Capítulo 7

Auto-Referência e Teoremas de Incompletude

7.1 Introdução

Vamos agora abordar o sistema dedutivo do Cálculo de Predicados sob o ponto de vista algorítmico.

O objetivo principal deste capítulo é demonstrar os dois Teoremas de Incompletude de Gödel [1]

Seu primeiro teorema produz uma proposição aritmética verdadeira em $\mathbb{N}$, mas não dedutível de um sistema natural de axiomas da aritmética de primeira ordem. Mediante uma codificação engenhosa de conceitos metamatemáticos (noções de fórmula, de dedução formal, de dedutível), mostrou que funções definidas recursivamente (o que hoje chamamos de funções recursivas primitivas - veja mais adiante) podem ser representadas na aritmética e, devido ao aspecto recursivo das definições daqueles conceitos metamatemáticos, foi possível produzir uma fórmula que (do ponto de vista metamatemático - ou da metalinguagem) afirmava sua própria indemonstrabilidade (algo como "*ξ é o código de uma fórmula para a qual não existe uma dedução formal*") e que seria verdadeira em $\mathbb{N}$. Seu segundo teorema

[1] Kurt Gödel, *Über formal unentscheidbare Sätze der Principia mathematica und verwandter Systeme I*, Monatshefte für Mathematik und Physik, 38 (1931), pp. 173-198. Traduzido para o inglês, com o título *On Formally Undecidable Propositions*, em Jan van Heijenoort, *From Frege to Gödel*, [?], pp. 596-616.

de incompletude afirma que se a aritmética (ou qualquer sistema expressivo o suficiente) for consistente, então dela não se pode deduzir sua própria consistência (que pode ser expressa por uma fórmula dizendo que, por exemplo, "*a igualdade* $0 = 1$ *não é deditível*").

Gödel demonstrou seus teoremas sob a hipótese de que seu sistema de axiomas fosse ω-consistente, ou, trocando em miúdos, que fosse verdadeira em $\mathbb{N}$. John Barkley Rosser [2] modificou a demonstração para que somente fosse necessária a hipótese de que os sistema de axiomas da teoria fosse consistente. Ficará claro, na exposição a seguir, que é preciso um mínimo de aritmética na teoria em questão para que se possa expressar os conceitos metamatemáticos pertinentes.

7.2 Fenômenos de Auto-Referência em Linguagens Formais

Relembremos que a ideia principal envolvida na demonstração dos Teoremas de Incompletude de Gödel foi criar uma fórmula na linguagem objeto (da aritmética) que, do ponto de vista da metalinguagem, traduz a noção de dedução formal (da forma "ξ *é o código de uma dedução da fórmula* ζ"). Com esta fórmula e um resultado de diagonalização, é possível obter uma fórmula que, novamente do ponto de vista da metalinguagem, expressa a ideia que "ζ *é o código desta fórmula (que você está lendo agora), a qual não é dedutível*". Convém notar que esta auto-referência só pode ser percebida no nível da metalinguagem, pois na linguagem objeto teremos apenas uma fórmulas relatando certas igualdades, desigualdades e congruências.

Em 1969, o lógico F. William Lawvere [3] propôs um enfoque unificado para todos esses resultados de diagonalização, usando a chamada Teoria das Categorias [4]. Esse trabalho foi simplificado por Noson Yanofsky [5], mostrando o seguinte resultado:

[2] No artigo *Extensions of Some Theorems of Gödel and Church*, The Journal of Symbolic Logic, vol. 1 (1936), pp. 87-91.

[3] Nascido em 9 de fevereiro de 1937, em Muncie, Indiana, EUA, e ainda hoje - setembro de 2009 - ativo, está na Universidade de Buffalo, N. Y., EUA.

[4] Reimpresso como F. William Lawvere, *Diagonal Arguments and Cartesian Closed Categories*, Reprints in Theory and Applications of Categories, No. 15, 2006, pp. 1-13.

[5] Em seu artigo *A Universal Approach to Self-Referential Paradoxes, Incompleteness and Fixed Points*, The Bulletin of Symbolic Logic, Volume 9, Number 3, Sept. 2003.

Teorema 7.2.1 (Teorema da Diagonalização de Lawvere-Yanofsky) Suponha que Y seja um conjunto munido de uma função $\alpha : Y \to Y$ sem pontos fixos (isto é, para todo $y \in Y$, $\alpha(y) \neq y$). Então, para todos os conjuntos (não vazios) T e todas as funções $f : T \times T \to Y$, existe uma função $g : T \to Y$ não representada por f, ou seja, não existe $t' \in T$, tal que $g(\cdot) = f(\cdot, t')$.

Demonstração: Seja $g : T \to Y$ definida por $g(t) = \alpha \circ f(t,t) = \alpha(f(t,t))$. Seja $t' \in T$. Então $g(t') = \alpha(f(t', t')) \neq f(t', t')$, pois α não tem pontos fixos. Isto significa que não existe nenhum elemento $t' \in T$, tal que $g(\cdot) = f(\cdot, t')$, pois, se calcularmos g neste elemento t', temos a desigualdade desejada.

Este diagrama ilustra esta demonstração,

$$\begin{array}{ccc} T \times T & \xrightarrow{\hat{f}} & \{0,1\} \\ \Delta \uparrow & & \downarrow \alpha \\ T & \xrightarrow{g} & \{0,1\} \end{array}$$

sendo que $\Delta(t) = (t,t)$ é a inclusão de T na diagonal de $T \times T$. $\square$

Este resultado foi baseado no seguinte teorema, devido a Georg Cantor [6], em que demonstra não existir uma função sobrejetora do conjunto dos números naturais $\mathbb{N}$ sobre $2^{\mathbb{N}}$, o conjunto de todas as sequências binárias (que é identificado também com o conjunto de todos os subconjuntos de $\mathbb{N}$, por meio de suas funções características):

Teorema 7.2.2 (Teorema da Diagonalização de Cantor) Não existe uma função sobrejetora $f : \mathbb{N} \to 2^{\mathbb{N}}$.

Demonstração: O conjunto de todas as funções $f : \mathbb{N} \to 2^{\mathbb{N}}$ pode ser identificado com o conjunto de todas as funções $\hat{f} : \mathbb{N} \times \mathbb{N} \to \{0,1\}$ pela apliacação $f \mapsto \hat{f}$ definida por $\hat{f}(m, f(m)(n))$ (observemos que $f(m)$ é uma função da $\mathbb{N}$ a valores em $\{0,1\}$ e, por isso, o valor da função $f(m)$ calculada em $n \in \mathbb{N}$ é escrito como $f(m)(n)$). Assim, reduzimos o problema ao teorema anterior, pois a função $\alpha : \{0,1\} \to \{0,1\}$, dada por $\alpha(0) = 1$ e $\alpha(1) = 0$ não

[6] Georg Ferdinand Ludwig Phillip Cantor (1845-1918) foi o matemático que deu início à Teoria dos Conjuntos.

tem pontos fixos e, portanto, existe $g : \mathbb{N} \to \{0,1\}$ não representada pela função $\hat{f}$, o que implica que a função f não pode ser sobrejetora. $\square$

A demonstração clássica deste teorema consiste na listagem das sequências $f(m) = \hat{f}(m, f(m)(\cdot))$ e alterar a diagonal da tabela 7.1 (usando a função $\alpha : x \in \{0,1\} \mapsto 1 - x \in \{0,1\}$).

n	$f(0)(n)$	$f(1)(n)$	$f(2)(n)$	$\dots$	$f(m)(n)$	$\dots$
0	$\mathbf{1 - f(0)(0)}$	$f(1)(0)$	$f(2)(0)$	$\dots$	$f(m)(0)$	$\dots$
1	$f(0)(1)$	$\mathbf{1 - f(1)(1)}$	$f(2)(1)$	$\dots$	$f(m)(1)$	$\dots$
2	$f(0)(2)$	$f(1)(2)$	$\mathbf{1 - f(2)(2)}$	$\dots$	$f(m)(2)$	$\dots$
$\vdots$	$\vdots$	$\vdots$	$\vdots$		$\vdots$	
m	$f(0)(m)$	$f(1)(m)$	$f(2)(m)$	$\dots$	$\mathbf{1 - f(m)(m)}$	$\dots$
$\vdots$	$\vdots$	$\vdots$	$\vdots$		$\vdots$	

Tabela 7.1: Diagrama para a demonstração do Teorema Diagonal de Cantor.

Uma forma bastante útil desses resultados é o seguinte enunciado, que pode ser considerado como uma contrapositiva do teorema da diagonalização.

Teorema 7.2.3 (Teorema do Ponto Fixo) Suponha que são dados os conjuntos (não vazios) T e Y, e as funções $f : T \times T \to Y$, $\alpha : Y \to Y$ e a função de inclusão na diagonal $\Delta : t \in T \mapsto (t,t) \in T \times T$. Suponha que a função $g : T \to Y$, dada por $g(t) = \alpha \circ f \circ \Delta(t)$, seja representada por $t_g \in T$, ou seja, que $g(t) = f(t, t_g)$, para todo $t \in T$. Então existe $y_g \in Y$ tal que $\alpha(y_g) = y_g$, ou seja, a função α admite um ponto fixo.

Demonstração: Observe- que $g(t_g) = f(t_g, t_g) = \alpha(f(t_g, t_g))$, ou seja, o elemento $y_g = f(t_g, t_g)$ é um ponto fixo da função α. $\square$

Para o tratamento dos teoremas de incompletude, primeiramente estudaremos um pouco da teoria das funções recursivas, que são as funções computáveis por qualquer noção de computação [7]. Em seguida, falaremos sobre a

[7] Esta é a chamada *Tese de Church*, proposta pelo lógico matemático americano Alonzo Church (1903-1995) em 1936 - no artigo *An Unsolvable Problem of Elementary Number*

aritmetização da linguagem objeto (o que ocorre no nível da metalinguagem) como meio de codificá-la e podermos referir a ela na própria linguagem objeto [8]. Por fim, demonstraremos os teoremas de incompletude, em que faremos uso do Teorema do Ponto Fixo, demonstrado para o caso particular que nos interessa, que é da função f que associa ao par de fórmulas $(A(x), B(x))$ a fórmula $A(\ulcorner B(x) \urcorner)$, sendo que $\ulcorner B(x) \urcorner$ é a codificação da fórmula $B(x)$. Teremos que demonstrar a partir de um copnjunto de hipóteses (axiomas de aritmética) várias propriedades dessa diagonalização para que possamos demonstrar os teoremas de incompletude.

Em relação ao segundo teorema da incompletude, que afirma que a aritmética, se for consistente, não poderá demonstrar sua própria consistência, devemos chamar a atenção ao fato de que será possível expressar por uma fórmula da aritmética a noção de que ela é consistente, para que o dito teorema tenha alguma significância.

7.3 Funções recursivas

A Teoria da Recursão é uma formalização da noção intuitiva de "computável" (e de "algoritmo"). Faremos uma breve introdução ao assunto, ressaltando apenas o necessário para os assuntos tratados neste texto.

Uma função $f : \mathbb{N}^n \to \mathbb{N}$ é uma **função primitiva recursiva** [9] se existir uma sequência finita de funções $f_i : \mathbb{N}^{n_i} \to \mathbb{N}$, $1 \leq i \leq m$, tal que f_m é f e cada f_i satisfaz uma das condições abaixo:

- **funções básicas:** f_i é $Z(x) = 0$ (constante igual a zero) ou uma projeção $P^n_j(x_1, \ldots, x_n) = x_j$, ou o sucessor $S(x) = x + 1$, ou
- **composição:** existem $j_1, \ldots, j_{k+1} < i$ tais que f_i é a composição $f_{j_{k+1}}(f_{j_1}, \ldots, f_{j_k})$, ou

Theory, American Journal of Mathhematics, vol. 58 (1936), no. 2, 345-363 - depois de serem demonstradas que todas as propostas do que seria uma função computável coincidiam com as funções recursivas. Leia mais sobre esse assunto na obra de W. Carnielli e R. Epstein, *Computabilidade, funções computáveis, lógica e os fundamentos da matemática*, [6].

