Chapter 9
Gödelisation of Peano Arithmetic

The key ingredient for Gödel's Incompleteness Theorems is the so-called Gödelisation process which allows us to code terms, formulae and even proofs within PA. In order to achieve this, we introduce Gödel's β-function with the help of which one can encode any FINITE sequence of natural numbers by single natural number.

Natural Numbers in Peano Arithmetic

As we have already seen in Chapter 7, every standard natural number corresponds to a unique $\mathscr{L}_{\mathsf{PA}}$-term. More precisely, every element $\sigma\mathbf{0}$ of $\mathbb{N}$ corresponds to a term $\underline{\sigma}0$. In order to simplify notations, we will from now on use variables such as $n, m, \ldots$ to denote elements of $\mathbb{N}$ and $\underline{n}, \underline{m}, \ldots$ their counterpart in the formal language $\mathcal{L}_{\mathsf{PA}}$, *i.e.*, if n stands for $\sigma\mathbf{0}$, then $\underline{n}$ denotes $\underline{\sigma}0$. Then FACT 7.1 yields

$$n \equiv m \quad \Longleftrightarrow \quad \mathsf{PA} \vdash \underline{n} = \underline{m}.$$

Moreover, by definition of $\underline{n}$ for $n \in \mathbb{N}$ we have

$$\underline{\mathbf{0}} \equiv 0$$
$$\underline{\mathbf{s}n} \equiv \mathsf{s}\underline{n}.$$

PROPOSITION 9.1. *Any two natural numbers $n, m \in \mathbb{N}$ satisfy the properties*

N_0: $\mathsf{PA} \vdash \mathsf{s}\underline{n} = \underline{\mathbf{s}n}$

N_1: $\mathsf{PA} \vdash \underline{m} + \underline{n} = \underline{m +^{\mathbb{N}} n}$

N_2: $\mathsf{PA} \vdash \underline{m} \cdot \underline{n} = \underline{m \cdot^{\mathbb{N}} n}$

N_3: *If $m \equiv n$ then* $\mathsf{PA} \vdash \underline{m} = \underline{n}$ *and if $m \not\equiv n$ then* $\mathsf{PA} \vdash \underline{m} \neq \underline{n}$

N_4: *If $m < n$ then* $\mathsf{PA} \vdash \underline{m} < \underline{n}$ *and if $m \not< n$ then* $\mathsf{PA} \vdash \underline{m} \not< \underline{n}$

N_5: $\mathsf{PA} \vdash \forall x\big(x < \underline{n} \leftrightarrow \bigvee_{k=\mathbf{0}}^{n-1} x = \underline{k}\big)$

Before we give a proof of PROPOSITION 9.1, let us recall the induction induction principle that we have introduced in Chapter 0: *If a statement A holds for* $\mathbf{0}$ *and if whenever A holds for a natural number n in $\mathbb{N}$ then it also holds for $n+1$, then the statement A holds for all natural numbers n in $\mathbb{N}$.* Note that this induction principle is more general than what we obtain from the induction axiom PA_6 in the standard model $\mathbb{N}$. The reason is, that PA_6 is restricted to properties which can be described by an $\mathscr{L}_{\mathsf{PA}}$-formula, whereas the induction principle applies to any statement about standard natural numbers. In order to distinguish between the induction principle for standard natural numbers and induction within PA using PA_6, we shall call the former *metainduction*.

Proof of Proposition 9.1. N_0 follows directly from the definition of $\underline{n}$ for natural numbers $n \in \mathbb{N}$.

We prove N_1 by metainduction on n. The case $n \equiv \mathbf{0}$ is obviously true, since $\underline{\mathbf{0}}$ is 0. For the induction step, let us assume $\mathsf{PA} \vdash \underline{m} + \underline{n} = \underline{m +^{\mathbb{N}} n}$. Using N_0 and PA_3 both within PA and in $\mathbb{N}$ we obtain

$$\mathsf{PA} \vdash \underline{m} + \underline{\mathbf{s}n} = \underline{m} + \mathsf{s}\underline{n} = \mathsf{s}(\underline{m} + \underline{n}) = \mathsf{s}(\underline{m +^{\mathbb{N}} n}) = \underline{\mathbf{s}(m +^{\mathbb{N}} n)} = \underline{m +^{\mathbb{N}} \mathbf{s}n}.$$

The proof of N_2 is similar and is left as an exercise to the reader.

The first part of N_3 follows from FACT 7.1 and the second part is a consequence of N_4, since whenever $m \not\equiv n$, then either $m < n$ or $n < m$.

So, let us turn to N_4. If $m < n$, then there is $k \in \mathbb{N}$ such that $m +^{\mathbb{N}} k \equiv n$ and $k \not\equiv \mathbf{0}$. By N_3 and N_1 we get $\mathsf{PA} \vdash \underline{m} + \underline{k} = \underline{m +^{\mathbb{N}} k} = \underline{n}$. It remains to show that $\mathsf{PA} \vdash \underline{k} \neq 0$. Since $k \not\equiv \mathbf{0}$, it is of the form $\mathbf{s}k'$ for some $k' \in \mathbb{N}$. Thus by N_0 and PA_0, $\mathsf{PA} \vdash \underline{k} = \underline{\mathbf{s}k'} = \mathsf{s}\underline{k'} \neq 0$. The second statement of N_4 follows from the first one and N_3 by observing that if $m \not< m$, then either $m \equiv n$ or $n \not< m$.

