'_1, \ldots, P'_{n-1}, \neg P_n\} \vdash H$$

Utilizando o teorema da dedução, concluímos:

$$\{P'_1, \ldots, P'_{n-1}\} \vdash (P_n \to H)$$
$$\{P'_1, \ldots, P'_{n-1}\} \vdash (\neg P_n \to H)$$

Aplicando a regra de substituição, Proposição 5.3, substituindo A por P_n, C por H e considerando $\beta = \{P'_1, \ldots, P'_{n-1}\}$ na Proposição 5.9, obtemos:

$$\{P'_1, \ldots, P'_{n-1}\} \vdash H$$

Temos que $P'_{n-1} = P_{n-1}$ ou $P'_{n-1} = \neg P_{n-1}$. Se $P'_{n-1} = P_{n-1}$, então

$$\{P'_1, \ldots, P'_{n-2}, P_{n-1}\} \vdash H.$$

Repetindo esse raciocínio, escolhendo duas outras interpretações complementares I'_{c_n} e J'_{c_n} de P_n tais que $I'_{c_n}[P_{n-1}] = J'_{c_n}[P_{n-1}] = F$, obtemos, de forma análoga:

$$\{P'_1, \ldots, P'_{n-2}, \neg P_{n-1}\} \vdash H.$$

Portanto, temos as duas deduções:

$$\{P'_1, \ldots, P'_{n-2}, P_{n-1}\} \vdash H$$
$$\{P'_1, \ldots, P'_{n-2}, \neg P_{n-1}\} \vdash H$$

Utilizando novamente o teorema da dedução e as Proposições 5.3 e 5.9, concluímos:

$$\{P'_1, \ldots\ldots, P'_{n-2}\} \vdash H.$$

Repetindo todo esse processo mais $n - 2$ vezes, concluímos $\vdash H$, isto é, H é um teorema do sistema axiomático $\wp_a$. **cqd**■

5.5.1 Observações sobre o teorema da completude

Em geral, um conceito é semântico quando considera a interpretação das fórmulas. É esse tipo de conceito que é expresso por $\beta \vDash H$. Nesse caso, todas as interpretações que satisfazem β também satisfazem H. Por outro lado, um conceito, em geral, é sintático quando não considera o significado das fórmulas. Se, por exemplo, $\beta \vdash H$, então existe uma prova de H utilizando axiomas e hipóteses de β. Observe, portanto, que $\beta \vdash H$ é um conceito sintático, o qual não considera o significado de H. Os teoremas da correção e da completude são centrais no estudo da Lógica e relacionam fatos do mundo sintático, como $\vdash H$, a fatos do mundo semântico, como $\vDash H$. Os dois teoremas dizem que, no sistema axiomático $\wp_a$:

$$H \text{ é um teorema se, e somente se, } H \text{ é uma tautologia.}$$

Isto é, o sistema axiomático $\wp_a$ é correto e completo e, portanto, os conceitos de consequência sintática e semântica coincidem. Tudo isso é verdade porque é válida em $\wp_a$ a equivalência: $\vDash H \Leftrightarrow \vdash H$. Isso significa que em $\wp_a$ os conceitos de tautologia e teorema coincidem. Além disso, dado um conjunto de fórmulas β, então $\beta \vDash H \Leftrightarrow \beta \vdash H$. Resumindo, como os teoremas da correção e completude valem em $\wp_a$, os fatos a seguir, que são importantes, são verdadeiros.

- Em $\wp_a$ consequências lógicas semântica e sintática são conceitos equivalentes.
- Em $\wp_a$ tautologia e teorema são conceitos equivalentes.

No início, quando começamos a estudar Lógica, é comum imaginar que nada adicionamos ao estudar o método sintático definido em $\wp_a$. Isso acontece devido às equivalências anteriores. Até parece que estamos falando a mesma coisa, mas com palavras diferentes. Mas não é bem assim. As diferenças ficam evidentes a partir do estudo da Lógica de Predicados, considerada no próximo capítulo. Por isso, observamos a seguir alguns aspectos sobre os teoremas da correção e da completude em $\wp_a$, Teoremas 5.2 e 5.3.

1. Na demonstração do teorema da completude, dada uma tautologia H, é demonstrada a existência de uma prova de H, no sistema axiomático $\wp_a$. Entretanto, tal prova não é exibida. Em momento algum da demonstração, explicitamos a sequência de fórmulas que define a prova de H. Esse tipo de demonstração é comum em matemática. Demonstramos a existência de um objeto, sem que ele se torne explícito.

2. Conforme a Definição 5.1, o sistema formal axiomático $\wp_a$ é definido utilizando o alfabeto da Lógica Proposicional, na forma simplificada, Definição 3.14. Como esse alfabeto não contém símbolos de verdade, uma importante questão é responder se a demonstração continua válida se os símbolos de verdade forem adicionados ao alfabeto. Para responder essa questão, devemos analisar a demonstração do teorema da completude, no caso em que a tautologia H contém um símbolo de verdade. Nesse caso, o raciocínio desenvolvido a partir da Tabela 5.2 não é mais correto. Isso, porque entre os símbolos da tabela temos um símbolo de verdade com interpretação fixa. Verifique, na demonstração, esse fato.

3. A demonstração do teorema da completude apresentada neste livro não é correta quando consideramos símbolos de verdade no alfabeto. Na verdade, se os símbolos de verdade são adicionados ao alfabeto, o novo sistema formal não é mais completo.

4. Observe que as duas últimas observações não se aplicam aos métodos semânticos, pois nesses métodos não há diferença alguma em adicionar, ou não, os símbolos de verdade.

5. No final da demonstração,[9] para eliminar os símbolos proposicionais $P_1, \ldots, P_{n-1}, P_n$ da dedução $\{P_1, \ldots.., P_{n-1}, P_n\} \vdash H$ e obter $\vdash H$, repetimos o processo de demonstração n vezes. Entretanto, tal fato não significa que o número global de passos da demonstração seja uma função linear de n. Em outras palavras, a complexidade do processo definido na demonstração não é uma função linear de n. Isso ocorre porque a demonstração considera também escolhas de interpretações complementares na Tabela 5.2, cujo número depende exponencialmente do número de símbolos proposicionais, que é igual a n.
6. São considerados neste capítulo vários teoremas como os da correção e completude em $\wp_a$. Por outro lado, o termo "teorema" também é utilizado para denominar uma fórmula H tal que $\vdash H$ em $\wp_a$. Portanto, o termo "teorema" é utilizado com dois significados diferentes. Um deles está na linguagem-objeto e o outro na metalinguagem.
7. Se H é um teorema, isto é, $\vdash H$, temos uma caracterização sintática de H. Nesse caso, há uma prova de H, sem a utilização de hipóteses. Observe que a caracterização de H é sintática porque o conceito de prova é sintático. Por outro lado, quando se utiliza o termo "teorema" na denominação "teorema da completude", não se tem uma caracterização sintática de fórmulas. Nesse caso, o teorema da completude trata da relação entre propriedades sintáticas e semânticas de uma fórmula H. O teorema diz que se H possui a propriedade semântica $\vDash H$, então também possui a propriedade sintática $\vdash H$. Portanto, o teorema da completude relaciona a sintaxe e a semântica das fórmulas e, nesse sentido, o termo "teorema" está na metalinguagem.
8. A conclusão é que o termo "teorema" é utilizado em dois níveis diferentes e, portanto, com significados diferentes. Na metalinguagem, relacionando a sintaxe e a semântica das fórmulas, como no teorema da completude; e na linguagem-objeto, caracterizando propriedades sintáticas, como em $\vdash H$.

5.6 Consistência no sistema $\wp_a$

Parece que sempre procuramos pela consistência. Ou seja, pela não contradição. Na Lógica também é assim e esse conceito é definido a seguir.

Definição 5.8 (consistência de um sistema axiomático) *Um sistema axiomático é consistente se, e somente se, dada uma fórmula H, não se pode ter $\vdash H$ e $\vdash \neg H$. Isto é, H e $\neg H$ não podem ser teoremas ao mesmo tempo.*

A consistência de um sistema axiomático é uma importante propriedade, pois se um sistema axiomático clássico, como o $\wp_a$, não for consistente, então é possível

[9] O entendimento da análise apresentada neste parágrafo depende de conceitos de Teoria da Computação [Sipser]. O leitor que ainda não conhece tais conceitos pode desconsiderar este parágrafo.

provar qualquer fórmula nesse sistema. E tal fato o trivializa, ou seja: temos um sistema no qual todas as fórmulas são teoremas. É um sistema no qual qualquer fórmula, tautologias e contradições, por exemplo, são teoremas. Se um sistema axiomático não é consistente, podemos provar, nesse sistema, qualquer fórmula. Lembre da curiosa história de Russell a respeito do aluno que pediu ao famoso lógico e filósofo Bertrand Russell que provasse que qualquer coisa segue da contradição. Ela mostra que é possível provar a partir do enunciado "2 + 2 = 5", que "Russell e o Papa são iguais." A prova de Russell, utilizando a notação deste capítulo, é apresentada a seguir:

1.	$\vdash 2+2=5$	hipótese;
2.	$\vdash 2=3$	subtraia 2 de ambos os lados da igualdade;
3.	$\vdash 3=2$	inverta os termos da igualdade;
4.	$\vdash 2=1$	subtraia 1 de ambos os lados da igualdade.

