NHI2049-13 LÓGICA BÁSICA
– NOTAS DE AULA –

JAIR DONADELLI

INTRODUÇÃO

De modo geral, a Lógica é a disciplina que se ocupa do estudo sistemático das formas de argumento, a ideia de que *a validade de um argumento é determinada pela sua forma lógica e não pelo seu conteúdo.* Um argumento válido é aquele em que existe uma relação específica, de suporte lógico, entre os pressupostos do argumento e sua conclusão. Especificamente, a Lógica Matemática estuda as formas de argumentos matemáticos por meio de uma linguagem formal criada artificialmente. Como parte disso, a Lógica Simbólica estuda as abstrações simbólicas que capturam as características formais dos argumentos da linguagem natural. A Lógica clássica de primeira ordem é uma lógica simbólica para a qual daremos uma introdução neste curso.

A teoria dos conjuntos criada por Georg Cantor exibia muitos paradoxos e havia o temor de que a matemática como um todo não estava assentada em um terreno seguro. Esse momento ficou conhecido como "a crise dos fundamentos da matemática" e a saída mais famosa para essa crise foi formulada por David Hilbert. O "programa de Hilbert", como ficou conhecido, influenciou muitos trabalhos realizados na primeira metade do século 20, inclusive de Kurt Gödel, um dos pensadores mais profundos desse século, e de Alan Turing, o pai da computação.

O programa de Hilbert. Foi uma proposta para resolver a crise nos fundamentos da matemática, Hilbert propôs fundamentar todas as teorias existentes em um conjunto finito e completo de axiomas e fornecer uma prova de que esses axiomas eram consistentes. A linguagem e a axiomatização da lógica de primeira ordem seguem alguns princípios estabelecidos no programa de Hilbert:

- a linguagem da lógica é composta por uma quantidade enumerável de símbolos;
- as fórmulas são sequências finitas de símbolos;
- as demonstrações são sequências finitas de fórmulas;
- há um algoritmo que, em finitos passos, determina se uma sequência de símbolos é uma fórmula ou não;
- há um algoritmo que, em finitos passos, determina se uma sequência de fórmulas é uma demonstração ou não.

Em resumo, propõe uma formulação de toda a matemática em uma linguagem formal e manipulada de acordo com regras bem definidas. Ao aparato sintático descrito acima acrescenta-se uma semântica, em geral dada por uma interpretação que atribui um valor-verdade (por exemplo, *verdadeiro* ou *falso*) para as sentenças da linguagem formal. As propriedades fundamentais dos sistemas de dedução são

- *completude*: uma prova de que todas as afirmações matemáticas verdadeiras podem ser provadas no formalismo;
- *consistência*: uma prova de que nenhuma contradição pode ser obtida no formalismo.
- *decidibilidade* (entscheidungsproblem): deve haver um algoritmo para decidir se qualquer afirmação matemática tem prova no formalismo.

Gödel mostrou que, em qualquer sistema lógico,a completude e a consistência propostas por Hilbert não podem ser atingidas.

Os lógicos contemporâneos

(1) constroem linguagens simbólicas, rigorosas e livres de ambiguidades e de contexto, adequadas para lidar com a relação de consequência. As linguagens possuem duas componentes relevantes:
   (a) a *sintática*: os símbolos da linguagem (chamado *alfabeto*) e as *regras gramaticais* de combinação de símbolos, às quais estes estão sujeitos, para a construção das fórmulas da linguagem;
   (b) a *semântica*: define precisamente o significado das fórmulas.
(2) constroem um *cálculo*, ou sistema dedutivo, especificando os axiomas dentre as fórmulas e as regras de inferência que independem da semântica (é um aparato sintático).

E assim temos um sistema lógico, ou uma lógica.

Linguagens. O discurso da lógica, assim como o da matemática, faz-se através da linguagem coloquial, irrestrita. A lógica se preocupa também com a estrutura formal das linguagens, as quais são, entre outras coisas, objetos de seu estudo.

Por que precisamos criar uma linguagem para formalizar as formas de raciocínio? Para evitar os paradoxos[1] e imprecisões decorrentes do uso da linguagem natural/coloquial. Isso é importante quando estudamos assuntos mais restritos, com menos complexidade expressiva, porém com maior exigência de rigor.