In order to prove N_5, we proceed by metainduction on n. The case $n \equiv \mathbf{0}$ is trivially satisfied. Now, assume that N_5 holds for some n and let $x < \underline{\mathbf{s}n} = \mathsf{s}\underline{n}$. Then, by LEMMA 8.6, we get $x \leq \underline{n}$, *i.e.*, either $x < \underline{n}$ or $x = \underline{n}$. Since the first case is equivalent to $\bigvee_{k=\mathbf{0}}^{n-1} x = \underline{k}$ by assumption, we obtain $\bigvee_{k=\mathbf{0}}^{n} x = \underline{k}$ as desired. The converse is a consequence of N_4. ⊣

On the one hand, by the SOUNDNESS THEOREM **??** we know that all what is provable within PA holds in every model of PA, in particular in the standard model $\mathbb{N}$. On the other hand, not every statement which is true in $\mathbb{N}$ must be provable within PA. With this respect, PROPOSITION 9.1 gives us a few statements which are true in the standard model $\mathbb{N}$ and which are provable within PA. In order to obtain more such statements, we shall introduce the notion of $\mathbb{N}$-*conformity*, but for this, we have to give a few preliminary notions.

We call an $\mathscr{L}_{\mathsf{PA}}$-formula φ a **strict $\exists$-formula**, if it is built up from atomic formulae and negated atomic formulae using $\wedge, \vee$, existential quantification "$\exists\nu$" and bounded universal quantification, *i.e.*, "$\forall \nu < \tau$" for some term τ. Furthermore, φ is said to be an **$\exists$-formula**, if there is a strict $\exists$-formula ψ such that $\varphi \Leftrightarrow_{\mathsf{PA}} \psi$. By exchanging the role of universal and existential quantification in the above definition,

we can analogously define (**strict**) $\forall$**-formulae**. Furthermore, if a formula is both an $\exists$- and a $\forall$-formula, then we call it a Δ**-formula**. In particular, every formula which contains only bounded quantifiers is a Δ-formula.

Example 9.1. The formulae "$x < y$" and "$x \mid y$" are Δ-formulae:

$$x \leq y \;\Leftrightarrow_{\mathsf{PA}}\; \exists r < y\,(x + r = y)$$
$$x \mid y \;\Leftrightarrow_{\mathsf{PA}}\; \exists r < \mathtt{s}y\,(rx = y)$$

PROPOSITION 9.2. *Let $\varphi(x_1, \ldots, x_n)$ be a formula whose free variables are among $x_1, \ldots, x_n$, and let $a_1, \ldots, a_n \in \mathbb{N}$.*

(a) *If φ is an $\exists$-formula and $\mathbb{N} \vDash \varphi(a_1, \ldots, a_n)$, then $\mathsf{PA} \vdash \varphi(\underline{a_1}, \ldots, \underline{a_n})$.*

(b) *If φ is a $\forall$-formula and $\mathbb{N} \vDash \neg\varphi(a_1, \ldots, a_n)$, then $\mathsf{PA} \vdash \neg\varphi(\underline{a_1}, \ldots, \underline{a_n})$.*

Proof. Observe first that (b) follows from (a), since the negation of a $\forall$-formula is an $\exists$-formula. Furthermore, note that it is enough to prove (a) for strict $\exists$-formulae. We proceed by induction on the complexity of φ.

- If φ is an atomic formula, it is of the form $\tau_0(x_1, \ldots, x_n) = \tau_1(x_1, \ldots, x_n)$ for some terms τ_0, τ_1 whose variables are among $x_1, \ldots, x_n$. We show by induction on the construction of terms that for every term $\tau(x_1, \ldots, x_n)$ and for all standard natural numbers $a_1, \ldots, a_n \in \mathbb{N}$, we have

$$\mathsf{PA} \vdash \underline{\tau^{\mathbb{N}}(a_1, \ldots, a_n)} = \tau(\underline{a_1}, \ldots, \underline{a_n}). \qquad (*)$$