Lembre-se de como Russell concluiu: "O Papa e eu somos dois. Mas, dado que dois é igual a 1, então eu e o Papa somos um. Logo, eu sou igual ao Papa". Pelo teorema da correção em $\wp_a$, as fórmulas contraditórias não são teoremas. Logo, $\wp_a$ é consistente. Pois, caso contrário, se não fosse consistente, não poderia ser correto. A demonstração desse fato é considerada no Teorema 5.4, a seguir.

Teorema 5.4 (consistência) *O sistema axiomático $\wp_a$ é consistente.*

Demonstração. Suponha que $\vdash H$. Logo, pelo teorema da correção, H é uma tautologia e, portanto, $\neg H$ é contraditória. Mas se $\neg H$ é contraditória, então não se pode ter $\vdash \neg H$. Pois, caso ocorresse $\vdash \neg H$, então, novamente pelo teorema da correção, $\neg H$ seria uma tautologia. Mas é impossível ter H e $\neg H$ como tautologias ao mesmo tempo. **cqd**■

Definição 5.9 (consistência de um conjunto de fórmulas) *Um conjunto de hipóteses Γ é consistente se, e somente se, não existe fórmula H tal que $\Gamma \vdash H$ e $\Gamma \vdash \neg H$. Isto é, H e $\neg H$ não podem ser provadas a partir de Γ.*

Teorema 5.5 (consistência e satisfatibilidade) *Um conjunto de fórmulas Γ é consistente se, e somente se, é satisfatível.*

Esquema de demonstração. Esta demonstração é feita considerando o caso mais simples em que o conjunto de fórmulas é finito. No caso mais geral, em que o conjunto de fórmulas pode ser infinito, a demonstração utiliza o teorema da compacidade, que não é considerado neste livro [Enderton], [Dalen], [Shoenfield]. Considere, portanto, Γ é finito $\Gamma = \{A_1, ..., A_n\}$.

Demonstração "$\Rightarrow$". Γ é consistente $\Rightarrow$ Γ é satisfatível. Se Γ é consistente, então não existe H tal que $\Gamma \vdash H$ e $\Gamma \vdash \neg H$. Suponha, por absurdo, que Γ não seja satisfatível. Logo, não existe interpretação I que satisfaça as fórmulas de Γ. Ou seja, para toda interpretação I, temos que $I[(A_1 \land \ldots \land A_n)] = F$. Logo, nesse caso, $(A_1 \land \ldots \land A_n) \to H$ e $(A_1 \land ... \land A_n) \to \neg H$ são tautologias.

Então, utilizando o teorema da completude, concluímos que $\vdash (A_1 \wedge \ldots \wedge A_n) \rightarrow H$ e $\vdash (A_1 \wedge ... \wedge A_n) \rightarrow \neg H$. Como $\Gamma = \{A_1, \ldots, A_n\}$, portanto, (veja exercícios) $\Gamma \vdash A_1 \wedge \cdots \wedge A_n$. Aplicando *modus ponens*, concluímos que $\Gamma \vdash H$ e $\Gamma \vdash \neg H$, o que é um absurdo, pois Γ é consistente. Portanto, Γ é satisfatível.

Demonstração "$\Leftarrow$". Γ é satisfatível $\Rightarrow$ Γ é consistente. Se Γ é satisfatível, existe interpretação I que satisfaz Γ. Suponha, por absurdo, que Γ não seja consistente. Logo, existe H tal que $\Gamma \vdash H$ e $\Gamma \vdash \neg H$. Utilizando o teorema da correção, H e $\neg H$ são consequências lógicas semânticas de Γ. Mas como I satisfaz Γ, então $I[H] = T$ e $I[\neg H] = T$, o que é um absurdo. Portanto, Γ é consistente. **cqd**■

5.7 Exercícios

Observação. Caso você não tenha experiência em demonstrações que utilizam indução finita, desconsidere os exercícios que necessitam de tal princípio para a solução. Isso não comprometerá sua compreensão do resto do livro.

1. Dadas duas fórmulas H e G, demonstre que, se H e G são tautologias, então o resultado da aplicação da regra *modus ponens* a H e G também é uma tautologia. Em outras palavras, se H e G são tautologias e $\dfrac{H,\ G}{E}\ MP$, então E é uma tautologia.
   Nota. A fórmula G é necessariamente do tipo: $G = H \rightarrow E$.
2. Responda, justificando sua resposta.
   (a) Por que uma derivação é um procedimento puramente sintático?
   (b) Seja β um conjunto de fórmulas. Se $\beta \vdash H$, então é possível concluir que H é uma tautologia?
   (c) Seja β um conjunto de fórmulas. Considere que $\beta \vdash H$. Que condições o conjunto de fórmulas β deve satisfazer para que necessariamente H seja uma tautologia?
   (d) Se H é tautologia, então $\beta \vdash H$ para qualquer conjunto β.
3. Considere as afirmações a seguir no sistema $\wp_a$. Demonstre quando forem verdadeiras ou, caso contrário, dê um contraexemplo.
   (a) Se $\beta \vdash H$ e $\beta \subset \varphi$, então $\varphi \vdash H$.
   (b) Se $\beta \vdash H$ e $\beta \supset \varphi$, então $\varphi \vdash H$.
   (c) Se $\beta \vdash H \wedge G$, então $\beta \vdash H$ e $\beta \vdash G$.
   (d) Se $\beta \cup \{H\} \vdash G$, então $\beta \vdash H \rightarrow G$.
   (e) Se $\beta \vdash H$ e $\varphi \vdash H \rightarrow G$, então $\beta \cup \varphi \vdash G$.
   (f) Se $\beta \vdash (P \wedge \neg P)$, então $\beta \vdash H$.
   (g) Se $\beta \vdash (P \wedge \neg P)$, então β é insatisfatível.

(h) Se $\beta \cup \{\neg H\} \vdash (P \land \neg P)$, então $\beta \vdash H$.

(i) Se $\beta \cap \varphi \vdash H$, então $\varphi \vdash H$.

(j) Se $\beta \cup \varphi \vdash H$, então $\varphi \vdash H$.

4. Demonstre a volta do teorema da dedução, forma sintática, Teorema 5.1, ou seja:

   Se $\beta \vdash (A \to B)$, então $\beta \cup \{A\} \vdash B$.

5. Considere o sistema $\wp_a$. Existe um conjunto de fórmulas β tal que $\beta \vdash H$ e $\beta \vdash \neg H$ para alguma fórmula H? Isto é, H e $\neg H$ são consequências lógicas de β?

6. Demonstre em $\wp_a$ que $\vdash (\neg\neg P \to P)$.

7. Demonstre em $\wp_a$ que se $\beta \vdash (A \lor B)$, então $\beta \vdash (B \lor A)$.

8. (Regra de Transitividade) Demonstre em $\wp_a$ que se $\beta \vdash (H_1 \to H_2), \ldots,$ $\beta \vdash (H_{n-1} \to H_n)$, então $\beta \vdash (H_1 \to H_n)$.

9. Considere um sistema axiomático $\wp_b$ em que se tem *modus ponens*, a Regra de Substituição e $\vdash (H \to (H \to G))$. Demonstre que $\wp_b$ é inconsistente.

10. O comprimento de uma prova $\beta \vdash H$ em $\wp_a$ é definido como o número de fórmulas da sequência que define a prova. Se essa prova é $H_1, ..., H_n$, o seu comprimento é igual a n. Isto é, $comp\ [\beta \vdash H] = n$.

    (a) Defina o princípio da indução finita em $\wp_a$, como o princípio da indução no comprimento da prova.

    (b) Demonstre a validade do princípio da indução em $\wp_a$.

11. Considere a regra *modus tollens*:[10] Dadas as fórmulas H e G, a regra de inferência denominada *modus tollens* (MT) é definida pelo seguinte procedimento: tendo $\neg G$ e $(H \to G)$ deduza $\neg H$.

    Demonstre que essa regra mantém a validade das fórmulas. Isto é, se $\neg G$ e $(H \to G)$ são tautologias, então $\neg H$ também é tautologia.

12. Considere a regra de dedução a seguir. Dadas as fórmulas A, B e C, a regra de inferência (OU) é definida pelo procedimento: tendo $(A \lor B \lor C)$, $\neg A$ e $\neg B$, deduza C.

    Demonstre que essa regra mantém a validade das fórmulas. Isto é, se $(A \lor B \lor C)$, $\neg A$ e $\neg B$ são tautologias, então C também é tautologia.

13. Complete a demonstração do teorema da correção em $\wp_a$, utilizando o princípio da indução no comprimento da prova.

14. Repita, passo a passo, a demonstração do teorema da completude em $\wp_a$ no caso particular em que H contém apenas os símbolos proposicionais P_1, P_2, P_3, P_4 e P_5.

[10] *modus tollens* é uma expressão do Latim originada da expressão *"modus tollendo tollens"*, que significa "modo que, negando, nega".