[8]Leiam esta frase com as reservas que serão apresentadas em momento oportuno.

[9]Essa noção surgiu com Richard Dedekind em 1888 como generalização natural das definições recursivas da soma e do produto. Consulte sua obra *Essays on the theory of numbers* , Dover Editions, Nova Iorque, 1963.

- **recursão primitiva:** existem $j, k < i$ e $f_i(x_1, \ldots, x_{n_i})$ é definida por recursão primitiva por $f_i(0, x_2, \ldots, x_{n_i}) = f_j(x_2, \ldots, x_{n_i})$ e para cada $r \geq 0$, $f_i(r+1, x_2, \ldots, x_{n_i}) = f_k(r, f_i(r+1, x_2, \ldots, x_{n_i}), x_2, \ldots, x_{n_i})$.

Uma função $f : A \subseteq \mathbb{N}^n \to \mathbb{N}$ é uma **função recursiva** se existir uma sequência de funções f_i como acima, satisfazendo também a cláusula

- **minimização:** existem $j, k < i$ tais que

  $$f_i(x_1, \ldots, x_{n_i}) = \mu_z[f_j(z, x_1, \ldots, x_{n_i}) = f_k(z, x_1, \ldots, x_{n_i})],$$

  sendo que o lado direito da igualdade significa que o valor de $f_i(x_1, \ldots, x_{n_i})$ é o menor número $z \in \mathbb{N}$ tal que vale a igualdade $f_j(z, x_1, \ldots, x_{n_i}) = f_k(z, x_1, \ldots, x_{n_i})$ e que existem os valores de $f_j(w, x_1, \ldots, x_{n_i})$ e $f_k(w, x_1, \ldots, x_{n_i})$ para todo $w \leq z$.

Uma relação $R \subseteq \mathbb{N}^n$ é respectivamente primitiva recursiva, ou recursiva, se sua função característica $\chi_R(\bar{x}) = 1$ se $\bar{x} \in R$ e $\chi_R(\bar{x}) = 0$ se $\bar{x} \notin R$ for respectivamente primitiva recursiva ou recursiva. Uma relação recursiva também é chamada de **decidível**.

Exemplo 7.3.1 Exemplos importantes de funções primitivas recursivas (preencha os detalhes e justifique as afirmações não óbvias, como exercício):

1. $f(a, b) = a + b$: podemos obtê-la pela sequência de construção $f(a, 0) = a$, $f(a, b+1) = S(f(a, b))$;

2. $f(a, b) = a \cdot b$: $f(a, 0) = 0$, $f(a, b+1) = f(a, b) + a$;

3. $f(a, b) = a^b$: construção $f(a, 0) = 1$, $f(a, b+1) = a \cdot f(a, b)$;

4. $f(a) = a!$ (fatorial): construção $f(0) = 1$, $f(a+1) = (a+1) \cdot f(a)$;

5. $f(a) = a \dot{-} 1 = \max\{a - 1, 0\}$: $f(0) = 0$, $f(a+1) = a$;

6. $f(a, b) = a \dot{-} b = \max\{a - b, 0\}$: $f(a, 0) = a$, $f(a, b+1) = f(a, b) \dot{-} 1$;

7. $f(a, b) = |a - b|$ (valor absoluto): $f(a, b) = (a \dot{-} b) + (b \dot{-} a)$;

8. $f(a, b) = \max\{a, b\}$: $f(a, b) = (a \dot{-} b) + b$;

9. $f(a, b) = \min\{a, b\}$: $f(a, b) = (a + b) - \max\{a, b\}$;

10. $f(a) = \mathsf{sg}(a) = \chi_{>0}(a)$ (a função característica dos números estritamente positivos): $f(0) = 0$, $f(a+1) = 1 = S(0)$;

11. $f(a,b) = \chi_{<}(a,b)$: $f(a,b) = \mathsf{sg}(b \dot{-} a)$;

12. $f(a,b) = \chi_{\leq}(a,b)$: $f(a,b) = \chi_{<}(a, S(b))$.

Exemplo 7.3.2 Vamos usar as funções do exemplo anterior para fazer construções mais complexas.

1. Seja $g(i,\bar{b})$ primitiva recursiva, $\bar{b} = b_1, \ldots, b_n$; então as funções $f(a,\bar{b}) = \sum_{i=0}^{a} g(i,\bar{b})$ e $h(a,\bar{b}) = \prod_{i=0}^{n} g(i,\bar{b})$ são primitivas recursivas: $f(0,\bar{b}) = h(0,\bar{b}) = g(0,\bar{b})$; $f(a+1,\bar{b}) = f(a,\bar{b}) + g(a,\bar{b})$ e $h(a+1,\bar{b}) = h(a,\bar{b}) \cdot g(a+1,\bar{b})$;

   No caso em que a soma ou o produto são contados a partir de $i = 1$, definimos os casos iniciais como $f(0,\bar{b}) = 0$ e $h(0,\bar{b}) = 1$.

2. $f(a,b) = a \div b$:

$$f(a,b) = \mathsf{sg}(b) \cdot \sum_{k=0}^{a} \mathsf{sg}\left(\prod_{j=0}^{k}((a+1) \dot{-} (j+1) \cdot b)\right).$$

3. $f(a,b) = a \bmod b$ (resto da divisão de a por b): $f(a,b) = a \dot{-} (a \div b) \cdot b$;

4. $f(a,b) = \binom{a}{b}$ (números de combinações de a, b a b, sem repetições): $\binom{a}{b} = \chi_{\leq}(b,a) \cdot \big((a!) \div (b! \cdot (a \dot{-} b)!)\big)$.

5. $\operatorname{div}(a,b) = 1 \dot{-} \mathsf{sg}(a \bmod b)$ é a função característica da relação b divide a;

6. $D(a) = \sum_{i=1}^{a} \operatorname{div}(a,i)$ conta o número de divisores de a;

7. $\chi_{\text{primos}}(a) = 1 \dot{-} \mathsf{sg}(|D(a) \dot{-} 2|)$ é a função característica do conjunto dor números primos;

8. $\pi(a) = \sum_{i=2}^{a} \chi_{\text{primos}}(i)$ diz o número de primos até a;

9. $f(n) = p_n$ (o n-ésimo número primo em ordem crescente): para verificar que esta função é primitiva recursiva, precisamos da desigualdade $p_n < F_n = 2^{2^n} + 1$; F_n é chamado do n-ésimo número de Fermat; observe

que se m divide F_n e F_{n+k}, $k > 0$, como F_n divide $F_{n+k} - 2$ (verifique), m divide 2; portanto $m = 1$, pois F_n é ímpar; portanto $F_0, \ldots, F_n$ são primos entre si; portanto existem pelo menos n primos ímpares (que dividem of F_j) até F_n; finalmente,

$$f(n) = p_n = \sum_{k=0}^{F_n} \mathsf{sg}\left(\prod_{j=0}^{k} |n + 1 - \pi(j)|\right);$$

10. $f(a, n) = b$, sendo que b é o maior expoente do primo p_n tal que p_n^b divide a, se $a \neq 0$, e $f(0, n) = 0$: $f(a, n) = \mathsf{sg}(a) \cdot \sum_{i=1}^{a} \mathrm{div}(a, p_n^i)$;

11. $f(a) = [\sqrt{a}]$ (a parte inteira da raiz quadrada de a): temos $f(a) = \sum_{i=1}^{a} \chi_{\leq}(i^2, a)$;

12. a função (evidentemente primitiva recursiva)

$$f(m, n) = \frac{(m + n)(m + n + 1)}{2} + n;$$

define uma bijeção de $\mathbb{N}^2$ sobre $\mathbb{N}$ (verifique); sejam $\pi_1(a) = m$ e $\pi_2(a) = n$ as funções tais que $\pi_1(f(m, n)) = m$ e $\pi_2(f(m, n)) = n$; então π_1 e π_2 são primitivas recursivas; por exemplo, $\pi_1(0) = 0$ e

$$\pi_1(a + 1) = (\pi_1(a) \dot{-} 1) \cdot \mathsf{sg}(\pi_1(a)) + (a + 1) \cdot (1 - \mathsf{sg}(\pi_1(a)));$$

para π_2, $\pi_2(0) = 0$ e $\pi_2(a + 1) = (\pi_2(a) + 1) \cdot \mathsf{sg}(\pi_1(a))$.

Exercício 7.3.1 Mostre que as funções descritas abaixo são todas recursivas primitivas.

1. a função característica da ordem lexicográfica de $\mathbb{N}^n$ (isto é, $(a_1, \ldots, a_n) < (b_1, \ldots, b_n)$ se existir $k < n$ tal que $a_k < b_k$ e $a_i = b_i$, para todo $i < k$) é primitiva recursiva;

2. defina uma relação primitiva recursiva $\prec$ em $\mathbb{N}^2$, que represente a ordem lexicográfica em todas as sequências finitas de números (por exemplo, usando expoentes de primos em fatorações de números);

3. $f(m) = m \oplus 1$, o menor elemento n, tal que $m \prec n$ (e $m \neq n$) é primitiva recursiva.

Teorema 7.3.1 (Outra Construção das Funções Primitivas Recursivas) Todas as funções primitivas recursivas de uma variável podem ser obtidas a partir de $Z(x) = 0$, $S(x) = x + 1$, $Q(x) = x \dot{-} 1$, $\pi_1(x)$ e $\pi_2(x)$ (que calculam a primeira e a segunda coordenadas de uma dupla ordenada codificada por um número x), aplicando as regras $f(x) = g(x) + h(x)$, $f(x) = g(x) \cdot h(x)$, $f(x) = g(h(x))$ e $f(x) = g^x(0)$, sendo que $g(x)$ e $h(x)$ já tenham sido definidas e $g^x(0)$ é definida por $g^0(0) = 0$, $g^{x+1}(0) = g(g^x(0))$.

Demonstração: Mostraremos como reduzir uma sequência de funcões primitivas recursivas $f_i : \mathbb{N}^{n_i} \to \mathbb{N}$, $1 \leq i \leq m$, tal que $n_m = 1$, que constŕoi a função $f_m(x)$ numa outra que também constrói $f(x)$, mas apenas com funções de uma variável. Isto é feito codificando-se n-uplas de variáveis numa única variável. As funções $\mathsf{sg}(x)$ e $\overline{\mathsf{sg}}(x) = 1 - \mathsf{sg}(x)$ são definidas pela regra de recursão do enunciado, usando-se a função constante 1 e a função $x \dot{-} 1$.

Observe que a partir das funções π_1 e π_2, podemos definir, para cada $n \geq 2$ e $1 \leq p \leq n$, funções $\Pi^n_p(x)$ por $\Pi^2_1 = \pi_1$, $\Pi^2_2 = \pi_2$, e, supondo definidas Π^n_p, para todo p, $1 \leq p \leq n$, e $n \geq 2$, definimos Π^{n+1}_q, $1 \leq q \leq n + 1$, como $\Pi^{n+1}_1 = \pi_1$ e $\Pi^{n+1}_{q+1} = \Pi^n_q \circ \pi_2$, $1 \leq q \leq n$. Ou seja, olhamos um número x como codificando um par ordenado, cuja segunda coordenada codifica uma $n - 1$ upla. Observe que as funções $\Pi^n_p(x)$ são obtidas de π_1 e π_2 usando apenas composições. Para definir cada $\Pi^n_p(x)$, precisamos de uma sequência de n composições.