  The statement is clear for terms τ of the form $\tau \equiv \nu$ (for a variable ν) or $\tau \equiv \mathtt{0}$. If τ is of the form $\mathtt{s}\tau'$ for some term τ', then by induction hypothesis $\mathsf{PA} \vdash \underline{a} = \tau'(\underline{a_1}, \ldots, \underline{a_n})$, where $a = \tau'^{\mathbb{N}}(a_1, \ldots, a_n) \in \mathbb{N}$. Thus $\tau^{\mathbb{N}}(a_1, \ldots, a_n)$ is $\mathtt{s}a$. Then by N_0 we have $\vdash_{\mathsf{PA}} \underline{\mathtt{s}a} = \mathtt{s}\underline{a} = \mathtt{s}\tau'(\underline{a_1}, \ldots, \underline{a_n}) = \tau(\underline{a_1}, \ldots, \underline{a_n})$ as desired. The proof for terms of the form $\tau_0 + \tau_1$ or $\tau_0 \cdot \tau_1$ are similar using N_1 and N_2 respectively. This shows $(*)$.
  Now assume $\mathbb{N} \vDash \tau_0(a_1, \ldots, a_n) = \tau_1(a_1, \ldots, a_n)$ and put $a \equiv \tau_0(a_1, \ldots, a_n)$ and $b \equiv \tau_1(a_1, \ldots, a_n)$. Then by $(*)$ and N_3 we get

$$\mathsf{PA} \vdash \tau_0(\underline{a_1}, \ldots, \underline{a_n}) = \underline{a} = \underline{b} = \tau_1(\underline{a_1}, \ldots, \underline{a_n}).$$

- Suppose that $\varphi(x_1, \ldots, x_n) \equiv \varphi_0(x_1, \ldots, x_n) \wedge \varphi_1(x_1, \ldots, x_n)$ and $\mathbb{N} \vDash \varphi(a_1, \ldots, a_n)$. Then $\mathbb{N} \vDash \varphi_0(a_1, \ldots, a_n)$ and $\mathbb{N} \vDash \varphi_1(a_1, \ldots, a_n)$. By induction hypothesis, $\mathsf{PA} \vdash \varphi_0(\underline{a_1}, \ldots \underline{a_n})$ and $\mathsf{PA} \vdash \varphi_1(\underline{a_1}, \ldots, \underline{a_n})$. Using $(\mathsf{I}\wedge)$ this shows that $\mathsf{PA} \vdash \varphi(\underline{a_1}, \ldots, \underline{a_n})$. The disjunctive case is similar.
- Let now $\varphi(x_1, \ldots, x_n) \equiv \forall y < \tau(x_1, \ldots, x_n)\psi(x_1, \ldots, x_n, y)$ and suppose that $\mathbb{N} \vDash \varphi(a_1, \ldots, a_n)$. Let $a \equiv \tau^{\mathbb{N}}(a_1, \ldots, a_n)$. Thus for every $b < a$, $\mathbb{N} \vDash \psi(a_1, \ldots, a_n, b)$. Thus by induction hypothesis, for every $b < a$ we have $\mathsf{PA} \vdash \psi(\underline{a_1}, \ldots, \underline{a_n}, \underline{b})$ and by $(*)$, $\mathsf{PA} \vdash \underline{a} = \tau(\underline{a_1}, \ldots, \underline{a_n})$. Now using N_5 we obtain

$$\mathsf{PA} \vdash \varphi(\underline{a_1}, \ldots, \underline{a_n}) \leftrightarrow \forall y \Big(\bigvee_{b=0}^{a-1} y = \underline{b} \rightarrow \psi(\underline{a_1}, \ldots, \underline{a_n}, y) \Big).$$

The right-hand side can clearly be derived in PA.

- Finally, let $\varphi(x_1, \ldots, x_n) \equiv \exists y \psi(x_1, \ldots, x_n, y)$. Then $\mathbb{N} \vDash \varphi(a_1, \ldots, a_n)$ implies that there exists $b \in \mathbb{N}$ such that $\mathbb{N} \vDash \psi(a_1, \ldots, a_n, b)$. Inductively, we get $\mathbb{N} \vDash \psi(\underline{a_1}, \ldots, \underline{a_n}, \underline{b})$ which proves the claim.

⊣

Note that any constants, relations, and functions that one can define in PA in the sense of Chapter 6 can be interpreted in the standard model $\mathbb{N}$.

A relation $R(x_1, \ldots, x_n)$ defined by

$$R(x_1, \ldots, x_n) :\Longleftrightarrow \psi_R(x_1, \ldots, x_n)$$

is said to be **$\mathbb{N}$-conform**, if for all $a_1, \ldots, a_n \in \mathbb{N}$ the following two properties are satisfied:

(a) If $\mathbb{N} \vDash \psi_R(a_1, \ldots, a_n)$, then $\mathsf{PA} \vdash \psi_R(\underline{a_1}, \ldots, \underline{a_n})$

(b) If $\mathbb{N} \vDash \neg\psi_R(a_1, \ldots, a_n)$, then $\mathsf{PA} \vdash \neg\psi_R(\underline{a_1}, \ldots, \underline{a_n})$.

For the sake of simplicity, the formula ψ_R is also called $\mathbb{N}$-conform.