15. Considere $\Gamma = \{A_1, \ldots, A_n\}$. Demonstre que $\Gamma \vdash A_1 \wedge \cdots \wedge A_n$.

16. (a) Como sabemos, a fórmula $(\neg false \vee P)$ é uma tautologia, isto é: $\vDash (\neg false \vee P)$. Utilize as ideias da demonstração do teorema da completude, Teorema 5.3, e verifique se é possível demonstrar que $\vdash (\neg false \vee P)$.

    (b) As ideias da demonstração do teorema da completude, Teorema 5.3, podem ser utilizadas no caso em que H contém o símbolo de verdade $false$?

    (c) Justifique informalmente a conclusão: se for adicionado ao alfabeto do sistema $\wp_a$ o símbolo de verdade $false$, a prova do teorema da completude do sistema axiomático resultante não pode ser considerada.[11]

[11] Na verdade, o sistema se torna incompleto. Isto é, com a adição do símbolo de verdade, não há prova de sua completude.

CAPÍTULO 6

A LINGUAGEM DA LÓGICA DE PREDICADOS

6.1 Introdução

Na primeira parte deste livro, estudamos a Lógica Proposicional, em que as fórmulas são veritativo-funcionais, ou seja, suas interpretações não dependem da estrutura interna de suas proposições, mas unicamente do modo como se combinam. O termo "veritativo-funcional" expressa, exatamente, esta propriedade: a verdade de uma sentença depende funcionalmente das suas partes. Em outras palavras, a interpretação da fórmula $P \vee Q$, por exemplo, depende apenas da interpretação e do modo como os símbolos proposicionais P e Q se combinam. E, nessa análise, não é necessário saber a constituição interna das proposições representadas pelos símbolos P e Q. Utilizando as proposições, muito pode ser feito, como, por exemplo, o que foi apresentado na primeira parte do livro. É possível introduzir conceitos importantes como sintaxe, semântica, prova, correção, completude etc. Entretanto, há um grupo fundamental de sentenças, ou argumentos, cuja interpretação é, também, determinada pela estrutura interna de seus enunciados simples. Esses tipos de argumentos são denominados silogismos categóricos e foram inicialmente estudados por Aristóteles. Sua análise é iniciada considerando os enunciados categóricos que tradicionalmente possuem as formas a seguir [Salmon]:

- "todos os alunos são inteligentes";
- "nenhum aluno é inteligente";

- "alguns alunos são inteligentes";
- "alguns alunos não são inteligentes".

Para interpretar enunciados categóricos como esses, é necessário interpretar os quantificadores "todos", "nenhum" e "algum". Considere, por exemplo, outra sentença: "Todos os alunos são inteligentes." Nesse caso, para saber se ela é verdadeira, ou falsa, é necessário, pelo menos, saber de quais alunos estamos falando. Se estamos considerando apenas o conjunto dos alunos campeões da Olimpíada Nacional de Matemática, certamente temos uma sentença verdadeira. Mas, se considerarmos todos os alunos do país, talvez não tenhamos o mesmo resultado. Portanto, para interpretar uma sentença como essa, é necessário saber algo que está implicitamente representado na própria estrutura da sentença. Em outras palavras, quando dizemos que todos os alunos são inteligentes, devemos deixar claro de que alunos estamos falando.

Este capítulo inicia a segunda parte do livro na qual consideramos o estudo da Lógica de Predicados e nela, estudamos sentenças como os enunciados categóricos. Os passos a serem seguidos são análogos àqueles considerados na primeira parte. Inicialmente, é definida a linguagem da Lógica de Predicados, analisada do ponto de vista sintático e semântico. Também são considerados alguns mecanismos que verificam fórmulas válidas. Então, para começar, analisamos neste capítulo os fundamentos da linguagem da Lógica de Predicados, cuja definição é semelhante à definição de outras linguagens. Como na linguagem da Lógica Proposicional, o alfabeto é definido inicialmente, e, em seguida, os outros elementos da linguagem. O resultado é uma linguagem mais rica que a da Lógica Proposicional, pois, além de conter os seus objetos, a linguagem da Lógica de Predicados possui quantificadores, símbolos funcionais e de predicados. Nesse sentido, a Lógica de Predicados é uma extensão da Lógica Proposicional, o que lhe confere maior poder de representação.

6.2 Alfabeto da Lógica de Predicados

A linguagem da Lógica de Predicados contém tudo que a linguagem da Lógica Proposicional contém e algo mais. Isso, porque a linguagem da Lógica de Predicados é uma extensão da linguagem da Lógica Proposicional. E sendo uma extensão, há vários tipos de sentenças, que não possuem representações adequadas na Lógica Proposicional, mas podem ser representadas, um pouco melhor, na Lógica de Predicados. Considere, por exemplo, as declarações:

- Todo aluno de Computação é inteligente. Zé é aluno de Computação. Logo, Zé é inteligente;
- A adição de dois números ímpares quaisquer é um número par;
- Dado um número qualquer, existe um número primo maior que ele.

A dificuldade em representar tais argumentos na Lógica Proposicional se deve às quantificações indicadas pelas palavras: "todo," "qualquer" e "existe". A linguagem da Lógica de Predicados estende a linguagem da Lógica Proposicional possibilitando a representação desses tipos de quantificação. Além disso, são considerados também funções, predicados e variáveis, de forma análoga ao Cálculo Diferencial. Assim, a Lógica de Predicados é obtida inicialmente pela extensão do alfabeto da Lógica Proposicional. Seguindo essa mesma ideia, a Lógica Proposicional também pode ser estendida de forma diferente da que consideramos aqui, ou, ainda, substituída, obtendo outras Lógicas. Isto é, há várias formas de extender, ou substituir, a Lógica Proposicional, como, por exemplo:

1. Lógicas Complementares são aquelas cujo objetivo é estender a Lógica Clássica. Um exemplo de Lógica Complementar é a Lógica Temporal.
2. Lógicas Alternativas ou heterodoxas têm como objetivo substituir a Lógica Proposicional Clássica. Exemplos de Lógicas Alternativas são as Lógicas Polivalentes, em que as sentenças declarativas podem ser interpretadas diferentemente de T e F.

Mas, no nosso caso, tais fatos são apenas informações. A Lógica de Predicados é uma extensão clássica da Lógica Proposicional e seu alfabeto é definido pelo conjunto de símbolos descritos a seguir.

Definição 6.1 (alfabeto da Lógica de Predicados) *O alfabeto da Lógica de Predicados é constituído por:*

1. *símbolos de pontuação: (,);*
2. *um conjunto enumerável de símbolos para variáveis:* $x, y, z, w, x_1, y_1, \dots$;
3. *um conjunto enumerável de símbolos para funções:* $f, g, h, f_1, g_1, h_1, f_2, g_2, \dots$;
4. *um conjunto enumerável de símbolos para predicados:* $p, q, r, p_1, q_1, r_1, p_2, q_2, \dots$;
5. *conectivos:* $\neg, \vee, \forall, \exists$.

Associado a cada símbolo para função ou predicado, temos um número inteiro não negativo k*. Esse número indica a aridade, ou seja, o número de argumentos da função ou predicado.*

Como podemos observar, o alfabeto da linguagem da Lógica de Predicados é uma extensão do alfabeto da Lógica Proposicional, Definição 3.14. Ele contém os mesmos símbolos de pontuação e parte dos conectivos. Mas, onde estão os símbolos proposicionais? Espere um pouco. Como analisamos mais adiante, eles estão representados, implicitamente, no conjunto dos símbolos para predicados. Além dos símbolos de pontuação e dos conectivos, o alfabeto da Lógica de Predicados contém conjuntos infinitos e enumeráveis de símbolos para funções, predicados e variáveis.

Variáveis. Os símbolos para variáveis formam um novo conjunto, o que não ocorre na Lógica Proposicional. Como é visto mais adiante, as variáveis têm um papel importante na Lógica de Predicados. Isso não é uma surpresa, pois usamos, com frequência, as variáveis em Matemática, Computação e muitas outras áreas. Em programação em Lógica, por exemplo, as variáveis são utilizadas na determinação das respostas dos programas [Casanova], [Chang], [Lloyd].

Variáveis e metavariáveis. Na Lógica Proposicional, o símbolo $\breve{P}$ é utilizado como uma metavariável, o que significa que ele representa qualquer símbolo proposicional. Como foi enfatizado anteriormente, $\breve{P}$ não é um símbolo da linguagem da Lógica Proposicional, mas sim um símbolo da metalinguagem. O símbolo $\breve{P}$ é utilizado para representar qualquer símbolo proposicional do conjunto $\{P, Q, R, J, P_1, Q_1, R_1, S_1, P_2, Q_2, \ldots\}$. Na Lógica de Predicados também há metavariáveis. Só que, nesse caso, o alfabeto também contém suas próprias variáveis. Isso significa que na Lógica de Predicados há variáveis que ocorrem na linguagem-objeto e outras que ocorrem na metalinguagem. As variáveis da linguagem são denotadas por $x, y, z, w, x_1, y_1, \ldots$.