[1]Considere o conjunto C de todos os números naturais definidos por uma sentença em português com no máximo 8 palavras. Certamente é um conjunto finito, de modo que deve haver naturais que não pertencem a C. O “menor número natural que não pertence a C” é um natural definido com no máximo 8 palavras!

Evidentemente, uma linguagem artificial com tais propósitos tem poder expressivo inferior à linguagem natural. Acrescentar expressividade às linguagens lógicas, aproximando-as da linguagem natural, sem que se perca o rigor aumenta-lhes em muito a complexidade.

As linguagens formais são rígidas e rigorosas enquanto que a linguagem natural, a que serve de comunicação entre os lógicos, é permissiva, temos assim dois níveis de linguagem.

Linguagem×Metalinguagem. Se a linguagem da matemática é a própria lógica, qual linguagem utilizaremos quando construímos a lógica?

A *linguagem* é o objeto de estudo (linguística) e a *metalinguagem* é a linguagem com que os estudiosos transmitem seus estudos. A metalinguagem é a linguagem usada para descrever algo sobre a lógica, pois a lógica matemática é, em si, parte da matemática e como qualquer outra parte da matemática, há resultados e teoremas que dizem a respeito dela.

Quando *mencionamos* um símbolo da linguagem usamos aspas simples para diferenciá-lo do seu *uso*, por exemplo,

- ’quadrado’ tem seis letras
- ’oxítona’ é proparoxítona
- ’a neve é branca’ é verdadeiro se, e somente se, a neve é branca

A expressão ’quadrado’ é o nome da palavra *quadrado* a qual dá nome a forma geométrica. Assim, em <u>quadrado tem quatro lados</u> a palavra ’quadrado’ está sendo usada para falar da forma geométrica. Em <u>’quadrado’ tem seis letras</u> não estamos falando mais da forma geométrica, mas da palavra que é o nome dessa forma.

1. Lógica de proposições — sintaxe

1.1. **A linguagem formal.** O **alfabeto** $\mathscr{A}_0$ é o conjunto dos símbolos que compõem linguagem

$$\mathscr{A}_0 = \{\, p_i \colon i \in \mathbb{N} \} \cup \{\neg, \rightarrow\} \cup \{(,)\}$$

onde temos

símbolos proposicionais atômicos:	$p_1\ p_2\ p_3\ \dots$
conectivos lógicos:	$\neg\ \rightarrow$
símbolos de pontuação:	( )

Lemos os símbolos $\neg$ e $\rightarrow$ como *negação* e *implicação*, respectivamente.

Ressaltamos que esses símbolos são apenas símbolos da linguagem, os conectivos lógicos, por exemplo, não devem ser confundidos com os operadores lógicos.

1.2. **Fórmulas.** As expressões que podemos formar são as cadeias (ou sequências) finitas de símbolos tomados do alfabeto $\mathscr{A}_0$ como, por exemplo,

- $((p_1 \to p_7) \to (\neg p_{100}))$,
- $())p_5$ e
- $\neg(\neg p_2)\neg$.

Claramente, os dois últimos exemplos são expressões que não interessam.

Na metalinguagem que usamos para descrever a lógica proposicional usamos letras gregas

$$\alpha,\ \beta,\ \gamma,\ \delta,\ \epsilon,\ \zeta,\ \eta,\ \theta,\ \iota,\ \kappa,\ \lambda,\ \mu,\ \nu,\ \pi,\ \rho,\ \sigma,\ \tau,\ \upsilon,\ \phi,\ \chi,\ \psi,\ \omega$$

para denotar expressões, são ditas *metavariáveis.*

Uma **fórmula bem formada** (FBF) ou simplesmente **fórmula** é qualquer expressão que pode ser formada aplicando-se um número *finito* de vezes as regras:

(F1) os símbolos atômicos são FBF, chamadas **fórmulas atômicas**;
(F2) se α é FBF, então $(\neg\alpha)$ é FBF;
(F3) se α e β são FBFs, então $(\alpha \to \beta)$ é FBF;
(F4) não há outras FBFs além das obtidas pelo uso das regras (F1), (F2) e (F3).

A última regra nos assegura que todas as FBFs, e só elas, podem ser construídas pelas regras anteriores.

Exemplo 1. São exemplos de fórmulas bem formadas

- p_1, $(\neg p_2)$, $(p_3 \to (p_1 \to (\neg p_1)))$;
- Se α, β, γ denotam FBF então $(\alpha \to (\beta \to \gamma))$ denota uma FBF que é diferente da FBF denotada por $((\alpha \to \beta) \to \gamma)$.