Now, let f be a function symbol whose defining formula is $\psi_f(x_1, \cdots, x_n, y)$, *i.e.*, $\mathsf{PA} \vdash \forall x_1 \ldots \forall x_n \exists! y \, \psi_f(x_1, \ldots, x_n, y)$ and

$$f x_0 \cdots x_n = y :\Longleftrightarrow \psi_f(x_1, \ldots, x_n).$$

Then we say that f is **$\mathbb{N}$-conform**, if its defining formula ψ_f is $\mathbb{N}$-conform. Let $f^{\mathbb{N}}$ be the interpretation of f in $\mathbb{N}$. If f is $\mathbb{N}$-conform, then for all $a_1, \ldots, a_n \in \mathbb{N}$

$$\mathsf{PA} \vdash \psi_f(\underline{a_1}, \ldots, \underline{a_n}, \underline{f^{\mathbb{N}}(a_1, \ldots, a_n)}).$$

and hence

$$\mathsf{PA} \vdash f(\underline{a_1}, \ldots, \underline{a_n}) = \underline{f^{\mathbb{N}}(a_1, \ldots, a_n)}).$$

To see this, suppose that f is $\mathbb{N}$-conform. For the sake of simplicity, suppose that $n \equiv 1$ and let $a \in \mathbb{N}$. Then $\mathbb{N} \vDash \psi_f(a, f^{\mathbb{N}}(a))$, hence by $\mathbb{N}$-conformity we get $\mathsf{PA} \vdash \psi_f(\underline{a}, \underline{f^{\mathbb{N}}(a)})$. On the other hand, we have $\mathsf{PA} \vdash \psi_f(\underline{a}, f(\underline{a}))$ and hence by functionality of ψ_f we get $\mathsf{PA} \vdash f(\underline{a}) = \underline{f^{\mathbb{N}}(a)}$.

COROLLARY 9.3.

(a) *Every relation which is defined by a Δ-formula is $\mathbb{N}$-conform.*

(b) *Every function which is defined by an $\exists$-formula is $\mathbb{N}$-conform.*

Proof. Condition (a) follows directly from PROPOSITION 9.2. For (b) it suffices to prove that every function whose defining formula is an $\exists$-formula is already a Δ-formula. Suppose that f is defined by the $\exists$-formula ψ_f, *i.e.*,

$$f(x_1, \ldots, x_n) = y :\Longleftrightarrow \psi_f(x_1, \ldots, x_n, y).$$

Now note that by functionality of ψ_f, we have

$$\psi_f(x_1, \ldots, x_n, y) \Leftrightarrow_{\mathsf{PA}} \forall z(\psi_f(x_1, \ldots, x_n, z) \to z = y).$$

Moreover, by TAUTOLOGY (K),

$$\forall z(\psi_f(x_1, \ldots, x_n, z) \to z = y) \Leftrightarrow \forall z(\neg\psi_f(x_1, \ldots, x_n, z) \vee z = y)$$

which is a $\forall$-formula. ⊣

Example 9.2. The binary coprimality relation "coprime" is $\mathbb{N}$-conform. To see this, first notice that by the previous example, we have that the defining formula of the divisibility relation is a Δ-formula, and therefore, by COROLLARY 9.3, "$|$" is $\mathbb{N}$-conform. Furthermore, by COROLLARY 8.20, the defining formula of "coprime" is equivalent to a Δ-formula, and therefore, by COROLLARY 9.3, the binary relation "coprime" is $\mathbb{N}$-conform.

Gödel's β-Function

The main goal of this section is to define a binary function (the so-called **β-function** introduced by Kurt Gödel) which encodes a FINITE sequence of natural numbers $c_0, \cdots, c_{n-1}$ in the standard model by a single number c such that for all $i < n$,

$$\mathsf{PA} \vdash \beta(\underline{c}, \underline{i}) = \underline{c_i}.$$

In fact, one can even do better than that and introduce a function β, such that for every unary function f definable in Peano Arithmetic,

$$\mathsf{PA} \vdash \forall k \exists c \forall i < k\big(\beta(c, i) = f(i)\big).$$

The first step is to encode *ordered pairs* of numbers by introducing a binary pairing function "op". We define

$$\mathrm{op}(x, y) = z :\Longleftrightarrow (x + y) \cdot (x + y) + x + \mathtt{1} = z.$$

Furthermore, we define the unary relation *not an ordered pair* "nop" and the two binary functions *first element* "fst" and *second element* "snd" by stipulating

$$\begin{aligned}
\mathrm{nop}(c) &:\Longleftrightarrow \neg\exists x \exists y\big(\mathrm{op}(x, y) = c\big)\,, \\
\mathrm{fst}(c) = x &:\Longleftrightarrow \exists y\big(\mathrm{op}(x, y) = c\big) \vee \big(\mathrm{nop}(c) \wedge x = \mathtt{0}\big)\,, \\
\mathrm{snd}(c) = y &:\Longleftrightarrow \exists x\big(\mathrm{op}(x, y) = c\big) \vee \big(\mathrm{nop}(c) \wedge y = \mathtt{0}\big)\,.
\end{aligned}$$

In particular, whenever $\mathrm{op}(x, y) = c$, then

$$c = \mathrm{op}\big(\mathrm{fst}(c), \mathrm{snd}(c)\big)\,.$$

Until now, we did not show that the definitions above are well-defined, however, this follows from the following

LEMMA 9.4. $\mathsf{PA} \vdash \mathrm{op}(x,y) = \mathrm{op}(x',y') \to x = x' \wedge y = y'$.