Notação. Como na Lógica Proposicional, utilizamos um pequeno risco acima de um símbolo para indicar que ele representa uma metavariável. Nesse sentido, $\breve{x}, \breve{p}$ e $\breve{f}$ representam metavariáveis de variáveis, símbolos proposicionais, de predicados e de funções.

Predicados. Observe inicialmente que os símbolos para predicados não ocorrem na Lógica Proposicional. Eles são utilizados para representar propriedades e relações entre objetos. Ao dizer, por exemplo, que Maria é bonita, temos que "bonita" é uma propriedade de Maria. Tal fato pode ser representado, na Lógica de Predicados, por $p(x)$. Nesse caso, de uma maneira informal, podemos dizer:

$p(x)$ é verdadeiro se, e somente se, x é bonita.

Dessa forma, o símbolo de predicado p é utilizado para representar a propriedade de ser bonita. E quando x é interpretado como sendo Maria, o resultado da interpretação da sentença é verdadeiro. Os símbolos para predicados também podem ser utilizados para expressar relações entre objetos. A relação "irmão", por exemplo, pode ser representada por $q(x, y)$. Neste caso, de maneira informal, dizemos que:

$q(x, y)$ é verdadeiro se, e somente se, a pessoa x é irmã de y.

Funções. Como os símbolos para predicados, os símbolos para funções também não ocorrem na Lógica Proposicional. Na Lógica, os símbolos para função têm utilização análoga àquela que ocorre na Aritmética. Isto é, não há diferença alguma entre as funções da Lógica e as da Aritmética. E, certamente, já conhecemos bastante sobre o conceito de função. Na Lógica, na Matemática e em Computação os conceitos de função e de predicado são fundamentais. Em Computação, por exemplo, há linguagens que se baseiam nesses conceitos, como as linguagens funcionais *SCHEME* e *LISP* [Winston], [Dybvig], e as linguagens em Lógica, [Casanova], [Chang] e [Lloyd].

Símbolos para constantes. Se temos símbolos para funções, temos as constantes. Isso porque cada símbolo para função possui um número k, não negativo, associado, que representa sua aridade. E quando $k = 0$, temos uma função com zero argumento. Nesse caso, como no Cálculo por exemplo, as funções com zero argumento, ou aridade nula, representam as constantes. Ou seja, elas são as funções constantes.

Notação. Os símbolos para funções zero-árias, com aridade nula, são denominados constantes. Elas são representadas por $a, b, c, a_1, b_1, c_1, a_2, b_2, \ldots$ Como há enumeráveis símbolos para funções com aridade zero, há também um conjunto infinito e enumerável de constantes.

Símbolos proposicionais. Cada símbolo para predicado possui um número k, não negativo, associado, que representa sua aridade. Quando $k = 0$, temos um predicado com zero argumento. E os predicados com aridade zero representam os símbolos proposicionais, que ocorrem no alfabeto da Linguagem Proposicional.

Notação. Os símbolos para predicados zero-ários são denominados símbolos proposicionais. Como na Lógica Proposicional, eles são representados por $P, Q, R, S, P_1, Q_1, R_1, S_1, P_2, Q_2, \ldots$. Observe que há um conjunto infinito e enumerável de símbolos proposicionais.

Conectivos. O conjunto dos conectivos contém $\neg$ e $\vee$, que pertencem à versão simplificada do alfabeto da Lógica Proposicional, Definição 3.14. Além desses conectivos, há também $\forall$ e $\exists$, que representam os quantificadores universal, para todo, e existencial, existe, respectivamente. Como é analisado mais adiante, tais quantificadores ampliam, em muito, o poder de representação da Lógica de Predicados, como também a complexidade das demonstrações e dos resultados.

Uma maneira informal de ver o quantificador universal. Na nossa linguagem cotidiana, utilizamos, com frequência, os quantificadores universal e existencial.

- Todo homem é mortal.
- Todos os homens são mortais.
- Os homens são mortais.
- Homens são sempre mortais.
- Somente mortais é que são homens.
- Apenas mortais é que são homens.
- Só mortais é que são homens.
- Se algo é homem, então é mortal.

Leia com atenção e verifique que todas essas sentenças têm o mesmo sentido. De maneira informal, se consideramos que a variável x representa homens e o predicado $q(x)$ é verdadeiro quando x é mortal, então as sentenças podem ser representadas por $(\forall x)q(x)$. Nesse caso, o quantificador $(\forall x)$ expressa a universalização de x, no sentido que todo, para todo, qualquer que seja, todos ou cada x, sem exceção, satisfaz $p(x)$. Mas cuidado! A tradução de sentenças que utilizam quantificadores universais para a Lógica não deve ser feita às pressas. Por exemplo, o par de sentenças a seguir não expressa as mesmas ideias:

- Somente os mortais são homens.

- Somente mortais é que são homens.

Da mesma forma, o par de sentenças a seguir não expressa as mesmas ideias:

- Todos os homens são mortais.
- Todos os mortais são homens.

Uma maneira informal de ver o quantificador existencial. Considere as sentenças:

- Existe homem inteligente.
- Há um homem inteligente.
- Há pelo menos um homem inteligente.
- Há homens inteligentes.
- Algum homem é inteligente.
- Alguns homens são inteligentes.

Verifique que todas essas sentenças têm o mesmo sentido. De maneira informal, se consideramos que a variável x representa homens e o predicado $q(x)$ é verdadeiro quando x é inteligente, então as sentenças podem ser representadas por $(\exists x)q(x)$. Nesse caso, o quantificador $(\exists x)$ expressa que existe um, alguns, algum, um ou pelo menos um x, que representa homem e que satisfaz $p(x)$.

Os conectivos $\wedge, \rightarrow$ e $\leftrightarrow$. Na linguagem da Lógica de Predicados, os conectivos $\wedge$, $\rightarrow$ e $\leftrightarrow$ são definidos a partir de $\neg$ e $\vee$, conforme é indicado na Lógica Proposicional.

6.3 Fórmulas da Lógica de Predicados

Após a definição do alfabeto, seguimos a mesma orientação do Capítulo 1. O próximo passo é definir as fórmulas. Mas na linguagem da Lógica de Predicados ocorrem vários elementos básicos necessários à definição de fórmula. Inicialmente, observe que na língua portuguesa há sentenças cuja interpretação é um valor de verdade e outras um objeto. A sentença declarativa "A capital de Minas Gerais é Belo Horizonte" é uma proposição, cuja interpretação resulta em um valor de verdade. Nesse caso, o valor de verdade é verdadeiro, pois Belo Horizonte é realmente a capital de Minas Gerais. Por outro lado, o resultado da interpretação da sentença "Capital do Brasil" não é um valor de verdade. O resultado é Brasília, que é um objeto. Em geral, na linguagem da Lógica de Predicados, as sentenças que representam objetos do domínio são os termos. Portanto, nesse sentido a sentença "Capital do Brasil" é um termo. Por outro lado, as fórmulas representam sentenças declarativas que podem ser interpretadas como verdadeiras ou falsas.

Os termos, átomos e fórmulas são definidos a seguir. A interpretação de termos, átomos e fórmulas da linguagem da Lógica de Predicados é analisada nos capítulos posteriores.

Definição 6.2 (termo) *O conjunto dos termos da linguagem da Lógica de Predicados é o menor conjunto que satisfaz as regras a seguir:*

1. *as variáveis são termos;*
2. *se* $t_1, t_2, \ldots, t_n$ *são termos e* $\breve{f}$ *é um símbolo para função n-ária, então* $\breve{f}(t_1, t_2, \ldots, t_n)$ *é um termo.*

Conforme a Definição 6.2, as variáveis são termos. Isso ocorre, pois, como veremos, os resultados das interpretações das variáveis representam objetos. Da mesma forma, o resultado da aplicação de uma função a um conjunto de termos é um termo. Isto é, a interpretação do resultado de uma função aplicada a outros objetos é um objeto. Além disso, observe que na Definição 6.2, o símbolo $\breve{f}$, com tracinho em cima, é uma metavariável que representa um símbolo para função qualquer.

Exemplo 6.1 (termos) A seguir, relacionamos algumas expressões que são termos e outras que não o são.

1. A variável x é um termo.
2. A constante a é um termo, pois é uma função zero-ária. Nesse caso, temos uma função aplicada a zero termo.
3. Se f é uma função binária, então $f(x, a)$ é um termo, pois x e a são termos. A aplicação de uma função binária f aos termos x e a é um termo. Observe que se f não for uma função binária, então $f(x, a)$ não é um termo.
4. Sejam g e f funções ternária e binária respectivamente. Nesse caso, $g(y, f(x, a), c)$ é um termo. Isso ocorre porque os argumentos da função g, em $g(y, f(x, a), c)$, são termos, e a aplicação da função g a esses termos também é um termo.
5. Como f é binária, então a concatenação de símbolos $f(y, x, z)$ não é um termo, pois f deve conter exatamente dois argumentos.

■

Nota. Se declaramos, por exemplo, que $h(x, y, z)$ é um termo, então, nesse caso, é considerado implicitamente que h é ternária.