Usando composições, somas e produtos, para cada par de funções $u(x)$ e $v(x)$, podemos construir a função

$$w(x) = W(u(x), v(x)) = \frac{(u(x) + v(x)) \cdot (u(x) + v(x) + 1)}{2} + v(x)$$

Sem perda de generalidade, assumiremos que isto define uma regra básica de construção deste teorema. e denotaremos $W(u, v) = \langle u, v\rangle$, e (indutivamente) denotamos $\langle u_1, \ldots, u_n\rangle = \langle u_1, \langle u_2, \ldots, u_n\rangle\rangle$, e observamos que sua construção pode ser feita em $n - 1$ passos, a partir das funções u_i e a regra W.

Sejam $f_1, \ldots, f_n$ funções primitivas recursivas, tais que descrevem a construção de uma função f_n unária. Vamos construir uma sequência de funções unárias $g_1, \ldots, g_m$, respeitando as regras do enunciado do teorema e tal que $g_m = f_n$.

Suponha que já tenhamos tratado de f_i, $i < j \le n$, e tenhamos obtido a sequência g_k, $k < l$.

Se f_j for $Z(x)$ ou $S(x)$, definimos $g_l = f_j$, chamamos $g_l = f_j^*$, e passamos a tratar o caso $j+1$.

Se f_j for $P_p^n(x_1, \ldots, x_n) = x_p$, com $n \ge 2$ e $1 \le p \le n$, sejam $g_l, \ldots, g_{l+n-1}$ a sequência de composições definindo $g_{l+n-1} = \Pi_p^n = f_j^*$, e passamos a tratar o caso $j+1$.

Se existem $a, b < j$, tal que f_j é a composição $f_a(f_b)$, então $g_l = f_a^*(f_b^*)$, e passamos a tratar o caso $j+1$.

Se existem $j_1, \ldots, j_{a+1} < j$, $a \ge 1$, tais que f_j é a composição $f_{j_{a+1}}(f_{j_1}, \ldots, f_{j_a})$, sejam $g_l = \langle f^*_{j_a-1}, f^*_{j_a}\rangle$, $g_{l+1} = \langle f^*_{j_a-2}, g_l\rangle, \ldots, g_{l+a-1} = \langle f_{j_1}, g_{l+a-2}\rangle$, e $g_{l+a} = f_j^*(g_{l+a-1})$, e passamos a tratar o caso $j+1$.

Se existem $a, b < j$ e $f_j(x_1, \ldots, x_n)$ é definida por recursão primitiva por $f_j(0, x_2, \ldots, x_n) = f_a(x_2, \ldots, x_n)$ e para cada $r \ge 0$, $f_j(r+1, x_2, \ldots, x_n) = f_b(r, f_j(r, x_2, \ldots, x_n), x_2, \ldots, x_n)$, então definimos as funções, $G_0(x) = \overline{\mathsf{sg}}(\Pi_1^{n+1}(x)) \cdot f_a^*(\langle \Pi_3^{n+1}(x), \ldots, \Pi_{n+1}^{n+1}(x)\rangle) + \mathsf{sg}(x) \cdot f_b^*(\langle \Pi_1^{n+1}(x) \dot{-} 1, \Pi_2^{n+1}(x), \ldots, \Pi_{n+1}^{n+1}(x)\rangle)$, $\beta(x) = \langle \Pi_1^{n+1}(x+1), G_0(x), \Pi_3^{n+1}(x), \ldots, \Pi_{n+1}^{n+1}(x)\rangle$, e $\varphi(x)$ definida por $\varphi(0) = 0$ e $\varphi(n+1) = \beta^n(0)$, e, por fim, $f_j^*(x) = \Pi_2^{n+1}(\varphi(x \dot{-} 1)$. □

Exemplos de funções recursivas que não são primitivas recursivas.

Exemplo 7.3.3 (A função de Ackermann) O lógico e matemático alemão Wilhelm Ackermann publicou um artigo em 1928 [10] exibindo um exemplo de uma função que tinha uma construção recursiva e que não é primitiva recursiva. Na verdade, para defini-la, é necessária uma dupla recursão não capturada pela recursão primitiva simples. Esta é a sua função: $f(0, y) = y+1$, $f(x+1, 0) = f(x, 1)$, $f(x+1, y+1) = f(x, f(x+1, y))$. Para mostrarmos que é recursiva, sejam

$$f_0(x, y, z, v) = \operatorname{div}(v, p_{2^x \cdot 3^y \cdot 5^z}),$$

$$f_1(x, y, z, v) = (1 - \mathsf{sg}(x)) \cdot (1 - \mathsf{sg}(|y + 1 - z|) \cdot f_0(0, y, z, v) +$$

[10] Veja a tradução de seu artigo em Jan van Heijenoort, *From Frege to Gödel*, [?], pp. 493-507.

$$+ \,\mathsf{sg}(x) \cdot (1 - \mathsf{sg}(y)) \cdot f_0(x-1, y, z, v) \cdot f_0(x, y, z, v) +$$

$$+ \,\mathsf{sg}(x) \cdot \mathsf{sg}(y) \cdot \mathsf{sg}\left(\sum_{u=0}^{z} f_0(x, y-1, u, v) \cdot f_0(x-1, u, z, v)\right),$$

$$f_2(x, y, v) = \mathsf{sg}\left(\sum_{z=0}^{v} f_1(x, y, z, v)\right)$$

(que vale 1 se existir $z \leq v$, tal que $f_1(x, y, z, v) = 1$, e 0, caso contrário),

$$f_3(x, y) = \mu_v(f_2(x, y, v) = 1)$$

e, finalmente,

$$f(x, y) = \mu_z(f_1(x, y, z, f_3(x, y)) = 1).$$

Agora veremos que $f(x, y)$ não é primitiva recursiva. Observe que $f(x, y) > y$, $f(x, y+1) > f(x, y)$, $f(x+1, y) > f(x, y)$ e $f(x+1, y+1) \geq f(x, y+2)$, para todo x e y (verifique).

Teorema 7.3.2 A função de Ackermann não é primitiva recursiva.

Demonstração: Se $f(x, y)$ é a função de Ackermann, mostraremos que para toda função primitiva recursiva $g(x)$ de uma variável, existe y tal que $g(x) < f(y, x)$, para todo x. Com isto, se $f(y, x)$ fosse primitiva recursiva, $h(x) = f(x, x)$ também o seria, donde existiria y, tal que $h(x) < f(y, x)$; em particular, $f(y, y) = h(y) < f(y, y)$, absurdo.

Observe que $f(0, n) = n + 1$, $f(1, n) = n + 3$, $f(2, n) = 3n + 3$, $f(3, n) = 6 \cdot 3^n + 3$. Então $y = 3$ garante que se $g(x)$ á uma das funções básicas $Z(x)$, $S(x)$, $Q(x)$, $\pi_1(x)$, $\pi_2(x)$ é majorada por $f(3, x)$.

As regras de soma e produto de funções são facilmente majoradas. Por exemplo, se $g(x) < f(y, x)$ e $h(x) < f(z, x)$, como f é crescente nas duas variáveis, tomando o máximo entre y e z, podemos supor que $y = z$. Daí, $g(x) \cdot h(x) < f(y, x)^2 < f(3, f(y, x)) < f(y + 3, x)$.

Se $g(x)$ é definida por $g(0) = 0$, $g(x+1) = \beta^x(0)$ e $\beta(x) < f(y, x)$, então $g(x+1) = \beta(g(x)) < f(y, g(x))$; vamos mostrar que $g(x) < f(y+1, x)$, por indução em x; para $x = 0$, temos que para todo z, $g(0) = 0 < f(z, 0)$; portanto $g(0) < f(y+1, 0)$. suponha que $g(x) < f(y+1, x)$. Então $g(x+1) < f(y, g(x)) < f(y, f(y+1, x)) = f(y+1, x+1)$. Com isto, demonstramos o teorema. $\square$

Exercício 7.3.2 Mostre que a função de Ackermann também majoriza as funções primitivas recursivas de várias variáveis. Ou seja, mostre que se $g(x_1, \ldots, x_n)$ é primitiva recursiva, então existe y, tal que

$$g(x_1, \ldots, x_n) < f(y, \max(x_1, \ldots, x_n)).$$

Exemplo 7.3.4 Uma enumeração recursiva das funções primitivas recursivas. Vamos definir uma função recursiva $F(m, n)$ que enumera todas as funções primitivas recursivas de uma variável (com infinitas repetições), ou seja, $F(m, n) = f_m(n)$ é primitiva recursiva, e se $g(n)$ for primitiva recursiva, existe pelo menos um número $m \in \mathbb{N}$, tal que $F(m, n) = g(n)$. Para mostrar que F não pode ser primitiva recursiva, suponha que seja. Então $F(n, n) + 1 = g(n)$ é primitiva recursiva e, portanto, existe m, tal que $F(m, n) = g(n)$. Calculando em m, temos $F(m, m) + 1 = g(m) = F(m, m)$, o que é absurdo. (Este método de listar os valores $F(m, m) = f_m(m)$ e alterar o valor para obter nova função $g(m) = F(m, m) + 1$ é chamado de **diagonalização**, e está no centro dos argumentos de incompletude e de indecidibilidade.

Eis a função:

$$F(m,n) = \begin{cases} 0 & \text{se} \quad m = 9a,\ a \in \mathbb{N} \\ S(n) & \text{se} \quad m = 9a+1,\ a \in \mathbb{N} \\ n \dot{-} 1 & \text{se} \quad m = 9a+2,\ a \in \mathbb{N} \\ \pi_1(n) & \text{se} \quad m = 9a+3,\ a \in \mathbb{N} \\ \pi_2(n) & \text{se} \quad m = 9a+4,\ a \in \mathbb{N} \\ F(\pi_1(a,n)) + F(\pi_2(a)) & \text{se} \quad m = 9a+5,\ a \in \mathbb{N} \\ F(\pi_1(a,n)) \cdot F(\pi_2(a)) & \text{se} \quad m = 9a+6,\ a \in \mathbb{N} \\ F(\pi_1(a), F(\pi_2(a), n)) & \text{se} \quad m = 9a+7,\ a \in \mathbb{N} \\ g^n(0) & \text{se} \quad m = 9a+8,\ a \in \mathbb{N}, \\ & \qquad\ \ \text{e } g(x) = F(a, x) \end{cases}$$

Exercício 7.3.3 Mostre que $F(m, n)$ é recursiva. (Imite a demonstração feita para a função de Ackermann.)

Lema 7.3.1 A função $F(m, n)$ enumera todas as funções primitivas recursivas de uma variável.

Demonstração: Vimos que as funções primitivas recursivas de uma variável são obtidas a partir das funções $Z(x)$, $S(x)$, $Q(x) = x \dot{-} 1$, $\pi_1(x)$, $\pi_2(x)$, usando as regras $f(x) + g(x)$, $f(x) \cdot g(x)$, $f(g(x))$ e $f(x) = \beta^x(0)$.

Obviamente, as funções $Z(x)$, $S(x)$, $Q(x) = x \dot{-} 1$, $\pi_1(x)$, $\pi_2(x)$ são enumeradas. Suponha, por indução, que $f(n) = f_m(n)$ e $g(n) = f_p(n)$. Então $f(n)+g(n) = F(9\cdot\langle m,p\rangle+5,n)$, $f(n)\cdot g(n) = F(9\cdot\langle m,p\rangle+6,n)$, $f(g(n)) = F(9\cdot\langle m,p\rangle+7,n)$, e $h(n) = f^n(0) = F(9\cdot m+8,n)$. □

Exercício 7.3.4 Mostre que se $G(m,n)$ é recursiva e $g_m(n) = G(m,n)$ é uma sequência de funções "enumeradas" por G, então existe uma função recursiva $h(n)$ que não é enumerada por G. (Use o método de diagonalização.) Conclua que não existe enumeração recursiva de todas as funções recursivas $f : \mathbb{N} \to \mathbb{N}$.