Proof. Assume that $\mathrm{op}(x,y) = \mathrm{op}(x',y')$. We show first that this implies $x + y = x' + y'$: Suppose towards a contradiction that $x + y < x' + y'$. Then, by PA_3 and LEMMA 8.6, we obtain $\mathtt{s}(x+y) = x + \mathtt{s}y \leq x' + y'$. Therefore,

$$\begin{aligned}
\mathrm{op}(x',y') = \mathrm{op}(x,y) &= (x+y)\cdot(x+y) + x + \mathtt{1} \\
&\leq (x+\mathtt{s}y)\cdot(x+y) + (x+\mathtt{s}y) \\
&= (x+\mathtt{s}y)\cdot(x+\mathtt{s}y) \\
&= \mathtt{s}(x+y)\cdot\mathtt{s}(x+y) \\
&\leq (x'+y')\cdot(x'+y') \\
&< \mathrm{op}(x',y'),
\end{aligned}$$

which is obviously a contradiction. By symmetry, the relation $x' + y' < x + y$ can also be ruled out, and therefore we have that $\mathrm{op}(x,y) = \mathrm{op}(x',y')$ implies $x+y = x'+y'$. Now, if $x+y = x'+y'$, then $(x+y)\cdot(x+y) = (x'+y')\cdot(x'+y')$, and since $\mathrm{op}(x,y) = \mathrm{op}(x',y')$, by LEMMA 8.3 we obtain $x + \mathtt{1} = x' + \mathtt{1}$, which implies $x = x'$ and also $y = y'$. ⊣

Now we are ready to define the β-function. Let

$$\gamma(a,i,y,x) :\equiv \big(1 + (\mathrm{op}(a,i)+1)\cdot y\big) \mid x$$

and define

$$\begin{aligned}
\beta(c,i) = a \;:\Longleftrightarrow\; &\big(\mathrm{nop}(c) \wedge a = \mathtt{0}\big) \vee \\
&\exists x \exists y \Big(c = \mathrm{op}(x,y) \wedge \Big(\big(\neg\exists b\, \gamma(b,i,y,x) \wedge a = \mathtt{0}\big) \vee \\
&\qquad \big(\gamma(a,i,y,x) \wedge \neg\exists b < a\, \gamma(b,i,y,x)\big)\Big)\Big).
\end{aligned}$$

Observe that as a consequence of the LEAST NUMBER PRINCIPLE, β is a binary function.

Before we can encode finite sequences with the β function, we have to prove a few auxiliary results. The first one states that for every m there exists a y which is a multiple of $\mathrm{lcm}(1,\ldots,m)$.

LEMMA 9.5. $\mathsf{PA} \vdash \forall m \exists y \forall k \big((k \neq \mathtt{0} \wedge k \leq m) \to k \mid y\big)$.

Proof. We proceed by induction on m. The case when $m = \mathtt{0}$ is clear. Assume that there is an y such that for every k with $\mathtt{0} < k \leq m$ we have $k \mid y$. Let $y' = y \cdot \mathtt{s}m$. Then, by LEMMA 8.11, every k with $\mathtt{0} < k \leq \mathtt{s}m$ divides y'. ⊣

As described in Chapter 6, for any $\mathscr{L}_{\mathsf{PA}}$-formula φ which is *functional*, *i.e.*, $\mathsf{PA} \vdash \forall x \exists! y \varphi(x, y)$, we can introduce a function symbol F_φ by stipulating

$$F_\varphi(x) = y :\Longleftrightarrow \varphi(x, y)\,.$$

If F is defined by some functional $\mathscr{L}_{\mathsf{PA}}$-formula, then we say that F is **definable** in PA.

The next result shows that for every function F which is definable in PA and for every $k > \mathtt{0}$, we can define $\max\big\{F(\mathtt{0}), \ldots, F(k - \mathtt{1})\big\}$.

LEMMA 9.6. *Let F be a function which is definable in* PA. *Then*

$$\mathsf{PA} \vdash \forall k > \mathtt{0}\, \exists! x \big(\exists i < k \big(F(i) = x\big) \wedge \forall i < k \big(F(i) \leq x\big)\big).$$

Proof. We prove the statement by induction on k starting with $\mathtt{1}$. For $k = \mathtt{1}$ one can clearly take $x = F(\mathtt{0})$. Assume that there is a unique x and there is $i_0 < k$ such that $F(i_0) = x$ and for all $i < k$, $F(i) \leq x$. Now if $F(k) \leq x$, then set $x' = x$; otherwise let $x' = F(k)$. Then for every $i \leq \mathtt{s}k$, $F(i) \leq x$ and the first condition is also satisfied since x' is either $F(i_0)$ or $F(k)$; uniqueness is trivial. ⊣

This leades to the following definition:

$$\max\nolimits_{i<k} F(i) = x :\Longleftrightarrow \exists i < k \big(F(i) = x\big) \wedge \forall i < k \big(F(i) \leq x\big)$$

The next result plays an important role in the coding of finite sequences.