Exemplo 6.2 (termos) Este exemplo considera, de maneira informal, funções e constantes que representam elementos da aritmética, como funções de adição, de subtração e números naturais. A rigor, os símbolos que representam tais funções não pertencem ao alfabeto da Lógica de Predicados, conforme Definição 6.1, que contém apenas símbolos para função pertencentes ao conjunto $\{f, g, h, \ldots\}$. Mas, para fins didáticos, consideremos, por um instante, que os símbolos $+$ e $-$ e as constantes $1, 2, 3, \ldots$ também pertencem ao alfabeto da Lógica de Predicados. Isso significa que $+$ e $-$, por exemplo, são iguais a símbolos funcionais binários considerados na Definição 6.1. Temos que:

1. x, 9, y, 10 são termos, pois variáveis e constantes são termos.
2. $+(4, 8)$ é um termo, pois a função $+$ aplicada a dois termos é um termo. Nesse caso, o resultado é interpretado como sendo igual a 13, isto é, o resultado é um novo termo.
3. $+(-(8, 7), 3)$ também é um termo. A adição aplicada aos termos $-(8, 7)$ e 3 é um termo. O resultado é interpretado como sendo igual a 4, que é um termo.

Observe que, neste exemplo, a notação utilizada para as funções é a prefixa. As equivalências entre notação prefixa e infixa são: $+(5, 8)$ corresponde a $(5 + 8)$, e $-(8, 7)$ corresponde a $(8 - 7)$. ■

Nem tudo são objetos na interpretação dos elementos da linguagem da Lógica de Predicados. Há, também, as sentenças cujas interpretações são valores de verdade. Elas são representadas por fórmulas declarativas que podem ser interpretadas como verdadeiras ou falsas. Os átomos, que representam expressões cuja interpretação é em valor de verdade, são definidos a seguir.

Definição 6.3 (átomo) *O conjunto dos átomos da linguagem da Lógica de Predicados é o menor conjunto que satisfaz as regras a seguir:*

1. *os símbolos proposicionais são átomos;*
2. *se $t_1, t_2, \ldots, t_n$ são termos e $\breve{p}$ é um símbolo para predicado n-ário, então, $\breve{p}(t_1, t_2, ..., t_n)$ é um átomo.*

Observe que na Definição 6.3, $\breve{p}$ é uma metavariável que representa um símbolo para predicado qualquer.

Exemplo 6.3 (átomos) Este exemplo considera um conjunto de átomos. Alguns deles são construídos a partir de termos definidos em exemplos anteriores.

1. O símbolo proposicional P é um átomo, pois é um predicado zero-ário. Temos, nesse caso, um predicado aplicado a zero termo.
2. Se p é um predicado binário, então $p(f(x, a), x)$ é um átomo. Observe que x, a e $f(x, a)$ são termos. A aplicação de p aos termos $f(x, a)$ e x é um átomo. Observe também que se p não for binário, então a expressão anterior não é um átomo.
3. $q(x, y, z)$ é um átomo. Nesse caso, consideramos implicitamente que q é ternário.
4. Os termos não são átomos. Por exemplo, $g(y, f(x, a), c), f(y, x, z)$ e $h(x, y, z)$ não são átomos.
5. Os átomos não são termos. Por exemplo, $P, p(f(x, a), x)$ e $q(x, y, z)$ não são termos.

■

Nota. Se declaramos, por exemplo, que $p(x, y, z)$ é um átomo, então, nesse caso, é considerado implicitamente que p é ternário.

Exemplo 6.4 (átomos) Este exemplo, de maneira informal, retoma a notação aritmética em que são considerados os predicados "maior que" e "diferente", representados pelos símbolos $>$ e $\neq$, respectivamente. A rigor, tais símbolos não pertencem ao alfabeto da Lógica de Predicados, conforme Definição 6.1, que contém apenas símbolos de predicado pertencente ao conjunto $\{p, q, r, \ldots\}$. Mas, para fins didáticos, novamente, é considerado que os predicados $>$ e $\neq$ também pertencem ao alfabeto da Lógica de Predicados, sendo iguais a símbolos de predicado binários considerados na Definição 6.1.

1. A expressão aritmética $>(+(5, 8), 3)$ representa um átomo. Isso porque $+(5, 8)$ e 3 são termos. E o predicado $>$ aplicado a esses termos é um átomo. Nesse caso, o resultado da interpretação da aplicação do predicado $>$ aos termos $+(5, 8)$ e 3 é igual ao valor de verdade T.
2. A expressão aritmética $\neq (-(8, 7), 3)$ é um átomo. A desigualdade é um predicado. A aplicação de $\neq$ aos termos $-(8, 7)$ e 3 é um átomo. O resultado da interpretação deste átomo é igual ao valor de verdade T. ■

Observe que, nos Exemplos 6.1 e 6.2, os resultados das interpretações dos termos são objetos. Por outro lado, nos Exemplos 6.3 e 6.4 os resultados das interpretações dos átomos são valores de verdade. Agora, estamos prontos para definir as fórmulas da Lógica de Predicados. A construção dessas fórmulas é feita a partir da concatenação de átomos e conectivos. Entretanto, como ocorre na Lógica Proposicional, não é qualquer concatenação de símbolos que é uma fórmula.

Definição 6.4 (fórmula) *O conjunto das fórmulas da linguagem da Lógica de Predicados é o menor conjunto que satisfaz as regras a seguir:*

1. *Todo átomo é uma fórmula.*
2. *Se H é uma fórmula, então $(\neg H)$ é uma fórmula.*
3. *Se H e G são fórmulas, então $(H \vee G)$ é uma fórmula.*
4. *Se H é uma fórmula e $\breve{x}$ uma variável, então $((\forall \breve{x})H)$ e $((\exists \breve{x})H)$ são fórmulas.*

Na definição anterior, cada item define uma regra para construção de fórmulas, a partir de fórmulas mais simples. As fórmulas mais elementares, que são os átomos, são consideradas inicialmente. Em seguida, utilizando os conectivos $\neg, \vee, \forall$ e $\exists$, obtemos fórmulas mais complexas. Como o conjunto de conectivos $\neg, \vee$ é completo, então também é possível obter fórmulas com os conectivos $\wedge, \rightarrow, \leftrightarrow$.

Notação. De forma análoga à Lógica Proposicional, as metavariáveis A, B, C, D, E, H, com possíveis subíndices, também representam fórmulas da Lógica de Predicados. Também, de forma análoga, temos os esquemas de fórmulas, que são fórmulas formadas a partir de metavariáveis, como, por exemplo: $((\exists \breve{x})H)$. Lembre que os esquemas de fórmulas se transformam em fórmulas quando as metavariáveis são substituídas por símbolos e fórmulas do alfabeto da Lógica.

Exemplo 6.5 (construção de fórmulas)

1. Os átomos $p(x)$, R e $q(x, a, z)$ são fórmulas.
2. Como R e $p(x)$ são fórmulas, obtemos a fórmula $((\neg p(x) \vee R))$. Conforme é analisado na primeira parte do livro, essa fórmula pode ser denotada como $(p(x) \rightarrow R)$. Seguindo esse raciocínio, são construídas fórmulas utilizando os conectivos $\wedge, \rightarrow$ e $\leftrightarrow$.
3. Como $(p(x) \rightarrow R)$ é uma fórmula, então $((\forall x)(p(x) \rightarrow R))$ também é uma fórmula.
4. Dadas as fórmulas anteriores, então $(((\forall x)(p(x) \rightarrow R)) \rightarrow q(x, a, z))$ também é uma fórmula.

Seguindo o raciocínio apresentado neste exemplo, infinitas, mas enumeráveis, fórmulas podem ser obtidas. ■

Definição 6.5 (expressão) *Uma expressão da Lógica de Predicados é um termo ou uma fórmula.*

6.4 Correspondência entre quantificadores

Conforme é analisado na Lógica Proposicional, os conectivos $\wedge$,$\rightarrow$ e $\leftrightarrow$ podem ser definidos a partir dos conectivos $\neg$ e $\vee$. Analogamente, é possível definir o quantificador existencial $\exists$ a partir do quantificador universal $\forall$, e vice-versa. Isso significa que o alfabeto da Lógica de Predicados pode ser simplificado, considerando apenas os conectivos $\neg$, $\vee$,$\forall$. Para ver como ocorre essa correspondência, considere, inicialmente, a afirmação: "Existe aluna de Computação que é bonita". Além disso, de maneira informal, suponha que o universo das pessoas consideradas seja o conjunto das alunas de Computação e ninguém mais. Suponha, também, que:

$p(x)$ é verdadeiro se, e somente se, x é bonita

Nesse sentido, a afirmação pode ser representada na Lógica de Predicados pela fórmula $(\exists x)p(x)$, na qual $(\exists x)$ é o quantificador existencial "existe x", onde x é aluna de computação. Se a afirmação anterior é verdadeira, então $(\forall x)\neg p(x)$ é falsa. Observe que esta fórmula representa a afirmação: "Toda aluna de Computação é feia". Mas, se $(\forall x)\neg p(x)$ é falsa, então a sua negação: $\neg((\forall x)\neg p(x))$ é verdadeira. Isto é, é falso que toda aluna de Computação é feia, ou seja, pelo menos uma aluna é bonita. Portanto, informalmente, as afirmações a seguir são equivalentes:

1. $(\exists x)p(x)$, que é interpretada como: "existe aluna de Computação que é bonita."
2. $\neg(\forall x)\neg p(x)$, que é interpretada como: "é falso que toda aluna de Ciência da Computação é feia."