Para perceber o que está dito no exercício acima, estendemos a noção de função recursiva para $f : A \to \mathbb{N}$ é **recursiva parcial**, $A \subseteq \mathbb{N}^k$, se é obtida pelas funções iniciais e regras de recursão primitiva e de minimização, $f(x,y) = \mu_z[g(x,y,z) = h(x,y,z)]$, só que agora sem restringirmos a minimização à existência de solução em z de $g(x,y,z) = h(x,y,z)$, para todo x e y. Com isto, podem existir x e y, tais que $g(x,y,z) \neq h(x,y,z)$ sempre. Neste caso, $A = \{(x,y) : \text{existe } z, \text{ tal que } g(x,y,z) = h(x,y,z)\}$ é o domínio de f.

Exercício 7.3.5 Mostre que existe uma função recursiva $H(m,n)$, definida para todo $m \in \mathbb{N}$, mas nem todos $n \in \mathbb{N}$, tal que enumera todas as funções recursivas parciais de uma variável.

Exercício 7.3.6 O problema da parada. Este problema pergunta se podemos decidir se, dado $m \in \mathbb{N}$, a função $f_m(n) = H(m,n)$ está definida para todo $n \in \mathbb{N}$. Isto é mais sensível na regra de minimização, em que, dados $x, y \in \mathbb{N}$, precisaríamos decidir se existe $z \in \mathbb{N}$, tal que $f(x,y,z) = g(x,y,z)$, para f e g recursivas. Essencialmente, parguntamos se a busca por tal z, partindo de $z_0 = 0$ e testando para cada $z_{n+1} = S(z_n)$ até que encontremos o número z que resolva a equação, pára. Usando a função H acima, mostre que o conjunto $R = \{m \in \mathbb{N}\text{: para todo } n \in \mathbb{N},\ H(m,n) \text{ está definida}\}$ não é recursivo.

Um conjunto $A \subseteq \mathbb{N}$ é **recursivamente enumerável** se existe uma função recursiva $f : \mathbb{N} \to \mathbb{N}$, tal que A é a imagem de f. Um conjunto $A \subseteq$

$\mathbb{N}^k$ é recursivamente enumerável se existem funções recursivas $f_i : \mathbb{N} \to \mathbb{N}$, $i = 1, \ldots, k$, tal que $A = \{(a_1, \ldots, a_k) \in \mathbb{N}^k : a_1 = f_1(n), \ldots, a_k = f_k(n)$, para algum $n \in \mathbb{N}\}$.

Exercício 7.3.7 Mostre que se A é recursivo, então é recursivamente enumerável

Exercício 7.3.8 Seja $A \subset \mathbb{N}^2$ o gráfico de uma função recursiva $f : \mathbb{N} \to \mathbb{N}$. Mostre que A é um conjunto primitivo recursivo. (Faça isto por indução na construção de f.

Exercício 7.3.9 Mostre que um conjunto $R \subseteq \mathbb{N}^k$ é recursivamente enumerável se, e só se, for o domínio de uma função recursiva (total ou parcial).

7.4 Aritmetização da linguagem

A grande ideia que Kurt Gödel teve para demonstrar seus teoremas de incompletude foi poder codificar na linguagem objeto da aritmética de primeira ordem algumas noções metamatemáticas, como a de fórmula e dedução formal. Na verdade, somente na metalinguagem é que se pode dizer que tais funções e fórmulas realmente codificam esses conceitos. Assim, a linguagem objeto reflete algo da metamatemática e, a menos que os pressupostos (axiomas) metamatemáticos sejam inconsistentes, demonstraremos que no nível da linguagem objeto não se pode definir o conceito metamatemático de *verdade*. Intuitivamente [11] falando, a percepção de que existe tal codificação de conceitos metamatemáticos na linguagem objeto somente ocorre para quem vive na metalinguagem e não para os que vivem na linguagem objeto.

Numa situação genérica, a linguagem objeto terá sua assinatura um subconjunto de $mathscrL = \mathcal{C} \cup \mathcal{F} \cup \mathcal{R}$, com $\mathcal{C} = \{\mathfrak{c}_n : n \in \mathbb{N}\}$, $\mathcal{F} = \bigcup_{n\geq 1} \mathcal{F}_n = \bigcup_{n\geq 1}\{\mathfrak{f}_{m,n} : m \in \mathbb{N}\}$ e $\mathcal{R} = \bigcup_{n\geq 1} \mathcal{R}_n = \bigcup_{n\geq 1}\{\mathfrak{r}_{m,n} : m \in \mathbb{N}\}$.

Na metamatemática, definimos uma função $\mathfrak{Cod} : \mathcal{L} \to \mathbb{N}$, que associa $\mathfrak{c}_n \in \mathcal{C} \mapsto c_n \in \mathbb{N}$, $\mathfrak{f}_{m,n} \in \mathcal{F}_n \mapsto f_{m,n} \in \mathbb{N}$ e $\mathfrak{r}_{m,n} \in \mathcal{R}_n \mapsto r_{m,n} \in \mathbb{N}$, enumerados como se segue. Definimos as funções $v, c : \mathbb{N} \to \mathbb{N}$, $r, f : \mathbb{N}^2 \to \mathbb{N}$, denotando $v(n) = v_n$, $c(n) = c_n$, $r(m, n) = r_{n,m}$ e $f(m, n) = f_{n,m}$, por

$$v_n = 8n + 25,\ c_n = 8n + 3,\ r_{n,m} = 8\left(\frac{(n+m)(n+m+1)}{2} + m\right) + 5,$$

[11] E, portanto, uma visão falsa se levada às últimas consequências.

$$f_{n,m} = 8\left(\frac{(n+m)(n+m+1)}{2} + m\right) + 7.$$

Exercício 7.4.1 Mostre que estas funções são primitivas recursivas e que suas imagens são disjuntas.

A intenção desta definição é enumerar "códigos" para variáveis, símbolos de constantes, símbolos de relações e de funções n-árias, respectivamente. Mais precisamente, dada uma linguagem (cuja assinatura seja finita ou infinita enumerável) L, uma **aritmetização** de L é uma tripla de funções injetoras (Φ_C, Φ_R, Φ_F), tal que Φ_C (respectivamente, Φ_R e Φ_F) associa a cada símbolo de constante (respectivamente, relação, função $n+1$-ária) um número da forma c_n (respectivamente, $r_{n,m}$, $f_{n,m}$).

Seja p_n, $n \in \mathbb{N}$ a enumeração (primitiva recursiva) de todos os números primos em ordem crescente, com $p_0 = 2$.

Definimos as funções $\neg : a \in \mathbb{N} \mapsto \neg a = 2^1 \cdot 3^a \in \mathbb{N}$, $\rightarrow : (a,b) \in \mathbb{N}^2 \mapsto a \rightarrow b = 2^9 \cdot 3^a \cdot 5^b \in \mathbb{N}$, e $\forall : (a,b) \in \mathbb{N}^2 \mapsto \forall a\, b = 2^{17} \cdot 3^a \cdot 5^b$. Dados $a, b_0, \ldots, b_n \in \mathbb{N}$, $a[b_0, \ldots, b_n]$ denota o número $p_0^a \cdot p_1^{b_0} \cdot \ldots \cdot p_{n+1}^{b_n}$.

Seja $K \subseteq \mathbb{N}$ um conjunto finito, contendo $r_{1,0}$ e apenas números da forma c_n, $r_{n,m}$, ou $f_{n,m}$. Tal K representa a assinatura de alguma linguagem L.

Definimos o conjunto Tr_K (dos termos de L) por: se $c_n \in K$, $c_n \in \mathsf{Tr}$; $v_n \in \mathsf{Tr}$, $n \in \mathbb{N}$; se $b_0, ..., b_n \in Tr$ e $f_{n,m} \in K$, $f_{n,m}[b_0, \ldots, b_n] \in Tr$.

Definimos o conjunto At_K (das fórmulas atômicas) como o conjunto dos números da forma $r_{n,m}[t_0, \ldots, t_n]$ e para todo $r_{n,m} \in K$ e $t_0, \ldots, t_n \in \mathsf{Tr}_K$.

Definimos o conjunto Fla_K (das fórmulas) como o menor conjunto contendo At_K e fechado por $a \rightarrow b$, $\neg a$ e $\forall a\, b$.

Exercício 7.4.2 Mostre que os conjuntos Tr_K, At_K e Fla_K são primitivos recursivos. Exiba uma enumeração recursiva de cada um destes conjuntos.

Uma **teoria** numa linguagem de assinatura L (ou L-teoria) é um conjunto consistente T de sentenças de L. A teoria T é uma **teoria completa** se para cada sentença ϕ, ou $T \vdash \phi$ ou $T \vdash \neg\phi$ (mas não ambas, devido à condição de ser consistente).

Exemplo 7.4.1 Dada uma estrutura M, a teoria $T(M) = \{\phi$: *ϕ é uma L-sentença e* $M \models \phi\}$ é uma teoria completa.

Uma teoria T é (recursivamente) axiomatizável se existir um conjunto de sentenças Σ finito ou infinito mas recursivo (os axiomas ou postulados de T), tal que $\Sigma \vdash \phi$ se, e só se, $T \vdash \phi$.

Agora vamos fixar uma linguagem $L = \{0, +, \cdot, S(\cdot), \leq\}$ e a L-estrutura $\mathbb{N}$ dos números naturais com a interpretação usual dos símbolos de L e as teorias Q e PA descritas a seguir. Usando a aritmetização acima, "0" é $c_0 = 3$, "+" é $f_{1,0} = 15$, $\cdot$ é $f_{1,1} = 39$, S é $f_{0,0} = 7$, "=" é $r_{1,0} = 13$ e "$\leq$" é $r_{1,1} = 29$.

Exemplo 7.4.2 A Teoria Q de R. M. Robinson [12]: Esta teoria, que é um fragmento bem fraco, mas suficientemente expressivo, de sentenças verdadeiras em $\mathbb{N}$, é axiomatizada pelo conjunto das oito sentenças:

(Q1) $\forall x\,(S(x) \neq 0)$

(Q2) $\forall x\,\forall y\,(S(x) = S(y) \to x = y)$

(Q3) $\forall x\,(x \neq 0 \to \exists y\,(x = S(y)))$

(Q4) $\forall x\,(x + 0 = x)$

(Q5) $\forall x\,\forall y\,(x + S(y) = S(x + y))$

(Q6) $\forall x\,(x \cdot 0 = 0)$

(Q7) $\forall x\,\forall y\,(x \cdot S(y) = (x \cdot y) + x)$

(Q8) $\forall x\,\forall y(x \leq y \iff \exists z\,(z + x = y))$

Exemplo 7.4.3 A Aritmética de Peano. também chamada pela sigla PA, é a teoria contendo as sentenças (Q1) a (Q8) da aritmética de Robinson e o esquema de axiomas de indução para fórmulas φ com variáveis livres x_0, $\ldots$, x_n

(Ind(φ)) $\forall x_1 \ldots \forall x_n\,[\varphi(0) \to (\forall x_0(\varphi(x_0) \to \varphi(S(x_0))) \to \forall x_0\,\varphi(x_0))]$.

Uma fórmula φ é dita limitada se todas as quantificações que aparecem nela são da forma $\forall x(x \leq t \to \theta)$ ou $\exists x(x \leq t \wedge \theta)$, sendo que t é um termo em que a variável x não ocorre. O conjunto das fórmulas limitadas

[12] Raphael Mitchel Robinson (1911-1995) expôs esta teoria no Congresso Internacional de Matemática de 1950, publicado como *An Essentially Undecidable Axiom System*, Proceedings of the International Congress of Mathematicians, Cambridge, MA, 1950 (Graves et al, editors), American Mathematical Society, Providence, RI, pp. 729-730. Publicou-o também na obra **Undecidable Theories**, North-Holland, Amsterdam, (com A. Mostowski and A. Tarski)

é denotado por Δ_0 (e, às vezes por Π_0 ou Σ_0). Definimos os conjuntos de fórmulas Σ_n, Π_n e Δ_n, por $\phi \in \Sigma_{n+1}$ (respectivamente, Π_{n+1}) se existir uma fórmula $\psi \in \Pi_n$ (respectivamente, Σ_n), tal que ϕ é lógicamente equivalente a $\exists x_1 \ldots \exists x_n\, \psi$ ((respectivamente, $\forall x_1 \ldots \forall x_n\, \psi$). Definimos $\Delta_n = \Sigma_n \cap \Pi_n$.