É importante entender que $(p_1 \to p_2)$, por exemplo, é uma fórmula da linguagem enquanto que $(\alpha \to \beta)$ não é uma fórmula da linguagem, mas uma expressão metalinguística que usamos para nos referir a um tipo de forma de FBF, aquelas que são escritas quando trocamos as letras gregas por fórmulas como, por exemplo, $(p_1 \to p_1)$, $(p_1 \to p_2)$, $((p_1 \to p_2) \to (p_1 \to p_3))$, $((p_1 \to p_2) \to (p_1 \to p_2))$, etc. enfim há uma infinidade delas. Dizemos que $\alpha \to \beta$ é um **esquema** de fórmula.

O conjunto $\mathscr{L}_0$ de todas as fórmulas da linguagem da lógica proposicional fica definido como *o menor conjunto*[2] $\mathscr{X}$ *formado pelas cadeias de símbolos da alfabeto que satisfazem as propriedades*:

(1) $p_1, p_2, \ldots \in \mathscr{X}$,
(2) se $\alpha \in \mathscr{X}$ então $(\neg\alpha) \in \mathscr{X}$,
(3) se $\alpha, \beta \in \mathscr{X}$ então $(\alpha \to \beta) \in \mathscr{X}$.

[2]"menor conjunto" quer dizer que se X é um conjunto de fórmulas que satisfaz as três propriedades então $X \supseteq \mathscr{L}_0$

Desse modo, $\mathscr{L}_0$ é indutivo e nos oferece um estratégia para demonstrações de propriedades da linguagem.

1.3. **Indução.** As fórmulas são objetos finitos gerados recursivamente e, portanto, podemos provar propriedades de todas as fórmulas usando indução estrutural.

METATEOREMA 1 (INDUÇÃO PARA FÓRMULAS) *Suponha que uma propriedade de fórmulas*

(1) *vale para toda fórmula atômica e*
(2) *se vale para a fórmula α então também vale para $(\neg\alpha)$ e*
(3) *se vale para as fórmulas α e β, então também vale para $(\alpha \to \beta)$.*

Então essa propriedade vale para todo elemento de $\mathscr{L}_0$.

DEMONSTRAÇÃO. Seja X o conjunto de todas as fórmulas de $\mathscr{L}_0$ que tenha uma dada propriedade de fórmulas. As fórmulas atômicas estão em X pela hipótese (1). Se $\alpha, \beta \in X$ então $(\alpha \to \beta) \in X$ por (2) e por (3). Assim, $\mathscr{L}_0 \subseteq X$, portanto, $\mathscr{L}_0 = X$. □

Exemplo 2. Vamos provar usando a indução que *toda fbf tem um quantidade par de parênteses.* Cada fórmula atômica tem 0 parênteses. Para todo α que tem um número par, digamos $2n$, de parênteses, $(\neg\alpha)$ tem $2n+2 = 2(n+1)$ parênteses, portanto par. Suponha que α e β tenham, respectivamente, $2n$ e $2m$ parênteses, então $(\alpha \to \beta)$ tem $2n+2m+2 = 2(n+m+1)$ parênteses. Usando a indução para fórmulas, concluímos que toda FBF tem um quantidade par de parênteses.

O exemplo a seguir ilustra uma **definição recursiva** para fórmulas baseada no princípio indutivo. Pela indução para fórmulas, se definimos uma característica para as fórmulas atômicas e se tal característica fica definida para $(\neg\alpha)$ e $(\alpha \to \beta)$ sempre que está definida para as fórmulas α e para β, então ela fica definida para toda FBF de $\mathscr{L}_0$.

Exemplo 3 (Grau de complexidade). As vezes é conveniente medir a complexidade de uma FBF pelo seu *grau* dado por:

(1) $\mathrm{grau}(\alpha) = 0$ se α é fórmula atômica;
(2) $\mathrm{grau}(\neg\alpha) = \mathrm{grau}(\alpha) + 1$; e
(3) $\mathrm{grau}(\alpha \to \beta) = \max\{\mathrm{grau}(\alpha), \mathrm{grau}(\beta)\} + 1$;

Pelo metateorema 1 o grau de complexidade está definido para toda fórmula de $\mathscr{L}_0$.