LEMMA 9.7. *Let G be an unary, strictly increasing function which is definable in* PA *and let $\varphi(\nu)$ be an $\mathscr{L}_{\mathsf{PA}}$-formula. Then*

$$\mathsf{PA} \vdash \forall m \Big(\forall j < m\, \forall j' < m \big(j \neq j' \to \mathrm{coprime}(G(j), G(j'))\big) \\ \to \exists x\, \forall j < m\, \big(G(j) \mid x \leftrightarrow \varphi(j)\big)\Big).$$

Proof. We proceed by induction on m starting with $m = 1$. If $\varphi(\mathtt{0})$ holds, let $x := G(\mathtt{0})$, otherwise, let $x := 1$. For the induction step assume that for all distinct $j, j' \leq \mathtt{s}m$, $G(j)$ and $G(j')$ are coprime and that there is an x such that for all $j < m$, $G(j) \mid x \leftrightarrow \varphi(j)$. By the LEAST NUMBER PRINCIPLE, let x_0 be the least such x. Now we consider the following two cases: If $\varphi(m)$ holds, let $x_1 := G(m) \cdot x_0$, otherwise, let $x_1 := x_0$. It remains to show that for all $j \leq m$, $G(j) \mid x_1 \leftrightarrow \varphi(j)$.

If $\varphi(m)$ fails (*i.e.*, $x_1 = x_0$), then, by induction hypothesis, for all $j < m$ we have $G(j) \mid x_0 \leftrightarrow \varphi(j)$ and $\mathrm{coprime}(G(j), G(m))$, where the latter implies by the choice of x_0 that $G(m) \nmid x_0$. To see this, assume that $G(m) \mid x_0$. Then there is an r such that $G(m) \cdot r = x_0$, and since $m \geq \mathtt{1}$, G is strictly increasing and $G(\mathtt{0}) \neq \mathtt{0}$ by $\mathrm{coprime}(G(\mathtt{0}), G(\mathtt{1}))$, we get that $G(m) > \mathtt{1}$ and consequently $r < x_0$. Moreover, since for all $j < m$ we have $\mathrm{coprime}(G(j), G(m))$, this implies

$$\forall j < m\big(G(j) \mid \underbrace{G(m) \cdot r}_{= x_0} \;\leftrightarrow\; G(j) \mid r\big)\,,$$

which contradicts the minimality of x_0.

If $\varphi(m)$ holds (*i.e.*, $x_1 = G(m) \cdot x_0$), then, since $\mathrm{coprime}(G(j), G(m))$ for all $j < m$ we have

$$G(j) \mid \underbrace{G(m) \cdot x_0}_{= x_1} \;\leftrightarrow\; G(j) \mid x_0\,.$$

Furthermore, we obviously have $G(m) \mid x_1$, which completes the proof ⊣

The following theorem states how the β-function can be used to code finite sequences.

THEOREM 9.8. *Let F be a function which is definable in* PA. *Then*

$$\mathsf{PA} \vdash \forall k\, \exists c\, \forall i < k\, \big(\beta(c, i) = F(i)\big).$$

Proof. Fix an arbitrary number k. Let $F'(i) := \mathrm{op}\big(F(i), i\big) + 1$ and let

$$m := \max\nolimits_{i<k} F'(i)\,.$$

By LEMMA 9.5 there is a y such that every $j \leq m$ divides y. Furthermore, by LEMMA 8.14 we have for all u with $u \mid y$ (*i.e.*, $1 \leq u \leq m$), and for all w

$$\mathrm{coprime}\big(1 + wy, 1 + (w + u)y\big)\,.$$

In particular, if $i < j < m$, then for $w := i + 1$ and $u := j - i$, we obtain

$$\mathrm{coprime}\big(1 + (i + 1)y, 1 + (j + 1)y\big)\,.$$

Finally, define the unary function G by

$$G(j) = z \;:\Longleftrightarrow\; z = 1 + (j + 1)y\,,$$

and let

$$\varphi_0(z) \;:\equiv\; \exists i < k\big(z = \mathrm{op}(F(i), i)\big)\,.$$

Then G is a strictly increasing function and we can apply LEMMA 9.7 in order to find a number x, such that for all $j < m$, where $m \geq \mathrm{op}\big(F(i), i\big)$ (for all $i < k$), we have

$$G(j) \mid x \;\leftrightarrow\; \varphi_0(j)\,,$$

in other words,

$$\forall j < m\,\Big(1 + (j + 1)y \mid x \;\leftrightarrow\; \exists i < k\,\big(j = \mathrm{op}(F(i), i)\big)\Big)\,.$$

It remains to show that for $c = \mathrm{op}(x, y)$ we have $\beta(c, i) = F(i)$ for all $i < k$. By our assumption on x we have $1 + (\mathrm{op}(F(i), i) + 1)y \mid x$ and $\gamma\big(F(i), i, y, x\big)$.