Exemplo 7.4.4 Fragmentos da Aritmética de Peano. Restringindo o esquema de indução a certos conjuntos de fórmulas, obtemos fragmentos importantes de PA:

(IO) Indução aberta: φ em (Ind(φ)) não tem quantificadores.

(IΔ_0) Indução limitada (também chamado de Aritmética Primitiva Recursiva): $\varphi \in \Delta_0$ em (Ind(φ))

(IΣ_n) $\varphi \in \Sigma_n$ em (Ind(φ)).

(IΠ_n) $\varphi \in \Pi_n$ em (Ind(φ)).

Exercício 7.4.3 Seja Γ um destes conjuntos de axiomas para fragmentos da aritmética, já codificados como conjunto de números. Mostre que Γ é primitivo recursivo.

Dada uma teoria $T \subseteq T(\mathbb{N})$, e uma função $f : A \subseteq \mathbb{N}^n \to \mathbb{N}$, dizemos que f é **representável** em T se existe uma fórmula $\phi_f(x_1, \ldots, x_n, y)$, tal que as variáveis $x_1, \ldots, x_n$ e y ocorrem livres em ϕ_f, e para cada $\bar{a} \in \mathbb{N}^n$ e $b \in \mathbb{N}$, $f(\bar{a}) = b$ se, e só se, $T \vdash \phi_f(\tilde{\bar{a}}, \tilde{b})$ (sendo $\tilde{n}$ o termo $(1 + (1 + \ldots + 1)\ldots))$ em que 1 aparece n vezes, para cada $n \in \mathbb{N}$), e $T \vdash \exists! y \phi_f(\bar{a}, y)$. Uma relação $P \subseteq \mathbb{N}^n$ é **expressível** em T se existir uma fórmula $\phi_P(\bar{x})$ tal que para todo $\bar{a} \in \mathbb{N}^n$, se $\bar{a} \in P$, $T \vdash \phi_P(\tilde{\bar{a}})$ e se $\bar{a} \notin P$, $T \vdash \neg\phi_P(\tilde{\bar{a}})$. Uma relação $P \subseteq \mathbb{N}^n$ é **fracamente expressível** em T se existir uma fórmula $\phi_P(\bar{x})$ tal que para todo $\bar{a} \in \mathbb{N}^n$, se $\bar{a} \in P$, $T \vdash \phi_P(\tilde{\bar{a}})$ e se $\bar{a} \notin P$, $T \not\vdash \phi_P(\tilde{\bar{a}})$. Uma relação $P \subseteq \mathbb{N}^n$ é **definível** em $\mathbb{N}$ se existir uma fórmula $\phi_P(\bar{x})$ tal que para todo $\bar{a} \in \mathbb{N}^n$, $\bar{a} \in P$ se, e só se, $\mathbb{N} \models \phi_P(\bar{a})$.

Para cada $n \in \mathbb{N}$, definimos o termo $\bar{n}$ como $\bar{0} = 0$, $\overline{n+1} = S(\overline{n})$.

Lema 7.4.1 A partir da teoria Q podemos deduzir as seguintes fórmulas:

1. $x + y = \bar{0} \to (x = \bar{0} \land y = \bar{0})$
2. $x \cdot y = \bar{0} \to (x = \bar{0} \lor y = \bar{0})$
3. $x + \bar{1} = S(x)$

4. $\bar{0} \leq x$

5. $S(x) \leq \overline{n+1} \to x \leq \bar{n}$

6. $S(x) + \bar{n} = x + \overline{n+1}$

7. $\bar{n} \leq x \to (x = \bar{n} \vee \overline{n+1} \leq x)$

8. $\bar{m} + \bar{n} = \overline{m+n}$

9. $\bar{m} \cdot \bar{n} = \overline{m \cdot n}$

10. $\bar{m} \neq \bar{n}$, se $m \neq n$, $m, n \in \mathbb{N}$

11. $x \leq \bar{n} \iff (x = \bar{0} \vee x = \bar{1} \vee \ldots \vee x = \bar{n})$

12. $x \leq \bar{n} \vee \bar{n} \leq x$

Demonstração: Vamos mostrar os itens (1), (5) a (8), (10) e (11), deixando os outros como exercício.

(1) Se $y \neq \bar{0}$, por (Q3), $Y = S(z)$, para algum z; por (Q4) e (Q1), $x + y = x + S(z) = S(x + z) \neq \bar{0}$.

(5) Usando (3), (Q8) e (Q5), temos $z + S(x) = \overline{n+1}$, então $S(z + x) = S(\bar{n})$, donde, por (Q2), $x + z = \bar{n}$.

(6) Vamos demonsmostrá-lo por indução em n. Para $n = 0$, por Q4 e (3), $S(x) + \bar{0} = S(x) = x + \bar{1}$. Supnha que de Q se deduz que $S(x) + \bar{n} = x + \overline{n+1}$. Então $S(x) + \overline{n+1} = S(x) + S(\bar{n}) = S((S(x) + \bar{n}) = S(x + \overline{n+1}) = x + S(\overline{n+1}) = x + \overline{n+2}$.

(7) Suponha que $\bar{n} \leq x$, mas que $x \neq \bar{n}$. Seja z, tal que $z + \bar{n} = x$. Então $z \neq \bar{0}$, pois $\bar{0} + \bar{n} = \bar{n}$ (verifique). Portanto $z = S(y)$, donde, por (6), $z + \bar{n} = y + \overline{n+1} = x$, donde segue que $\overline{n+1} \leq x$.

(8) Vamos mostrá-lo por indução em n. Para $n = 0$, por (Q3), $\bar{m} + \bar{0} = \bar{m}$. Suponha que Q mostre que $\bar{m} + \bar{n} = \overline{m+n}$. Então, por (Q5), $\bar{m} + \overline{n+1} = \bar{m} + S(\bar{n}) = S(\bar{m} + \bar{n}) = S(\overline{m+n}) = \overline{m+n+1}$.

(10) Suponha que $n < m$. Se $n = 0$ e $m = 1$, por (Q1), $\bar{1} = s(\bar{0}) \neq \bar{0}$. Suponha que Q mostree que $\bar{n} \neq \bar{m}$, para todo $n < m$. Então seja $n < m+1$. Se $n = 0$, novamente (Q1) resolve. Se $n = n_0 + 1$, por hipótese de indução, $Q \vdash \overline{n_0} \neq \bar{m}$, logo, por (Q2), $S(\overline{n_0}) = \bar{n} \neq S(\bar{m}) = \overline{m+1}$.

(11) Se $x = \bar{k}$, para algum $k = 0, \ldots, n$, então $\overline{n-k} + x = \bar{n}$, donde segue por (Q8) que $x \leq \bar{n}$. A recíproca demonstramos por indução em n. Se $n = 0$, se $z + x \leq \bar{0}$, por (1), segue que $x = \bar{0}$. Suponha que Q mostre $x \leq \bar{n} \iff (x = \bar{0} \vee x = \bar{1} \vee \ldots \vee x = \bar{n})$. Se $x \leq \overline{n+1}$, se $x = \bar{0}$, então vale a conclusão $(x = \bar{0} \vee x = \bar{1} \vee \ldots \vee x = \overline{n+1})$. Se $x \neq \bar{0}$, por (Q3), existe y, tal que $x = S(y)$. Então a hipótese $x \leq \overline{n+1}$ pode ser escrita como $S(y) \leq S(\bar{n})$, ou seja, existe z, tal que $z + S(y) = S(z + y) = S(\bar{n})$, donde segue que $y \leq \bar{n}$. Por hipótese de indução, temos $(y = \bar{0} \vee y = \bar{1} \vee \ldots \vee y = \bar{n})$. Daí, substituindo $x = S(y)$, temos o desejado. □

Exercício 7.4.4 A função beta de Gödel. Seja $\beta(x, y, z) = \mathrm{rm}\,(x, 1 + z(y + 1))$, o resto da divisão de x por $1 + zy$. Mostre que esta função é primitiva recursiva e que é representável em Q (e, portanto em PA), por uma fórmula Δ_1.

Exercício 7.4.5 O algoritmo de Euclides. Sejam $a, b \in \mathbb{N}$ não nulos. Para calcularmos o máximo divisor comum (mdc) de a e b, usamos o algoritmo de Euclides (que é demonstrado no livro VII dos *Elementos*.) Podemos supor que $0 < a < b$. Divida b por a, obtendo quociente q_0 e resto r_0, $0 \leq r < a$. Se o resto não for zero, divida a por r_0, obtendo quociente q_1 e resto r_1. O próximo passo consiste em dividir o divisor da conta anterior pelo seu resto. Continue o processo, obtendo quocientes $r_k = q_k\, r_{k-1} + r_{k+1}$, até que $r_n = 0$, para algum n. Se $r_{n-1} \neq 0$ e $r_n = 0$, então r_{n-1} é o máximo divisor comum de a e b. (Demonstre isto.) Conclua também que existem $c, d \in \mathbb{Z}$, tais que $c \cdot a + d \cdot b$ é o mdc de a e b.

O Teorema Chinês dos Restos, que veremos a seguir, indica a possibilidade de resolvermos um sistema de congruências, e será usado para mostrarmos que podemos codificar e decodificar (mediante a função beta de Gödel) sequências finitas de números naturais.

Teorema 7.4.1 (Teorema Chinês dos Restos) Sejam $k_1, \ldots, k_n \in \mathbb{N}$ números dois a dois primos entre si, $m_1, \ldots, m_n \in \mathbb{N}$, tais que, para cada $i = 1, \ldots, n$, m_i e k_i são primos entre si, e $a_1, \ldots, a_n \in \mathbb{N}$ quaisquer. Então existe $a \in \mathbb{N}$, tal que, para todo $i = 1, \ldots, n$, $m_i \cdot a \equiv a_i \mathrm{mod}\ k_i$.

Demonstração: Demonstraremos por indução em n. Para evitar trivialidades, podemos supor que $k_i > 1$, $i = 1, \ldots, n$.

Para $n = 1$, temos que resolver a congruência $m_1 \cdot x \equiv a_1 \text{mod } k_1$. Como m_1 e k_1 são primos entre si, existem $c, d \in \mathbb{Z}$, tais que $c \cdot m_1 + d \cdot k_1 = 1$. Somando-se um múltiplo positivo de k_1 a c, se $c < 0$, podemos supor que $c > 0$, donde segue que $c \cdot m_1 \equiv 1 \text{mod } k_1$. Daí, se $x = c \cdot a_1$, temos que $m_1 \cdot c \cdot a_1 \equiv a_1 \text{mod } k_1$. E mais, todas as soluções da congruência são da forma $x = c \cdot a_1 + t \cdot k_1$, $t \in \mathbb{N}$.