Seguindo esse raciocínio, o quantificador $\forall$ é definido a partir de $\exists$.

Definição 6.6 (correspondência entre quantificadores) *Considere uma fórmula H e uma variável $\breve{x}$. Os quantificadores existencial $\exists$ e universal $\forall$ se relacionam pelas correspondências:*

1. $((\forall\breve{x})H)$ *denota* $\neg((\exists\breve{x})(\neg H))$;
2. $((\exists\breve{x})H)$ *denota* $\neg((\forall\breve{x})(\neg H))$.

Qualquer um dos quantificadores pode ser definido a partir do outro, utilizando as correspondências da definição anterior. O quantificador existencial, por exemplo, pode ser definido a partir da correspondência entre as fórmulas $((\exists\breve{x})H)$ e $\neg((\forall\breve{x})(\neg H))$. Se for essa a correspondência utilizada, então o conectivo $\exists$ pode ser retirado do alfabeto da Lógica de Predicados, Definição 6.1, obtendo um alfabeto simplificado da Lógica de Predicados.

6.5 Símbolos de Pontuação

Analogamente às simplificações da Lógica Proposicional, os parênteses das fórmulas da Lógica de Predicados podem ser omitidos quando não há problemas sobre suas interpretações. As fórmulas também podem ser escritas utilizando várias linhas para uma melhor leitura. Na Lógica de Predicados também há uma ordem de precedência entre os conectivos que possibilita a simplificação das fórmulas. Nesse caso, a ordem de precedência dos conectivos $\neg, \vee, \wedge, \rightarrow$ e $\leftrightarrow$ é a mesma considerada na Lógica Proposicional, e os conectivos $\forall$ e $\exists$ possuem precedências equivalentes.

Definição 6.7 (ordem de precedência) *Na Lógica de Predicados, a ordem de precedência dos conectivos é a seguinte:*

1. *maior precedência:* $\neg$;
2. *precedência intermediária superior:* $\forall$, $\exists$;
3. *precedência intermediária inferior:* $\rightarrow$, $\leftrightarrow$;
4. *precedência inferior:* $\vee$, $\wedge$.

O exemplo a seguir mostra como a ordem de precedência dos conectivos é utilizada para simplificar as fórmulas, retirando símbolos de pontuação.

Exemplo 6.6 (precedência) Este exemplo considera a simplificação de uma fórmula pela eliminação de símbolos de pontuação. Considerando a ordem de precedência dos conectivos, Definição 6.7, a concatenação de símbolos

$$(\forall x)(\exists y)p(x,y) \rightarrow (\exists z)\neg q(z) \wedge r(y)$$

representa a fórmula

$$((((\forall x)((\exists y)p(x,y))) \rightarrow ((\exists z)(\neg q(z))) \wedge r(y)). \blacksquare$$

6.6 Características Sintáticas das Fórmulas

Como na Lógica Proposicional, algumas propriedades semânticas podem ser definidas para as expressões da Lógica de Predicados. Dada uma expressão, as partes que a compõem possuem denominações especiais, conforme a definição a seguir.

Definição 6.8 (subtermo, subfórmula, subexpressão) *Os elementos a seguir definem as partes de um termo ou fórmula E:*

1. *Se $E = \breve{x}$,*
   então a variável $\breve{x}$ é um subtermo de E;
2. *Se $E = \breve{f}(t_1, t_2, \ldots, t_n)$, então, $\breve{f}(t_1, t_2, \ldots, t_n)$ e t_i, para todo i, são subtermos de E;*
3. *Se t_1 é subtermo de t_2 e t_2 é subtermo de E, então t_1 é subtermo de E;*
4. *Se $E = (\neg H)$ então H e $(\neg H)$ são subfórmulas de E;*
5. *Se E é uma das fórmulas $(H \vee G)$, $(H \wedge G)$, $(H \rightarrow G)$ ou $(H \leftrightarrow G)$,*
   então H, G e E são subfórmulas de E;
6. *Se $E = ((\forall \breve{x})H)$, então H e $((\forall \breve{x})H)$ são subfórmulas de E;*
7. *Se $E = ((\exists \breve{x})H)$, então H e $((\exists \breve{x})H)$ são subfórmulas de E;*
8. *Se H_1 é subfórmula de H_2 e H_2 é subfórmula de E,*
   então H_1 é subfórmula de E;
9. *Todo subtermo ou subfórmula é também uma subexpressão.*

Dados um termos t e uma expressão E, então t é um subtermo de E se t é uma parte de E que é termo. Nesse caso, t pode ser igual ou diferente de E. Se for diferente, t é um subtermo próprio de E. Da mesma forma, dadas duas fórmulas G e H, então G é uma subfórmula de H se G é uma parte de H que é fórmula. Como no caso dos termos, G pode ser igual ou diferente de H. Se for diferente, G é uma subfórmula própria de H.

Exemplo 6.7 (subfórmula) Considere a fórmula:

$$H = (((\forall x)p(x)) \rightarrow (p(x)) \wedge ((\forall y)r(y))).$$

A fórmula $p(x)$ é uma subfórmula de H que ocorre duas vezes em H. As outras subfórmulas de H são: H, $(\forall x)p(x)$, $(\forall y)r(y)$, $r(y)$, e $((\forall x)p(x)) \rightarrow (p(x)$. ■

Comprimento de uma fórmula. O comprimento de uma fórmula da Lógica de Predicados é definido a seguir. Como na Lógica Proposicional, esse conceito é utilizado na demonstração de inúmeros resultados, principalmente aqueles que utilizam o princípio da indução finita.

Definição 6.9 (comprimento de uma fórmula) *Dada uma fórmula H, da Lógica de Predicados, o comprimento de H, denotado por $comp[H]$, é definido como se segue:*

1. *se H é um átomo, então $comp[H] = 1$;*
2. $comp[\neg H] = comp[H] + 1$;
3. $comp[H \vee G] = comp[H] + comp[G] + 1$;
4. $comp[H \wedge G] = comp[H] + comp\ [G] + 1$;
5. $comp[H \rightarrow G] = comp[H] + comp[G] + 1$;
6. $comp[H \leftrightarrow G] = comp[H] + comp[G] + 1$;
7. *se $H = (\forall \check{x})G$, então $comp[(\forall \check{x})G] = 1 + comp[G]$;*
8. *se $H = (\exists \check{x})G$, então $comp[(\exists \check{x})G] = 1 + comp[G]$.*

Para simplificar a definição do comprimento de uma fórmula, consideramos que todos os conectivos $\neg, \vee, \wedge, \rightarrow, \leftrightarrow, \forall$ e $\exists$ fazem parte do alfabeto. Observe que isso é um abuso de notação, pois o alfabeto pode ser simplificado considerando apenas os conectivos $\neg, \vee$ e $\forall$.

Exemplo 6.8 (comprimento de uma fórmula) Este exemplo considera duas fórmulas e seus respectivos comprimentos.

- Se $H = (p(x) \rightarrow q(y))$, então $comp[H] = 1 + comp[p(x)] + comp[q(y)] = 3$.
- Se $G = (\forall x)p(x) \leftrightarrow (\exists y)q(y)$, então $comp[G] = 1 + comp[(\forall x)p(x)] + comp[(\exists y)q(y)] = 1 + 1 + comp[p(x)] + 1 + comp[q(y)] = 5$. ■

6.7 Classificações de variáveis

As variáveis que ocorrem nas fórmulas da Lógica de Predicados possuem várias classificações. E entender essa classificação é importante para interpretar tais fórmulas. Nesta seção estudamos essa classificação e no próximo capítulo o uso dela para compreender a interpretação de fórmulas com variáveis. Nas fórmulas da Lógica de Predicados, as variáveis podem ocorrer na forma livre ou ligada. Além disso, dependendo da classificação das variáveis de uma fórmula, temos a classificação da fórmula propriamente dita. Inicialmente, para determinar se a ocorrência de uma variável é livre ou ligada, é necessário determinar primeiro os escopos dos quantificadores que ocorrem na fórmula. Mas, afinal, o que significa "escopo"? Uma semântica dessa palavra, adequada ao presente contexto, é o de abrangência. Isto é, o escopo de um quantificador significa, na Lógica, a sua abrangência ou domínio de influência. O conceito de escopo é definido a seguir.

Definição 6.10 (escopo de um quantificador) *Seja E uma fórmula da Lógica de Predicados:*

1. *Se* $(\forall \breve{x})H$ *é uma subfórmula de* E,
   então o escopo de $(\forall \breve{x})$ *em* E *é a subfórmula* H;
2. *Se* $(\exists \breve{x})H$ *é uma subfórmula de* E,
   então o escopo de $(\exists \breve{x})$ *em* E *é a subfórmula* H.