So, it is enough to check that $F(i)$ is minimal with this property. Assume towards a contradiction that there is an $a < F(i)$ with $\gamma(a, i, y, x)$, *i.e.*,

$$\mathtt{1} + (\mathrm{op}(a, i) + \mathtt{1})y \mid x\,.$$

Then, by the formula φ_0, there is a j with $j = \mathrm{op}(a, i) = \mathrm{op}(F(i'), i')$ for some $i' < k$, and by LEMMA 9.4, we finally obtain $i = i'$ and $a = F(i') = F(i)$. ⊣

Note that all functions — in particular the β-function — that we have introduced in this section can be defined by an $\exists$-formula and therefore they are $\mathbb{N}$-conform.

Encoding Finite Sequences

This section has the aim to show how the β-function can be used to encode a finite sequence of number. But what is meant by the words "finite" and "number"? In the standard model this coincides with FINITENESS and the usual natural numbers. In general, however, this means that the sequence has limited length k for some k; *i.e.*, in non-standard models its length can actually be a non-standard number.

Now sequences of natural numbers can be viewed as functions from some $\{\mathtt{0}, \cdots, n\}$ to the natural numbers, where n is the length of the sequence. However, in PA we cannot specify the domain of a definable function, so we will use $\beta(\,\cdot\,, \mathtt{0})$ to encode the length of a sequence. Concretely, we will encode $\langle F(i) \mid i < n\rangle$ using some c (whose existence is guaranteed by THEOREM 9.8) such that

$$\begin{aligned} \beta(c, \mathtt{0}) &= n \\ \forall i < n(\beta(c, i + \mathtt{1}) &= F(i)). \end{aligned}$$

This motivates us to introduce the functions

$$\begin{aligned} \mathrm{lh}(c) &:\equiv \beta(c, \mathtt{0}) \\ c_i &:\equiv \beta(c, i + \mathtt{1}). \end{aligned}$$

We will also call $\mathrm{lh}(c)$ the *length* of c. Furthermore, we define s to be a *sequence*, (denoted $\mathrm{seq}(s)$), if s is the smallest code for $\langle s_i \mid i < \mathrm{lh}(s)\rangle$:

$$\mathrm{seq}(s) :\Longleftrightarrow \forall t < s(\mathrm{lh}(t) = \mathrm{lh}(s) \to \exists i < \mathrm{lh}(s)(t_i \neq s_i)).$$

Note that the defintion of seq assures that codes for finite sequences are unique, *i.e.*,

$$\mathsf{PA} \vdash \mathrm{seq}(s) \wedge \mathrm{seq}(s') \wedge \mathrm{lh}(s) = \mathrm{lh}(s') \wedge \forall i < \mathrm{lh}(s)(s_i = s'_i) \to s = s'.$$

Example 9.3. The simplest example is the empty sequence $\langle\,\rangle$ which is defined by $\langle\,\rangle = s :\Longleftrightarrow \mathrm{seq}(s) \wedge \mathrm{lh}(s) = \mathtt{0}$. By taking a closer look at the definition of the

β-function, one can easily see that $\langle\,\rangle$ is actually 0, since it is the smallest code s with $\beta(s,0)=0$.

Secondly, we consider one-element sequences: The sequence just consisting of x is given by

$$\langle x\rangle = s :\Longleftrightarrow \mathrm{seq}(s)\wedge \mathrm{lh}(s)=1\wedge s_0=x.$$

In the same way, one can define two-element sequences as

$$\langle x,y\rangle = s :\Longleftrightarrow \mathrm{seq}(s)\wedge \mathrm{lh}(s)=2\wedge s_0=x\wedge s_1=y.$$

More generally, if F is definable in PA, then one can define

$$\langle F(i)\mid i<k\rangle = s :\Longleftrightarrow \mathrm{seq}(s)\wedge \mathrm{lh}(s)=k\wedge \forall i<k(s_i=F(i)).$$

THEOREM 9.8 assures that such s always exists and since it is the least code it is unique.

The functions $c,i\mapsto c_i,\mathrm{lh}$ and $\langle\cdot\rangle$ are all defined by $\exists$-formulae and are thus $\mathbb{N}$-conform as a consequence of COROLLARY 9.3. We will use the same notation for the corresponding function in $\mathbb{N}$, for example we write $\langle n,m\rangle$ for $\langle n,m\rangle^{\mathbb{N}}$.

Next we show how finite sequences can be concatenated.