Suponha que todo sistema de $n - 1$ congruências tenha solução e consideremos o sistema $m_i \cdot x \equiv a_i \text{mod } k_i$, $i = 1, \ldots, n$, satisfazendo as hipóteses. Seja $x = c \cdot a_1 + tk_1$ uma solução da primeira congruência, com o número t a ser determinado. Substituindo nas outras congruências, temos $(m_i \cdot k_1)t \equiv a'_i \text{mod } k_i$, $i = 2, \ldots, n$, sendo que $a'_i \equiv -c \cdot a_1 \text{mod } k_1$ é um número positivo (explique como obtê-lo). Pelas hipóteses sobre os coeficientes m_i e k_i, $i = 1, \ldots, n$, o mdc entre k_i e $(m_i \cdot k_1)$ é 1, portanto, por hipótese de indução existe solução a' para este novo sistema (na variável t), o que nos dá uma solução a para o sistema original. □

O próximo teorema foi uma ideia interessante que Gödel teve para codificar e decodificar sequências finitas de números naturais, usando sua função beta e aplicando o Teorema Chinês dos Restos.

Teorema 7.4.2 Seja $\beta(x, y, z) = \mathsf{rm}\,(x, 1 + z(y + 1))$ a função beta de Gödel. Então, dados $n \geq 0$ e $a_0, \ldots, a_{n-1} \in \mathbb{N}$, existem $b, c \in \mathbb{N}$, tais que $\beta(b, 0, c) = n$ e $\beta(b, j + 1, c) = a_j$, $0 \leq j < n$.

Demonstração: Seja $N = \max\{n + 1, a_0, \ldots, a_{n-1}\}$, e $c = N!$. Então, se $0 \leq j < i \leq n$, $1 + j \cdot c$ e $1 + i \cdot c$ são primos entre si, pois se $r > 0$ divide ambos, então divide a diferença de ambos, $(i - j) \cdot N!$. Pela escolha de N, $(i - j) < N$, donde segue que r divide $N!$. Portanto r divide $(1 + j \cdot N!) - j \cdot N! = 1$. Pelo teorema chinês dos restos, seja b uma solução do sistema

$$\begin{cases} x & \equiv & n & \text{mod} & 1 + N! \\ x & \equiv & a_0 & \text{mod} & 1 + 2 \cdot N! \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ x & \equiv & a_{n-1} & \text{mod} & 1 + n \cdot N! \end{cases}$$

Com isto, demonstramos o teorema. $\square$

Nos exemplos e exercícios a seguir, $T \subseteq T(\mathbb{N})$ é uma teoria recursivamente axiomatizável e contendo Q, numa linguagem contendo uma quantidade finita de símbolos não lógicos. (Lembre-se que os símbolos, termos e fórmular da linguagem são números inteiros.)

Exemplo 7.4.5 Lembre-se de que uma dedução de ϕ a partir de T é uma sequência de fórmulas $\phi_0, \ldots, \phi_n$, tais que, ϕ_n é ϕ, e para cada $i \leq n$, ou $\phi_i \in T$, ou ϕ_i é um axioma lógico, ou existem $j, k < i$, tais que ϕ_k é a fórmulas $\phi_j \to \phi_i$ (regra do destacamento, ou *Modus Ponens*), ou existe $j < i$ e variável x, tais que ϕ_i é a fórmula $\forall x\, \phi_j$ (regra da generalização). Descreva uma função primitiva recursiva $\Delta(x)$, tal que $\Delta(x) = 1$ se $x = \langle n, \phi_0, \ldots, \phi_n \rangle$, sendo que $\phi_0, \ldots, \phi_n$ é uma dedução a partir de T, $\langle a_0, \ldots, a_k \rangle$ é definido por $\langle a_0, a_1 \rangle = (a_0 + a_1) \cdot (a_0 + a_1 + 1)/2 + a_1$ (a função primitiva recursiva que define uma bijeção de $\mathbb{N}^2$ sobre $\mathbb{N}$), $\langle a_0, \ldots, a_k \rangle = \langle a_0, \langle a_1, \ldots, a_k \rangle \rangle$, se $k > 1$, e $\Delta(x) = 0$, caso contrário.

Lema 7.4.2 Se f é uma função recursiva, então ela é representável em T por uma fórmula $\phi_f \in \Delta_1$.

Demonstração: Verificaremos este lema por indução na construção de f.

Consideremos os casos iniciais. Se $f(x) = 0$, $\phi_f(x, y)$ é $x = x \wedge y = 0$; se $f(x) = S(x)$, $\phi_f(x, y)$ é $y = S(x)$, que são fórmulas Δ_1.

Agora vejamos a regra de composição. Se $g(x_1, \ldots, x_n)$ é representável por ϕ_g e $h_j(x_1, \ldots, x_k)$ são representáveis por (fórmulas Δ_1) ϕ_{h_j}, $j = 1, \ldots, n$, então a função

$$f(x_1, \ldots, x_k) = g(h_1(x_1, \ldots, x_k), \ldots, h_n(x_1, \ldots, x_k))$$

é representável pela fórmula

$$\exists z_1, \ldots, \exists z_n [\phi_f(z_1, \ldots, z_n, y) \wedge \bigwedge_{j=1}^{n} \phi_{h_j}(x_1, \ldots, x_k, z_j)],$$

que é uma fórmula Σ_1, e também por

$$\forall z_1, \ldots, \forall z_n [\bigwedge_{j=1}^{n} \phi_{h_j}(x_1, \ldots, x_k, z_j) \to \phi_f(z_1, \ldots, z_n, y)],$$

que é uma fórmula Π_1.

Agora, a regra da recursão primitiva. Se $f(x_0,\ \bar{x})$, $\bar{x} = (x_1,\ \ldots,\ x_n)$, é definida por $f(0,\ \bar{x}) = g_0(\bar{x})$, e $f(m+1,\ \bar{x}) = g_1(m,\ \bar{x},\ f(n,\ \bar{x}))$, g_i é representada por (fórmulas Δ_1) ϕ_{g_i}, $i = 0, 1$, e a função $\beta(x_1, x_2, x_3)$ de Gödel é representada por (uma fórmula Δ_1) $\phi_\beta(x_1, x_2, x_3, y)$, então f é representada por $\exists z_1, \ldots, \exists z_5[\phi_\beta(z_1\ ,\ 0,\ z_2,\ z_3) \wedge \phi_{g_0}(\bar{x},\ z_3) \wedge \forall z_6 < x_0(\phi_\beta(z_1,\ z_6,\ z_2,\ z_4) \wedge \phi_\beta(z_1,\ S(z_6),\ z_2,\ z_5) \wedge \phi_{g_1}(z_6,\ \bar{x},\ z_4,\ z_5)) \wedge \phi_\beta(z_1,\ x_0,\ z_2,\ y)]$, e também por $\forall z_1$, $\ldots$, $\forall z_5$, $[\phi_\beta(z_1\ ,\ 0,\ z_2,\ z_3) \wedge \phi_{g_0}(\bar{x},\ z_3) \wedge \forall z_6 < x_0 \phi_\beta(z_1,\ z_6,\ z_2,\ z_4) \wedge \phi_\beta(z_1,\ S(z_6),\ z_2,\ z_5) \wedge \phi_{g_1}(z_6,\ \bar{x},\ z_4,\ z_5)) \to \phi_\beta(z_1,\ x_0,\ z_2,\ y)]$.

Finalmente, para o caso da minimização, o tratamento é análogo ao da recursão primitiva, e fica como exercício para as(os) leitoras(es). □

Exercício 7.4.6 Mostre que f é representável em T se, e só se, f é recursiva. (Use a função beta para codificar a recursão primitiva e a função Δ do exemplo 7.4.5 para buscar deduções a partir de T.)

Conclua que uma relação P será (fracamente) expressível em T se, e só se, P for recursiva (respectivamente, recursivamente enumerável).

O próximo teorema mostra que a teoria Q é expressiva o suficiente para dela deduzirmos todas as sentenças Σ_1 satisfeitas em $\mathbb{N}$.

Teorema 7.4.3 Seja ϕ uma sentença Σ_1 na linguagem da teoria Q, tal que $\mathbb{N} \models \phi$. Então $Q \vdash \phi$.

Demonstração: Basta mostrar que se $\phi(x_1, \ldots, x_n)$ for uma fórmula Δ_0 e se existirem $k_1, \ldots, k_n \in \mathbb{N}$, tais que $\mathbb{N} \models \phi(\bar{k}_1,\ \ldots,\ \bar{k}_n)$, então $Q \vdash \phi(\bar{k}_1,\ \ldots,\ \bar{k}_n)$. Seja $F(x_1,\ \ldots, x_n)$ a função característica do conjunto $\{(k_1, \ldots, k_n) \in \mathbb{N}^n\colon\ \mathbb{N} \models \phi(\bar{k}_1, \ldots, \bar{k}_n)\}$. Então F é primitiva recursiva, portanto representável em Q. □

Seja $\ulcorner\cdot\urcorner : \mathbb{N} \to \mathbb{N}$ a função definida por $\ulcorner 0 \urcorner = 3$, $\ulcorner n+1 \urcorner = 2^7 \cdot 3^{\ulcorner n \urcorner}$. (Lembre-se de que 3 é o número associado ao símbolo de constante 0, 7 é o número associado ao símbolo da função S, sucessor, e de como definimos termos em Tr_K.) Esta função calcula o número do termo $S^n(0)$ (verifique).

Exercício 7.4.7 Seja v (o número de) uma variável e $t \in Tr_K$. Definimos a função $Sb^t_v : \mathbb{N} \to \mathbb{N}$, por $Sb^t_v(n) = 0$ se $n \notin Tr_K$; $Sb^t_v(v) = t$, $Sb^t_v(c_j) = c_j$, $Sb^t_v(f_{k,l}(t_0, \ldots, t_{k-1})) = f_{k,l}(Sb^t_v(t_0), \ldots, Sb^t_v(t_{k-1}))$. Mostre que esta função é primitiva recursiva.

Exercício 7.4.8 Seja v (o número de) uma variável e $t \in Tr_K$. Definimos a função $S^t_v : \mathbb{N} \to \mathbb{N}$, por $S^t_v(n) = 0$ se $n \notin \mathsf{Fla}_K$; se $n = r_{k,l}[t_0, \ldots, t_{k-1}] \in At_K$, $S^t_v(n) = r_{k,l}[Sb^t_v(t_0), \ldots, Sb^t_v(t_{k-1})]$; se $n = \neg a \in \mathsf{Fla}_K$, $S^t_v(n) = \neg(S^t_v(a))$; se $n = a \to b \in \mathsf{Fla}_K$, $S^t_v(n) = (S^t_v(a)) \to (S^t_v(b))$; se $n = \forall a\, b \in \mathsf{Fla}_K$, e $a \neq v$, então $S^t_v(n) = \forall a\, S^t_v(b)$; se $n = \forall a\, b \in \mathsf{Fla}_K$, e $a = v$, então $S^t_v(n) = n$. Mostre que a função S^t_v é primitiva recursiva. Ela calcula a fórmula obtida de ϕ, substituindo as ocorrências livres de v pelo termo t.

Denotamos $\phi(v)$ (o número de) uma fórmula em que a variável v pode ser livre e por $\phi(t) = S^t_v(\phi)$.

7.5 Teoremas de incompletude

Agora já possuímos todas as ferramentas matemáticas necessárias para demonstrarmos os teoremas de incompletude.

Exercício 7.5.1 Mostre que a relação $\text{Ded}_T = \{\langle s, \ulcorner\phi\urcorner\rangle : T \vdash \phi$, e s é o código $\langle n, \phi_1, \ldots, \phi_n\rangle$, de uma dedução de ϕ a partir de $T\}$ é representável em T (por uma fórmula limitada $\text{Ded}(x, y)$). Portanto $\text{Dem}_T = \{\ulcorner\phi\urcorner : T \vdash \phi\}$ é fracamente expressível em T, pela fórmula $\text{Dem}_T(y)$ dada por $\exists x\, \text{Ded}_T(x, y)$.

Lema 7.5.1 Seja $B(x)$ a fórmula $\text{Dem}_T(x)$. Então vale cada uma das asserções a seguir.