1.4. **Leitura única.** O leitor atento pode perguntar se as definições dos símbolos e a regra de formação das fórmulas garantem que a as fórmulas de $\mathscr{L}_0$ não são ambíguas no sentido de que uma dada fórmula não pode ser lida de mais de uma maneira de acordo com as regras estabelecidas. De fato, pode se provar que uma fórmula de $\mathscr{L}_0$ deve satisfazer exatamente uma dentre as condições (F1), (F2) e (F3) que regem a formação de fórmulas.

A demonstração do seguinte resultado decorre da seguinte propriedade de fórmulas bem formadas: em toda FBF não atômica há um e apenas um conectivo lógico que satisfaz a condição "à esquerda dele, o número de abre-parênteses é exatamente um a mais que o número de fecha-parênteses".

METATEOREMA 2 (TEOREMA DA LEITURA ÚNICA) *Para toda* FBF α*, uma, e apenas uma, das afirmações abaixo é verdadeira:*

- α *é uma fórmula atômica;*
- *existe uma única* FBF β *tal que* α *é a fórmula* $(\neg\beta)$*;*
- *existem únicas* FBF*s* β *e* γ *tais que* α *é a fórmula* $(\beta \to \gamma)$.

Exercício 4. Prove usando indução para fórmulas que o número de abre-parênteses em uma fórmula é sempre igual ao número de fecha-parênteses.

Exercício 5. Prove, usando indução para fórmulas, que se houver conectivos nessa fórmula, então, dada qualquer ocorrência de um conectivo, o número de abre-parênteses à esquerda desse conectivo é estritamente maior que o número de fecha-parênteses à esquerda dele.

Exercício 6. Prove que em uma fórmula do tipo $(\neg\alpha)$ ou do tipo $(\alpha \to \beta)$, com α e β fórmulas, o número de abre-parênteses que estão à esquerda do conectivos $\neg$ (no primeiro caso) e $\to$ (no segundo) é exatamente um a mais que o número de fecha-parênteses à esquerda desses conectivos.

Exercício 7. Prove usando indução para fórmulas que em toda fórmula α ocorre apenas um conectivo lógico que satisfaz a condição "à esquerda dele, o número de abre-parênteses é exatamente um a mais que o número de fecha-parênteses".

DEMONSTRAÇÃO. Vamos demonstrar que toda $\alpha \in \mathscr{L}_0$ tem a propriedade P: a fórmula é atômica ou nela ocorre um único conectivo tal que, à sua esquerda, o número de abre-parênteses é exatamente um a mais do que o número de fecha-parênteses.

Dada $\alpha \in \mathscr{L}_0$, se α é atômica então α tem P.

Considere $\alpha \in \mathscr{L}_0$ uma fórmula que tem P. Vamos provar que $(\neg\alpha)$ tem P. Pelo exercício 6, temos para o conectivo $\neg$ que à esquerda dele o número de abre-parênteses é exatamente um a mais que o número de fecha-parênteses. Logo, resta provar que é o único em $(\neg\alpha)$ com essa propriedade.

Por contradição, se houver outro, então esse conectivo $c \in \{\neg, \to\}$ é um conectivo em α. Se c, em $(\neg\alpha)$, tem à sua esquerda um número de abre-parênteses igual ao de fecha-parênteses mais 1, então, em α, o número de abre-parênteses à esquerda é igual ao de fecha-parênteses à esquerda, contrariando a afirmação do Exercício 5. Portanto, se α tem P então $(\neg\alpha)$ também tem P.

Agora, sejam $\alpha, \beta \in \mathscr{L}_0$ fórmulas que têm a propriedade P. Vamos provar que $(\alpha \to \beta)$ tem P. De acordo com o exercício 6, para o conectivo $\to$, o número de abre-parênteses à esquerda dele é exatamente um a mais que o número de fecha-parênteses. Portanto, resta provar que é o único em $(\alpha \to \beta)$ com essa propriedade

Por contradição, se houver outro, então esse conectivo $c \in \{\neg, \to\}$ é um conectivo ou em α ou em β. Se c, em $(\alpha \to \beta)$, tem à sua esquerda um número de abre-parênteses igual ao de fecha-parênteses mais 1, então, em α (ou em β, caso c ocorra em β), o número de abre-parênteses à esquerda é igual ao de fecha-parênteses à esquerda, contrariando a afirmação do Exercício 5. Portanto, se α e β têm P então $(\alpha \to \beta)$ também tem P.