Portanto, o escopo de um quantificador é tudo aquilo que vem após ele, mas que está no seu domínio, ou abrangência de sua influência. Isso significa que dada, por exemplo, uma fórmula do tipo $((\exists x)H \wedge G)$, o escopo de $(\exists x)$ em $((\exists x)H \wedge G)$ é a subfórmula H. Observe que não temos o resto da fórmula escrito após o quantificador $(\exists x)$. Isso acontece porque a abrangência do quantificador é apenas a subfórmula H e essa abrangência não inclui G.

Exemplo 6.9 (escopo de um quantificador) Considere a fórmula a seguir:

$$E = (\forall x)(\exists y)((\forall z)p(x,y,w,z) \rightarrow (\forall y)q(z,y,x,f(z_1))).$$

Nesse caso temos:

1. o escopo do quantificador $(\forall x)$ em E é
   $(\exists y)((\forall z)p(x,y,w,z) \rightarrow (\forall y)q(z,y,x,f(z_1)))$;
2. o escopo do quantificador $(\exists y)$ em E é
   $((\forall z)p(x,y,w,z) \rightarrow (\forall y)q(z,y,x,f(z_1)))$;
3. o escopo do quantificador $(\forall z)$ em E é $p(x,y,w,z)$. Observe que, nesse caso, o escopo de $(\forall z)$ não é o resto da fórmula E, a partir de $(\forall z)$;
4. o escopo do quantificador $(\forall y)$ em E é $q(z,y,x,f(z_1))$.

O escopo de um quantificador em uma fórmula E é a subfórmula de E referida pelo quantificador. O escopo é a abrangência ou domínio do quantificador. A subfórmula $((\forall z)p(x,y,w,z) \rightarrow (\forall y)(z,y,x,f(z_1)))$ é o escopo de $(\exists y)$, pois essa é a subfórmula de E referida pelo quantificador $(\exists y)$, que está quantificando sob o seu escopo. Da mesma forma, o quantificador $(\forall z)$ está quantificando apenas sobre a subfórmula $p(x,y,w,z)$ e não sobre o resto da fórmula E, a partir de $(\forall z)$. ■

Se uma variável x ocorre no escopo de um quantificador em x, como, por exemplo $(\forall x)$, dizemos que tal ocorrência é ligada. Ou seja, ela é ligada ao quantificador $(\forall x)$. Caso contrário, se x não ocorre no escopo de um quantificador em x, dizemos que tal ocorrência é livre. Portanto, dada uma variável x em uma fórmula E, ela pode ocorrer livre ou ligada em E, conforme é definido a seguir.

Definição 6.11 (ocorrência livre e ligada) *Sejam* $\breve{x}$ *uma variável e* E *uma fórmula:*

1. *Uma ocorrência de* $\breve{x}$ *em* E *é ligada se* $\breve{x}$ *está no escopo de um quantificador* $(\forall \breve{x})$ *ou* $(\exists \breve{x})$ *em* E;
2. *Uma ocorrência de* $\breve{x}$ *em* E *é livre se não for ligada.*

Fique atento e observe o seguinte: as ocorrências das variáveis dos quantificadores não são livres e nem ligadas. Isto é, a variável $\breve{x}$ em $(\forall \breve{x})$, ou em $(\exists \breve{x})$, não é classificada.

Exemplo 6.10 (ocorrência livre e ligada) Considere a fórmula E do Exemplo 6.9. A fórmula E é escrita novamente indicando as variáveis livres com índices v e as ligadas com índices g.

$$E = (\forall x)(\exists y)((\forall z)p(x_g, y_g, w_v, z_g) \to (\forall y)q(z_v, y_g, x_g, f(z_{1_v}))).$$

Temos, então, as seguintes observações:

1. A variável z ocorre ligada em $p(x, y, w, z)$, pois z está no escopo de $(\forall z)$.
2. A variável z ocorre livre em $q(z, y, x, f(z_1))$, pois, nesse caso z não está no escopo de nenhum quantificador em z.
3. A ocorrência da variável y em $q(z, y, x, f(z_1))$ é uma ocorrência ligada, pois nesse caso y está no escopo de dois quantificadores $(\exists y)$ e $(\forall y)$.
4. A ocorrência da variável y em $q(z, y, x, f(z_1))$ está ligada ao quantificador mais próximo, isto é, y está ligada a $(\forall y)$.
5. As variáveis que ocorrem nos quantificadores não são livres e nem ligadas.

■

Definição 6.12 (variável livre e ligada) *Sejam $\breve{x}$ uma variável e E uma fórmula que contém $\breve{x}$:*

1. *A variável $\breve{x}$ é ligada em E, se existe pelo menos uma ocorrência ligada de $\breve{x}$ em E;*
2. *A variável $\breve{x}$ é livre em E, se existe pelo menos uma ocorrência livre de $\breve{x}$ em E.*

Exemplo 6.11 (variável livre e ligada) Considere a fórmula E do Exemplo 6.9. Nesse caso:

1. as variáveis x, y, e z são ligadas em E;
2. as variáveis w, z, e z_1 são livres em E;
3. a variável z é livre e ligada em E pois há uma ocorrência livre e uma ocorrência ligada de z em E. ■

Conforme o Exemplo 6.10, uma variável pode ocorrer livre e ligada em uma mesma fórmula. Nesse caso, como é analisado no próximo capítulo, a interpretação dessa variável é feita de forma diferente, dependendo se a ocorrência é livre ou ligada. A interpretação de uma variável que ocorre ligada é determinada pelo quantificador ao qual ela está ligada. Por outro lado, as ocorrências livres são interpretadas de forma diferente e não dependem de quantificadores. Além disso, para interpretar uma fórmula da Lógica de Predicados, além de interpretar as variáveis que ocorrem livres, é necessário interpretar os símbolos de predicados e de funções que ocorrem na fórmula. Nesse sentido, definimos como símbolos livres de uma fórmula, os símbolos que não estão ligados a quantificadores. E, para interpretar uma fórmula, é necessário interpretar, de forma própria, esses símbolos livres. Ou seja, para interpretar uma fórmula, é necessário, antes de qualquer coisa, interpretar as variáveis que ocorrem livres, os símbolos de predicados e os de funções.

Definição 6.13 (símbolos livres) *Dada uma fórmula E, os seus símbolos livres são as variáveis que ocorrem livres em E, os símbolos de função e os símbolos de predicado.*

Os símbolos livres de uma fórmula são todos os seus símbolos, exceto as variáveis ligadas, as variáveis dos quantificadores, os conectivos, e os símbolos de pontuação.

Exemplo 6.12 (símbolos livres) O conjunto $\{w, z, z_1, p, q, f\}$ é formado pelos símbolos livres da fórmula E, do Exemplo 6.9. Observe que as variáveis que ocorrem apenas na forma ligada não são símbolos livres. ■

A partir da classificação das variáveis de uma fórmula, temos a classificação da fórmula que as contém.

Definição 6.14 (fórmula fechada) *Uma fórmula é fechada quando não possui variáveis livres.*

Exemplo 6.13 (fórmula fechada) A fórmula E do Exemplo 6.9 não é fechada, pois contém variáveis livres. Considere as fórmulas a seguir:

1. Adicione o quantificador $(\forall z_1)$ no início da fórmula E. O resultado, indicado por E_1, não é uma fórmula fechada, pois contém as variáveis livres w e x.

$$E_1 = (\forall z_1)(\forall x)(\exists y)((\forall x)p(x, y, w, z) \rightarrow (\exists y)q(z, y, x, f(z_1)))$$

2. Adicione o quantificador $(\forall x)$ no início da fórmula E_1. O resultado, indicado por E_2, não é uma fórmula fechada, pois contém as variável livre w.

$$E_2 = (\forall x)(\exists y)((\forall x)p(x, y, w, z) \rightarrow (\exists y)q(z, y, x, f(z_1)))$$

3. Adicione o quantificador $(\forall w)$ no início da fórmula E_2. O resultado, indicado por E_3, é uma fórmula fechada, pois não contém variáveis livres.

$$E_3 = (\forall w)(\forall x)(\exists y)((\forall x)p(x, y, w, z) \rightarrow (\exists y)q(z, y, x, f(z_1)))$$

Nesse caso, as ocorrências livres de w, z e z_1 na fórmula E se tornam ocorrências ligadas em E_3 devido à adição de quantificadores universais.

■

No exemplo anterior, Exemplo 6.13, é dada uma fórmula E, que não é fechada. A partir dela, pela adição de quantificadores universais, obtemos uma fórmula fechada. É como fazer um "fecho" da fórmula inicial. Assim, dada uma fórmula H da Lógica de Predicados, podemos obter, pela adição de quantificadores no início da fórmula H, dois tipos de fórmulas fechadas: o fecho universal e o existencial de H.

Definição 6.15 (fecho de uma fórmula) *Seja H uma fórmula da Lógica de Predicados e*
$\{\breve{x}_1, ..., \breve{x}_n\}$ *o conjunto das variáveis livres em H:*

1. *O fecho universal de H, indicado por $(\forall *)H$,*
   é dado pela fórmula $(\forall \breve{x}_1)...(\forall \breve{x}_n)H$;
2. *O fecho existencial de H,indicado por $(\exists *)H$,*
   é dado pela fórmula $(\exists \breve{x}_1)...(\exists \breve{x}_n)H$.