1. Se $Q \subseteq T$ e $T \vdash \phi$ então $T \vdash B(\ulcorner\phi\urcorner)$.
2. Se $I\Sigma_1 \subseteq T$ e $T \vdash B(\ulcorner\phi \to \psi\urcorner) \to (B(\ulcorner\phi\urcorner) \to B(\ulcorner\psi\urcorner))$.
3. Se $I\Sigma_1 \subseteq T$ e Se ϕ é fórmula, então $T \vdash B(\ulcorner\phi\urcorner) \to B(\ulcorner B(\ulcorner\phi\urcorner)\urcorner)$.

Demonstração: Mostraremos os itens 1 e 2, deixando o item 3 como exercício (que decorre de um argumento parecido com o de 2).

1. Seja $\phi_1, \ldots, \phi_n$ uma dedução de ϕ. Então $T \vdash \text{Ded}(s, \ulcorner\phi\urcorner)$, sendo $s = \langle n, \phi_1, \ldots, \phi_n\rangle$. Portanto $T \vdash \exists x \text{Ded}(x, \ulcorner\phi\urcorner)$, ou seja $T \vdash B(\ulcorner\phi\urcorner)$.

2. Se x é o código de uma demonstração de $\phi \to \psi$ e z é o código de uma demonstração de ϕ, então $\langle \pi_1(x)+\pi_1(z)+1,\, C(x,y,\ulcorner\psi\urcorner)\rangle$ é o código de uma demonstração de ψ, sendo que $C(x,y,z)$ é a função primitiva recursiva que calcula o código $\langle \Pi_2^{\pi_1(x)}(x),\ \ldots,\ \Pi_{\pi_1(x)}^{\pi_1(x)}(x),\ \Pi_2^{\pi_1(y)}(y),\ \ldots,\ \Pi_{\pi_1(y)}^{\pi_1(y)}(y),\ z\rangle$. Precisamos de $I\Sigma_1$ aqui para podermos demonstrar que

$$T \vdash \forall x \forall z [\exists w\, ((\mathrm{DED}(x,\ulcorner\phi\to\psi\urcorner) \wedge \mathrm{DED}(z,\ulcorner\phi\urcorner)) \ \to$$

$$(\mathrm{DED}(w,\ulcorner\psi\urcorner) \wedge w = C(x,y,\ulcorner\psi\urcorner)]$$

(observe que a fórmula entre colchetes é Σ_1). $\square$

Lema 7.5.2 (Lema da diagonalização) Seja $\phi(x,\bar{y})$ uma fórmula cujas variáveis livres são x e $\bar{y} = y_1,\ldots,y_n$, e T uma teoria recursivamente axiomatizável, contendo a teoria Q. Então existe uma fórmula $\psi(\bar{y})$, cujas variáveis livres são $\bar{y}$, tal que

$$T \vdash \psi(\bar{y}) \iff \phi(\ulcorner\psi(x,\bar{y})\urcorner,\bar{y}).$$

Demonstração: Seja $\phi(x,\bar{y})$ dada, e seja $F(n)$ a função definida por $F(n) = \delta(\ulcorner\delta\urcorner,\bar{y})$, se $n = \delta(x,\bar{y})$ e x, $\bar{y}$ ocorrem livres na fórmula δ, e $F(n) = 0$, se n não é desta forma. Então F é primitiva recursiva (verifique), e representável por uma fórmula Δ_1, $\alpha(x,v)$. Seja $\chi(x,\bar{y})$ a fórmula $\exists v(\alpha(x,\,v)\wedge\phi(v,\,\bar{y}))$ e $\psi = F(\chi) = \chi(\ulcorner\chi(x,\bar{y})\urcorner,\bar{y})$.

Temos que a partir de Q (contida em T) podemos deduzir as equivalências

$$\psi \iff \exists v\,(\alpha(\ulcorner\chi(x,\bar{y})\urcorner,v)\wedge\phi(v,\bar{y})) \iff$$

$$\iff \exists v\,(v = F(\ulcorner\chi(x,\bar{y})\urcorner \wedge \phi(v,\bar{y})) \iff$$

$$\iff \exists v(v = \ulcorner\psi\urcorner \wedge \psi(x,\bar{y})) \iff \phi(\ulcorner\psi\urcorner,\bar{y}).$$

$\square$

Exercício 7.5.2 Versão do lema da diagonalização para $\mathbb{N}$. Seja $\phi(x,\bar{y})$ uma fórmula cujas variáveis livres são x e $\bar{y} = y_1,\ldots,y_n$, e seja $F(n)$ a função definida por $F(n) = \delta(\ulcorner\delta\urcorner,\bar{y})$, se $n = \delta(x,\bar{y})$ e x, $\bar{y}$ ocorrem livres na fórmula δ, e $F(n) = 0$, se n não é desta forma. Seja $\alpha(x,v)$ uma fórmula

Δ_1 definindo o gráfico de F. Seja $\chi(x, \bar{y})$ a fórmula $\exists v(\alpha(x, v) \wedge \phi(v, \bar{y}))$ e $\psi = F(\chi) = \chi(\ulcorner \chi(x, \bar{y}) \urcorner, \bar{y})$. Mostre que

$$\mathbb{N} \models \forall \bar{y} \, [\psi(\bar{y}) \iff \phi(\ulcorner \psi(x, \bar{y}) \urcorner, \bar{y})].$$

Teorema 7.5.1 (Primeiro Teorema de Incompletude de Gödel) Se $T \supseteq Q$ é uma teoria consistente e recursivamente axiomatizável, tal que de T não se deduz nenhuma sentença Σ_1 falsa em $\mathbb{N}$, então existe ψ tal que $T \not\vdash \psi$ e $T \not\vdash \neg\psi$.

Demonstração: Seja ψ uma sentença dada por diagonalização da fórmula $\neg \mathrm{D{\scriptsize EM}}_T(x)$, ou seja, $T \vdash \psi \iff \neg \mathrm{D{\scriptsize EM}}_T(\ulcorner \psi \urcorner)$.

Se $T \vdash \psi$, como a fórmula $\mathrm{D{\scriptsize EM}}_T(\ulcorner \psi \urcorner)$ é Σ_1, codificando a dedução de ψ, obtemos $\mathbb{N} \models \mathrm{D{\scriptsize EM}}_T(\ulcorner \psi \urcorner)$, donde segue que $T \vdash \mathrm{D{\scriptsize EM}}_T(\ulcorner \psi \urcorner)$. Por outro lado, de $T \vdash \psi \iff \neg \mathrm{D{\scriptsize EM}}_T(\ulcorner \psi \urcorner)$, obtemos que $T \vdash \neg \mathrm{D{\scriptsize EM}}_T(\ulcorner \psi \urcorner)$, ou seja, T é inconsistente.

Se $T \vdash \neg\psi$, de $T \vdash \psi \iff \neg \mathrm{D{\scriptsize EM}}_T(\ulcorner \psi \urcorner)$, $T \vdash \mathrm{D{\scriptsize EM}}_T(\ulcorner \psi \urcorner)$, que é uma sentença Σ_1. Da hipótese de que a partir de T não se deduzem sentenças Σ_1 falsas em $\mathbb{N}$, temos que $\mathbb{N} \models \mathrm{D{\scriptsize EM}}_T(\ulcorner \psi \urcorner)$. Portanto existe um número $a \in \mathbb{N}$ que codifica uma dedução de ψ a partir de T, donde segue que $T \vdash \neg\psi$, ou seja T é inconsistente.

Portanto, sendo T consistente, $T \not\vdash \psi$ e $T \not\vdash \neg\psi$. $\square$

A hipótese de que a partir de T não se deduz nenhuma sentença Σ_1 falsa em $\mathbb{N}$ é muito forte, e pode ser eliminada, como veremos a seguir.

Teorema 7.5.2 (Teorema de Incompletude de Gödel e Rosser) Se $T \supseteq Q$ é uma teoria consistente e recursivamente axiomatizável, então existe ψ tal que $T \not\vdash \psi$ e $T \not\vdash \neg\psi$.

Demonstração: Seja $\mathsf{Rf}(x, y)$ a fórmula Δ_1 definindo a relação recursiva "x é o código de uma dedução da negação da fórmula y". Seja $\delta(y)$ a fórmula $\exists x \, \mathsf{Rf}_T(x, y) \wedge \forall z < x \neg \mathrm{D{\scriptsize ED}}_T(z, y)$, que é uma fórmula Σ_1. Seja ψ obtida por diagonalização de $\delta(y)$.

Suponha que $T \vdash \psi$. Então $T \vdash \mathrm{D{\scriptsize ED}}_T(\bar{a}, \ulcorner \psi \urcorner)$, para algum $a \in \mathbb{N}$. Sendo T consistente, $T \not\vdash \neg\psi$, donde segue que, para todo $b \in \mathbb{N}$, $\mathbb{N} \models \neg \mathsf{Rf}_T(\bar{b}, \ulcorner \psi \urcorner)$ e, como a fórmula é Σ_1, $T \vdash \neg \mathsf{Rf}_T(\bar{b}, \ulcorner \psi \urcorner)$, para todo $b \in \mathbb{N}$.

Por outro lado, da diagonalização, segue que $T \vdash \delta(\ulcorner\psi\urcorner)$, ou seja, $T \vdash \exists x\, \mathsf{Rf}_T(x, \ulcorner\psi\urcorner) \wedge \forall z < x \neg \textsc{Ded}_T(z, \ulcorner\psi\urcorner)$. Seja $b \in \mathbb{N}$, tal que $b > a$. Então $T \vdash \forall x\, \mathsf{Rf}_T(x, \ulcorner\psi\urcorner) \to (\bar{b} < x)$, donde segue que $T \vdash \forall z < \bar{b} \neg \textsc{Ded}_T(z, \ulcorner\psi\urcorner)$, ou seja, $T \vdash \neg \textsc{Ded}_T(\bar{a}, \ulcorner\psi\urcorner)$, o que implica que T é inconsistente.

Agora suponha que $T \vdash \neg\psi$ e seja $a \in \mathbb{N}$, tal que $\mathbb{N} \models \mathsf{Rf}_t(a, \ulcorner\psi\urcorner)$. Da diagonalização, $T \vdash \forall x (\mathsf{Rf}_T(x, \ulcorner\psi\urcorner) \to \exists z < x\, \textsc{Ded}_T(z, \ulcorner\psi\urcorner))$, donde segue que $T \vdash (\mathsf{Rf}_T(\bar{a}, \ulcorner\psi\urcorner) \to \exists z < \bar{a}\, \textsc{Ded}_T(z, \ulcorner\psi\urcorner))$. Como $Q \subset T$, temos que existe $b \in \mathbb{N}$, $b < a$, tal que $T \vdash \textsc{Ded}_T(\bar{b}, \ulcorner\psi\urcorner)$. Daí, segue que b é o código de uma demonstração de ψ a partir de T, isto é, $T \vdash \psi$, o que implica que T é inconsistente.

Portanto $T \not\vdash \psi$ e $T \not\vdash \neg\psi$. $\square$

Exercício 7.5.3 Seja T recursivamente axiomatizável e contendo Q.

(1) Mostre que os conjuntos $P = \{\sigma : T \vdash \sigma\}$ e $B = \{\sigma : T \vdash \neg\sigma\}$ são recursivamente enumeráveis, mas não são recursivos.

Seja $C \subset \mathsf{Fla}_K$, tal que $A \subseteq C$ e $C \cap B = \varnothing$. Seja $C' = \{\overbrace{\neg \ldots \neg}^{j}\sigma$: σ não é da forma $\neg\theta$, j é ímpar e, ou $\neg\sigma \in C$, ou $\sigma \notin C\} \cup \{\overbrace{\neg \ldots \neg}^{k}\sigma$: σ não é da forma $\neg\theta$, k é par e, ou $\sigma \in C$, ou $\neg\sigma \notin C\}$.