Pelo metateorema da indução podemos concluir que todo $\alpha \in \mathscr{L}_0$ tem P. $\square$

Exercício 8. Use os exercícios acima e conclua o metateorema da leitura única, enunciado na página 6.

1.5. **Subfórmulas.** A partir do exercício 7 acima podemos descrever a formação de uma fórmula α através de uma árvore binária: começando por α (nó **raiz**), em cada nó temos uma fórmula β da forma $(\gamma \to \delta)$ em que γ e δ são fórmulas e γ pode ser, eventualmente, uma "fórmula vazia" (no caso do conectivo $\neg$). As fórmulas γ e δ são os nós filhos de β. Esse processo termina quando γ e/ou δ são símbolos atômicos (nós **folha**). Essa árvore é chamada de **árvore de formação** da fórmula.

Exemplo 9. A fórmula $\Big((p_1 \to p_2) \to \big((\neg p_3) \to (p_4 \to p_5)\big)\Big)$ pode ser lida de uma única maneira, representada pelo diagrama da figura 1, a árvore de formação da fórmula.

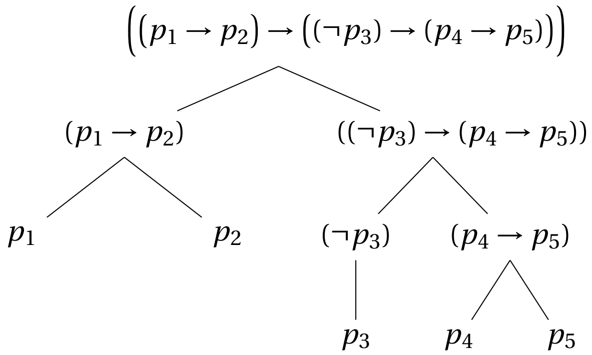

FIGURA 1. A árvore de formação da fórmula $((p_1 \to p_2) \to ((\neg p_3) \to (p_4 \to p_5)))$.

Todas as fórmulas presentes na árvore de formação da fórmula α são as subfórmulas de α. Esse é um conceito definido recursivamente a seguir. As fórmulas intermediárias que aparecem no processo de construção de uma fórmula através das regras (F1)–(F3) são chamadas de **subfórmulas próprias**.

Exemplo 10. p_1, p_2, $(\neg p_1)$ e $(p_2 \to (\neg p_1))$ são subfórmulas de $(p_1 \to (p_2 \to (\neg p_1)))$.

Formalmente, a definição do conjunto das subfórmulas de uma fórmula é recursiva. As subfórmulas de α definem o conjunto

(1) $\mathrm{Sf}(\alpha) = \{\alpha\}$ para toda FBF atômica p;
(2) $\mathrm{Sf}(\neg\alpha) = \mathrm{Sf}(\alpha) \cup \{(\neg\alpha)\}$;
(3) $\mathrm{Sf}(\alpha \to \beta) = \mathrm{Sf}(\alpha) \cup \mathrm{Sf}(\beta) \cup \{(\alpha \to \beta)\}$;

As subfórmulas próprias de α são os elementos de $\mathrm{Sf}(\alpha) \setminus \{\alpha\}$.

1.6. **Abreviaturas e simplificações.** Aqui e em outros momentos vamos assumir algumas convenções de notação para facilitar nossa escrita e o estudo da lógica de proposições. Todas estas simplificações são *convenções metalinguísticas*. Essas regras são *informais*, nos momentos que exigem resultados mais rigorosos, não devemos considerar essas simplificações.

Usaremos os símbolos $\vee$, $\wedge$ e $\leftrightarrow$, respectivamente, *disjunção*, *conjunção* e *bi-implicação*, como abreviações para certas construções envolvendo apenas $\neg$ e $\to$

- $(\alpha \wedge \beta)$ é a abreviação de $(\neg(\alpha \to (\neg\beta)))$,
- $(\alpha \vee \beta)$ é a abreviação de $((\neg\alpha) \to \beta)$, e
- $(\alpha \leftrightarrow \beta)$ é a abreviação de $((\alpha \to \beta) \wedge (\beta \to \alpha))$.

Representamos os conectivos $\vee$, $\wedge$, $\to$ e $\leftrightarrow$ genericamente pelo símbolo $\square$, no caso de múltiplas ocorrências de conectivos usaremos índices $\square_1$, $\square_2$ e assim por diante. Com isso $\alpha \square \beta$ indica uma fórmula composta por α, β e algum dos conectivos e qual deles, especificamente, não importa no momento em que se usa o símbolo $\square$ com o cuidado de que duas ocorrências numa mesma frase significa o mesmo conectivo.