Exemplo 6.14 (fecho de uma fórmula) Considere as fórmulas.

1. A fórmula E_3, do Exemplo 6.13, é o fecho universal da fórmula E. Logo, $E_3 = (\forall *)E$.
2. O fecho existencial de E é dado pela fórmula:

   $$E_4 = (\exists w)(\exists z)(\exists z_1)(\forall x)(\exists y)((\forall x)p(x, y, w, z) \to (\forall y)q(z, y, x, f(z_1)))$$

   A fórmula E_4 é obtida a partir da fórmula E, adicionando os quantificadores existenciais nas suas variáveis livres. Logo, $E_4 = (\exists *)E$.
3. Considere a fórmula fechada $H = (\exists x)p(x)$. Como H não possui variáveis livres, então o seu fecho universal, ou existencial, é igual a H. Isto é, nesse caso, $(\forall *)H = (\exists *)H = H$.

■

6.8 Formas normais

Como na Lógica Proposicional, dada uma fórmula H, da Lógica de Predicados, existe uma fórmula G, equivalente a H, que está na forma normal. Tais fórmulas são definidas a partir dos literais.

Definição 6.16 (literal na Lógica de Predicados) *Um literal, na Lógica de Predicados, é um átomo ou a negação de um átomo. Um átomo é um literal positivo. A negação de um átomo é um literal negativo.*

Lembre se de que, na Lógica Proposicional, um literal é um símbolo proposicional, ou sua negação. Nesse sentido, os símbolos proposicionais da Lógica Proposicional correspondem aos átomos na Lógica de Predicados.

Exemplo 6.15 (literal) As fórmulas a seguir são literais:

1. Como P é um átomo, então P e $\neg P$ são literais;
2. Como $p(f(x,a),x)$ é um átomo, $\neg p(f(x,a),x)$ é um literal;
3. $q(x,y,z)$ e $\neg q(x,y,z)$ são literais.

■

As formas normais na Lógica de Predicados são definidas de forma análoga às formas normais da Lógica Proposicional.

Definição 6.17 (forma normal) *Seja H uma fórmula da Lógica de Predicados.*

1. *H está na forma normal conjuntiva, fnc, se é uma conjunção de disjunções de literais.*
2. *H está na forma normal disjuntiva, fnd, se é uma disjunção de conjunções de literais.*

Observe que, mesmo havendo quantificadores na Lógica de Predicados, fórmulas na fnc e na fnd não os consideram. As formas normais são conjunções ou disjunções de literais, que, por sua vez, são formadas a partir de átomos. Mas, afinal, que aplicação tem as formas normais na Lógica de Predicados? É claro que tem muitas e apresentamos, a seguir, uma delas. Em Programação em Lógica, o elemento fundamental da sintaxe da linguagem é a cláusula de programa, que é uma fórmula na forma normal disjuntiva.

Definição 6.18 (cláusula de programa) *Uma cláusula de programa, na Lógica de Predicados, é uma cláusula do tipo $C = (\forall *)G$, onde G está na forma normal disjuntiva e contém exatamente um literal positivo.*

Exemplo 6.16 (cláusula de programa) A fórmula C_1 a seguir é uma cláusula de programa. As outras fórmulas não são cláusulas de programa, pois não contêm exatamente um literal positivo.

$$\begin{aligned}
C_1 &= (\forall x)(\forall y)(p(x) \vee \neg q(x) \vee \neg r(x,y)) \\
&= (\forall *)((q(x) \wedge r(x,y)) \rightarrow p(x)), \\
C_2 &= (\forall x)(\forall y)(\neg p(x) \vee \neg r(x,y)) \\
&= (\forall *)(\neg p(x) \vee \neg r(x,y)), \\
C_3 &= (\forall y)(\forall z)(\neg p(y) \vee q(z) \vee r(z)) \\
&= (\forall *)(\neg p(y) \vee q(z) \vee r(z)).
\end{aligned}$$

Observe que a fórmula C_1 contém um literal positivo, sendo, portanto, uma cláusula. Além disso, ela pode ser apresentada em duas notações diferentes equivalentes, como indicado pelas equivalências a seguir:

$$\begin{aligned}
C_1 &= (\forall *)(p(x) \vee \neg q(x) \vee \neg r(x,y)) \\
&\Leftrightarrow (\forall *)(p(x) \vee \neg (q(x) \wedge r(x,y))) \\
&\Leftrightarrow (\forall *)(\neg (q(x) \wedge r(x,y)) \vee p(x)) \\
&\Leftrightarrow (\forall *)((q(x) \wedge r(x,y)) \rightarrow p(x)).
\end{aligned}$$

Portanto, C_1 pode ser denotada por:
$C_1 \quad = (\forall *)((q(x) \wedge r(x,y)) \rightarrow p(x))$.
Como toda cláusula é um fecho universal, o quantificador $(\forall *)$ é omitido, ficando implícito. Nesse caso, C_1 é, ainda, denotada por:
$C_1 \quad = (q(x) \wedge r(x,y)) \rightarrow p(x)$.
E em Programação em Lógica, C_1 é denotada por:
$C_1 \quad = p(x) \leftarrow q(x), r(x,y)$,
onde o conectivo $\wedge$ é representado por vírgula e a implicação é invertida. ■

Notação. Uma cláusula de programa:

$$(\forall *)(B \vee \neg A_1 \vee \ldots \vee \neg A_n)$$

é denotada por:

$$B \leftarrow A_1, \ldots, A_n.$$

Nesse caso, B é a cabeça da cláusula e $A_1, \ldots, A_n$ é a cauda.

Definição 6.19 (cláusula unitária) *Uma cláusula de programa unitária é uma cláusula do tipo $B \leftarrow$. Nesse caso, a cláusula não contém literais negativos.*

Uma cláusula unitária é também denominada como fato.

Definição 6.20 (programa lógico) *Um programa lógico é um conjunto de cláusulas de programa.*

Portanto, o ato de programar em Programação em Lógica, corresponde, mais ou menos, a determinar um conjunto de cláusulas que representam aquilo que é o objeto da programação. A seguir apresentamos um programa lógico e no próximo capítulo analisamos a semântica desse programa.

Exemplo 6.17 (programa lógico) O conjunto de cláusulas de programa a seguir forma um programa lógico.

1. $p(a,b) \ \leftarrow$

2. $p(f(x),y) \ \leftarrow \ p(x,z), q(g(f(x),z),y)$

Nesse programa lógico temos uma cláusula unitária e outra cláusula que não é unitária. É claro que não dá para entender nada a respeito do seu significado. Isso ocorre por estarmos apresentando apenas a sintaxe e ainda falta a semântica, que estabelece o significado dos símbolos envolvidos. Consideramos essa semântica no próximo capítulo, onde analisamos o significado desse programa. ■

6.9 Exercícios

1. Responda, justificando sua resposta, às seguintes questões:
   (a) Todo termo é uma fórmula?
   (b) Todo literal é uma expressão?
   (c) Toda expressão é um literal?
2. Seja E a fórmula a seguir:
   $E = (\forall w)(\forall z)(\forall z_1)(\forall x)(\exists y)((\forall x)p(x, y, w, z) \to (\forall y)q(z, y, x, z1))$
   (a) Determine todas as subfórmulas de E.
   (b) Determine todas as subexpressões de E.
3. Reescreva os parênteses das fórmulas a seguir.
   (a) $(\forall x)p(x) \lor \neg(\forall x)q(x) \to r(y)$
   (b) $(\exists z)p(z) \leftrightarrow \neg q(y)$
   (c) $(\exists x)(\forall x)\neg p(x)$
4. Elimine o máximo de símbolos de pontuação das fórmulas a seguir, mantendo a fórmula original.
   (a) $((\forall x)(p(x) \to q(x)))$
   (b) $(((\forall x)p(x) \to q(y)) \to ((\forall z)q(z)))$
   (c) $(((\exists y)((\forall z)p(z)) \land r(z)) \lor ((\forall x)q(x)))$
5. Seja E uma fórmula e $\breve{x}$ uma variável. Responda justificando sua resposta:
   (a) É possível haver ocorrências de $\breve{x}$ em E livres e ligadas?
   (b) É possível a variável $\breve{x}$ ser livre e ligada em E ao mesmo tempo?
6. Dê exemplo de uma fórmula H na qual uma variável x ocorre tanto livre quanto ligada.
7. Responda:
   (a) Existe fórmula sem símbolo livre?
   (b) Quais são os símbolos livres de uma fórmula fechada?
   (c) Toda variável é símbolo livre?
8. Determine o fecho universal e existencial das fórmulas a seguir:
   (a) $F_1 = p(x, y)b)$
   (b) $F_2 = (\exists x)p(x, y)$
   (c) $F_3 = (\exists y)(\exists x)p(x, y)$

9. Dê exemplos:

   (a) de uma fórmula cujo fecho existencial contém apenas quantificadores universais;

   (b) de uma fórmula cujo fecho universal ou existencial não contém quantificadores;

   (c) de uma fórmula cujo fecho universal ou existencial é igual a ela própria.