(2) Mostre que se C fosse recursivo, C' também seria recursivo.

(3) Mostre que se $D \subseteq \mathsf{Fla}_K$ é recursivo, $A \subset D$ e, para toda $\sigma \in \mathsf{Fla}_K$, se $\sigma \in D$ então $\neg\sigma \notin D$, então existe $\sigma \in \mathsf{Fla}_K$, tal que $\sigma, \neg\sigma \notin D$.

(4) Mostre que, para todo $\sigma \in \mathsf{Fla}_K$, se $\sigma \in C'$ então $\neg\sigma \notin C'$ e, ou $\sigma \in C'$, ou $\neg\sigma \notin C'$. Conclua que C não pode ser recursivo.

Dois conjuntos A e B como no exercício são ditos recursivamente inseparáveis.

Teorema 7.5.3 O Teorema de Löb. Sejam T teoria recursivamente axiomatizável, contendo $I\Sigma_1$, e $B(x)$ a fórmula $\textsc{Dem}_T(x)$. Se $T \vdash B(\ulcorner\phi\urcorner) \to \phi$ então $T \vdash \phi$

Demonstração: Seja ψ uma sentença, tal que $T \vdash \psi \iff (B(\ulcorner\psi\urcorner) \to \phi)$, dada pelo lema da diagonalização para a fórmula $B(x) \to \phi$.

Como $T \vdash \psi \to (B(\ulcorner\psi\urcorner) \to \phi)$, então $T \vdash B(\ulcorner\psi \to (B(\ulcorner\psi\urcorner) \to \phi)\urcorner)$, donde segue que $T \vdash B(\ulcorner\psi\urcorner) \to B(\ulcorner(B(\ulcorner\psi\urcorner) \to \phi)\urcorner)$, e portanto $T \vdash B(\ulcorner\psi\urcorner) \to (B(\ulcorner B(\ulcorner\psi\urcorner)\urcorner) \to B(\ulcorner\phi\urcorner))$. Como $T \vdash B(\ulcorner\psi\urcorner) \to B(\ulcorner B(\ulcorner\psi\urcorner)\urcorner)$, temos que $T \vdash B(\ulcorner\psi\urcorner) \to B(\ulcorner\phi\urcorner)$. Por hipótese, $T \vdash B(\ulcorner\phi\urcorner) \to \phi$, donde segue que $T \vdash B(\ulcorner\psi\urcorner) \to \phi$. Como $T \vdash \psi \iff (B(\ulcorner\psi\urcorner) \to \phi)$, temos que $T \vdash \psi$. Portanto $T \vdash B(\ulcorner\psi\urcorner)$, donde segue que $T \vdash \phi$. $\square$

Um modo de expressar a consistência de T (axiomatizável) é a sentença $\textsc{Cons}_T$ dada por $\neg B(\ulcorner 0 = S(0)\urcorner)$.

Teorema 7.5.4 (Segundo Teorema de Incompletude de Gödel) Se $T \supseteq I\Sigma_1$ é uma teoria consistente e recursivamente axiomatizável então $T \nvdash \textsc{Cons}_T$.

Demonstração: Se $T \vdash \textsc{Cons}_T$, ou seja, $T \vdash \neg B(\ulcorner 0 = S(0)\urcorner)$, então $T \vdash B(\ulcorner 0 = S(0)\urcorner) \to 0 = S(0)$. Portanto, pelo Teorema de Löb, $T \vdash 0 = S(0)$. Como $T \vdash 0 \neq S(0)$, T é inconsistente. $\square$

Dizemos que uma teoria T_0, numa linguagem contendo símbolos em $K_0 \subset \mathbb{N}$, intepreta uma teoria T_1, numa linguagem contendo símbolos em $K_1 \subset \mathbb{N}$, se existe uma K_0 fórmula $\chi(x)$ e uma correspondência $\phi \mapsto \Phi_p hi$, de fórmulas da linguagem de T_1 para fórmulas na linguagem de T_0, tal que ϕ e Φ_ϕ tenham as mesmas variáveis livres, $\Phi_{\neg\phi}$ é $\neg\Phi_\phi$, $\Phi_{\phi\to\psi}$ é $\phi_\phi \to \Phi_\psi$ e $\Phi_{\exists x\,\phi}$ é $\exists x\,\chi(x) \wedge \Phi_\phi$, tal que se $T_1 \vdash \phi$, então $T_0 \vdash \Phi_\phi$.

Exercício 7.5.4 Redemonstre os dois teoremas de incompletude, com a hipótese de que T seja recursivamente axiomatizável e que T apenas interpreta Q ou $I\Sigma_1$.

7.6 Indefinibilidade da Verdade

O próximo teorema, devido a Alfred Tarski, mostra que a noção de verdade para a linguagem-objeto da aritmética não pode ser definida por uma fórmula dessa linguagem. No entanto, se impusermos restrições na quantidade de alternância de quantificadores nas fórmulas, então podemos definir (parcialmente) a noção de verdade.

Teorema 7.6.1 (Teorema da Indefinibilidade da Verdade de Tarski) O conjunto $V(\mathbb{N}) = \{\ulcorner\phi\urcorner : \mathbb{N} \models \phi\}$ não é definível na linguagem de T, ou seja, não existe nehuma fórmula $\Theta(x)$ tal que $V(\mathbb{N}) = \{n : \mathbb{N} \models \Theta(n)\}$.

Demonstração: Suponha que exista tal $\Theta(x)$ e seja ψ a sentença dada pelo lema da diagonalização (para $\mathbb{N}$) para a fórmula $\neg\Theta(x)$, ou seja, $\mathbb{N} \models \psi \iff \neg\Theta(\ulcorner\psi\urcorner)$.

Se $\mathbb{N} \models \psi$, então $N \models \neg\Theta(\ulcorner\psi\urcorner)$, donde segue que $\mathbb{N} \not\models \psi$, contradição.

Se $\mathbb{N} \models \neg\psi$, então $N \models \Theta(\ulcorner\psi\urcorner)$, donde segue que $\mathbb{N} \models \psi$, outra contradição.

Portanto, não pode existir tal fórmula. $\square$

Relembramos que uma fórmula φ é dita limitada se todas as quantificações que aparecem nela são da forma $\forall x(x \leq t \to \theta)$ ou $\exists x(x \leq t \wedge \theta)$, sendo que t é um termo em que a variável x não ocorre. O conjunto das fórmulas limitadas é denotado por Δ_0 (e, às vezes por Π_0 ou Σ_0). Definimos os conjuntos de fórmulas Σ_n, Π_n e Δ_n, por $\phi \in \Sigma_{n+1}$ (respectivamente, Π_{n+1}) se existir uma fórmula $\psi \in \Pi_n$ (respectivamente, Σ_n), tal que ϕ é lógicamente equivalente a $\exists x_1 \ldots \exists x_n\, \psi$ ((respectivamente, $\forall x_1 \ldots \forall x_n\, \psi$). Definimos $\Delta_n = \Sigma_n \cap \Pi_n$.

Para terminarmos este texto, vamos mostrar que existem definições parciais de verdade para a aritmética, ou seja, para cada $n \in \mathbb{N}$, existe uma fórmula $\textsc{Sat}_{\Sigma_n}(x)$, tal que, se $\bar{m}$ for o termo correspondente ao numeral $m \in \mathbb{N}$, $N \models \textsc{Sat}_{\Sigma_n}(\bar{m})$ se, e somente se, existir uma sentença $\phi \in \Sigma_n$, tal que $\bar{m} = \ulcorner\phi\urcorner$ e $\mathbb{N} \models \phi$.

Comecemos analisando o caso das fórmulas limitadas (ou Σ_0, ou também, Δ_0).

Exercício 7.6.1 Dada uma fórmula $\phi \in \Delta_0$, cujas variáveis livres estejam dentre as variáveis $x_0, \ldots, x_N$, para algum $N \in \mathbb{N}$, e dada uma atribuição de valores $s : x_i \in \{x_0, \ldots, x_N\} \mapsto a_i \in \mathbb{N}$, mostre que é possível verificar, efetuando uma quantidade finita de operações, se é verdade que $\mathbb{N} \models \phi[s]$. Descreva informalmente um algoritmo que decida este problema.

Deste modo, a função $\mathsf{Sat}_{\Delta_0}(x, y) : \mathbb{N}^2 \to \mathbb{N}$ que vale 1 se x for substituída pelo número de Gödel de uma fórmula $\phi \in \Delta_0$ e y pelo código de uma sequência finita s de números naturais, de comprimento pelo menos $N = \max\{j \in \mathbb{N} : VL(\phi) \subseteq \{x_0, \ldots, x_j\}\}$, e $\mathbb{N} \models \phi[s]$. Portanto, existe uma fórmula $Sat_{\Delta_0}(x_0, x_1) \in \Delta_1$ [13], representando a relação $\mathsf{Sat}_{\Delta_0}(x, y) = 1$. Tendo esta fórmula em mãos, podemos escrever uma fórmula $\textsc{Sat}_{\Delta_0}(x_0) \in \Delta_1$, que define o conjunto dos números de Gödel das sentenças Δ_0 que são verdadeiras (ou satisfeitas) em $\mathbb{N}$.

Exercício 7.6.2 Descreva explicitamente a fórmula $\textsc{Sat}_{\Delta_0}$.

Exercício 7.6.3 Mostre que, para uma sentença $\phi \in \Delta_0$, $\mathbb{N} \models \phi$ se, e somente se, $\mathbb{N} \models \textsc{Sat}_{\Delta_0}(\ulcorner\phi\urcorner)$.

Dizemos que a fórmula $\textsc{Sat}_{\Delta_0}$ é uma *definição parcial de verdade* para sentenças Δ_0.

Recursivamente, definimos $Sat_{\Sigma_0} = Sat_{\Pi_0} = Sat_{\Delta_0}$ e, assumindo que já tenhamos definido as fórmulas Sat_{Σ_n} e Sat_{Π_n}, então definimos as fórmulas

> $Sat_{\Sigma_{n+1}}(x_0, x_1)$: "*$x_0$ é o número de Gödel de uma fórmula do tipo $\exists x\phi$, com $\phi \in \Pi_n$, x_1 é atribuição de valores às variáveis livres de ϕ e existe uma atribuição de valores s' que coincide com x_1 nas variáveis livres de ϕ e $Sat_{\Pi_n}(x_0, s)$*",
>
> $Sat_{\Pi_{n+1}}(x_0, x_1)$: "*$x_0$ é o número de Gödel de uma fórmula do tipo $\forall x\phi$, com $\phi \in \Pi_n$, x_1 é atribuição de valores às variáveis livres de ϕ e para toda atribuição de valores s' que coincide com x_1 nas variáveis livres de ϕ e $Sat_{\Pi_n}(x_0, s)$*".

Estas fórmulas Sat_{Σ_n} e Sat_{Π_n} são chamadas de *definição parcial de verdade* para fórmulas Σ_n e Π_n, respectivamente.

Exercício 7.6.4 Mostre que, para uma fórmula $\phi \in \Sigma_n$ e uma atribuição s, $\mathbb{N} \models \phi[s]$ se, e somente se, $\mathbb{N} \models Sat_{\Delta_0}(\ulcorner\phi\urcorner, \operatorname{cod}(s))$, sendo que $\operatorname{cod}(s)$ é o numeral que representa o código da sequência s.

Exercício 7.6.5 Mostre que, para uma fórmula $\phi \in \Pi_n$ e uma atribuição s, $\mathbb{N} \models \phi[s]$ se, e somente se, $\mathbb{N} \models Sat_{\Delta_0}(\ulcorner\phi\urcorner, \operatorname{cod}(s))$, sendo que $\operatorname{cod}(s)$ é o numeral que representa o código da sequência s.

---

[13] É necessário assumir o fragmento $I\Sigma_1$ para demonstrar que esta fórmula é realmente Δ_1.