Representamos os símbolos atômicos por algumas das letras do final do alfabeto da Língua Portuguesa, por exemplo $p, q, r, s, t, u, v, x, z$. Sempre que precisamos de muitas símbolos atômicos usamos os símbolos formais.

1.6.1. *Omissão de parênteses.* Para simplificar notação e facilitar a leitura omitimos a escrita de parênteses de acordo com as seguintes regras, para evitar ambiguidade.

(1) Omitimos os parênteses mais externos: $\neg\alpha$ deve ser lido como $(\neg\alpha)$ e $\alpha \square \eta$ deve ser lido como $(\alpha \square \eta)$.

(2) Adotamos a seguinte ordem de precedência para os conectivos: $\neg$, $\wedge$, $\vee$, $\rightarrow$, $\leftrightarrow$. Assim, por exemplo, $p \vee q \wedge r$ deve ser lido como $(q \vee (q \wedge r))$; $\neg p \vee r$ deve ser lido como $((\neg p) \vee r)$; $\neg p \vee r \rightarrow q \wedge s$ deve ser lido como $(((\neg p) \vee r) \rightarrow (q \wedge s))$.

(3) As repetições de um mesmo conectivo são aninhadas pela direita: $p \rightarrow q \rightarrow r \rightarrow s$ deve ser lido como $(p \rightarrow (q \rightarrow (r \rightarrow s)))$.

Por exemplo,

- $\neg\alpha \vee \beta$ lê-se $((\neg\alpha) \vee \beta)$,
- $\neg\neg\neg\alpha \wedge \beta$ lê-se $((\neg(\neg(\neg\alpha))) \wedge \beta)$,
- $\neg(\neg\neg\alpha \wedge \beta)$ lê-se $(\neg((\neg(\neg\alpha)) \wedge \beta))$,
- $\alpha \vee \beta \rightarrow \gamma$ lê-se $((\alpha \vee \beta) \rightarrow \gamma)$,
- $\delta \rightarrow \alpha \vee (\beta \rightarrow \gamma)$ lê-se $(\delta \rightarrow (\alpha \vee (\beta \rightarrow \gamma)))$,
- $\alpha \vee \beta \wedge \gamma \vee \delta \wedge \xi \vee \phi \wedge \rho$ deve ser lido como $(\alpha \vee ((\beta \wedge \gamma) \vee ((\delta \wedge \xi) \vee (\phi \wedge \rho))))$.

Apêndice: conjuntos indutivos. Nas definições gramaticais das linguagens dos sistemas lógicos usualmente recorremos às definições indutivas. Supõe-se dado um conjunto E, uma parte não vazia P de E e um conjunto F de operações sobre elementos de E que resulta em elemento de E. O conjunto $D \subseteq E$ é **indutivo** se

(I1) $P \subseteq D$;

(I2) D é fechado para as operações de F;

(I3) nenhum objeto está em D a não ser que possa ser obtido a partir dos elementos de P por um número finito de aplicações de operações de F.

Por exemplo, o conjunto dos números inteiros positivos pares pode ser definido a partir de $E = \mathbb{Z}$ pondo $P = \{2\}$ e F só tem a operação $d(n) = n + 2$. O conjunto dos inteiros positivos pode ser definido indutivamente tomando E o conjunto dos números primos acrescentado do 1 e F com a única operação $o(m, n) = m \cdot n$.

Uma diferença entre essa duas definições do mesmo conjunto, e que geralmente importa quando se trata de linguagens, é exemplificada a seguir. No primeiro exemplo, o número 12 é obtido de modo único, a saber, a partir do 2 aplica-se o "soma 2" cinco vezes. Já na segunda definição há pelo menos dois modos distintos: $o(o(3,2),2)$ e $o((2,2),3)$.

Se cada elememento de D é gerado a partir dos elementos de P e operações de F de modo único, então dizemos ter **legibilidade única**.

Exercício 11. Seja D um conjunto indutivo obtido a partir de E, P e F como acima. Prove que D é o menor conjunto que satisfaz as regras (I1) e (I2). Para isso, prove que D é dado pela intersecção de todos os subconjuntos de E que satisfazem as regras (1) e (2).