PROPOSITION 9.9. *It holds that*

$$\mathsf{PA}\vdash \forall s\forall s'\exists t(\,\mathrm{seq}(t)\wedge \mathrm{lh}(t)=\mathrm{lh}(s)+\mathrm{lh}(s')\wedge\forall i<\mathrm{lh}(s)(t_i=s_i\wedge \forall i<\mathrm{lh}(s')(t_{\mathrm{lh}(s)+i}=s'_i)).$$

Proof. Put $f(0)=\mathrm{lh}(s)+\mathrm{lh}(t)$, $F(i)=\beta(s,i)$ for $0<i<\mathrm{lh}(s)+1$, and $F(i)=\beta(t,i-\mathrm{lh}(s))$ for $i\geq \mathrm{lh}(s)+1$. This clearly defines a function, so we can apply THEOREM 9.8 and obtain a code t such that for all $i<\mathrm{lh}(s)+\mathrm{lh}(t)+1,\beta(t,i)=F(i)$. In particular, this means that $\mathrm{lh}(t)=\mathrm{lh}(s)+\mathrm{lh}(s')$, $(t)_i=\beta(t,i+1)=\beta(s,i+1)=s_i$ for $i<\mathrm{lh}(s)$. Similarly, we get $t_{\mathrm{lh}(s)+i}=s'_i$ for $i<\mathrm{lh}(s')$. The LEAST NUMBER PRINCIPLE then enables us to choose t minimal with the properties from above, *i.e.*, so that $\mathrm{seq}(t)$. ⊣

We define

$$s^\frown s'=t :\Longleftrightarrow \mathrm{seq}(t)\wedge \mathrm{lh}(t)=\mathrm{lh}(s)+\mathrm{lh}(s')\wedge\forall i<\mathrm{lh}(s)(t_i=(s)_i)\wedge \forall i<\mathrm{lh}(s')(t_{\mathrm{lh}(s)+i}=s'_i).$$

Note that this is indeed functional by PROPOSITION 9.9. Moreover, it is easy to check that concatenation is associative, *i.e.*,

$$\mathsf{PA}\vdash (s^\frown s')^\frown s''=s^\frown(s'^\frown s'').$$

Therefore, we can omit brackets and write $s^\frown s'^\frown s''$ instead of $s^\frown(s'^\frown s'')$.

Encoding Power Functions

In the previous paragraphs we have seen how the β-function allows us to encode finite sequences. Now we will use this to show how recursive functions can be defined in PA. We will not do this in general, since the only crucial function we need is the power function. Further examples of recursive functions are to be found in the exercises.

The definability of the power function is remarkable, since it means that we can define exponentiation from addition and multiplication; however, as we will see in Chapter **??**, multiplication cannot be defined from addition. The idea is to interpret the power x^k as the sequence $\langle 1, x, \cdots, x^{k-1}, x^k\rangle$ of length $k+1$.

We introduce the function

$$x^k = y \;:\Longleftrightarrow \exists t \exists k \big(\mathrm{seq}(t) \wedge \mathrm{lh}(t) = \mathtt{s}k \wedge t_0 = 1 \wedge \forall i < k (t_{\mathtt{s}i} = x \cdot t_i) \wedge t_k = y\big).$$

Why is this functional? Clearly, the function x^k has (if defined) a unique value. In order to see that it is always defined, we can use induction: For $k = 0$ it is clear. Now assume that there is a sequence s of length $k+1$ such that $s_0 = 1$ and for all $i < k$, $s_{\mathtt{s}i} = x \cdot s_i$. Consider $t = s^\frown\langle x \cdot s_k\rangle$. Then t is a sequence of length $k+2$ which satisfies the desired properties.

Note that the power function is defined by an $\exists$-formula and therefore it is $\mathbb{N}$-conform by COROLLARY 9.3. Furthermore, oberve that the power function fulfills the usual recursive definition, *i.e.*,

$$\mathsf{PA} \vdash \forall x (x^0 = 1)$$
$$\mathsf{PA} \vdash \forall x \forall k (x^{\mathtt{s}k} = x \cdot x^k).$$

Now our aim is to encode terms, formulae and proofs by making use of unique prime decomposition. This can be shown in PA. However, for us it suffices to show that the function mapping x, y, z to $2^x \cdot 3^y \cdot 5^z$ is injective. The general result is left as an excercise to the interested reader. We define primality by

$$\mathrm{prime}(x) :\Longleftrightarrow x > 1 \wedge \forall z \big(z \mid x \to (z = x \vee z = 1)\big).$$

If $\mathrm{prime}(x)$, we say that x is *prime*. Note that prime can be defined by an $\exists$-formula, since "$\forall z$" can be replaced with "$\forall z \leq x$". Now using that $2, 3$ and 5 are prime in the standard model $\mathbb{N}$ and PROPOSITION 9.2 we obtain

$$\mathsf{PA} \vdash \mathrm{prime}(2) \wedge \mathrm{prime}(3) \wedge \mathrm{prime}(5).$$

Moreover, prime decomposition up to 5 is easily seen to be unique:

$$\mathsf{PA} \vdash 2^x \cdot 3^y \cdot 5^z = 2^{x'} \cdot 3^{y'} \cdot 5^{z'} \to x = x' \wedge y = y' \wedge z = z'.$$

This is usually proved by induction on $x + y + z$. Note that the simplest way to achieve this is use the characterization

$$\mathsf{PA} \vdash \mathrm{prime}(x) \leftrightarrow \forall y \forall z (x \mid yz \to x \mid y \vee x \mid z)$$

of primality (see EXERCISE 9.1). This is an easy consequence of BÉZOUT'S LEMMA (see EXERCISE 8.2)

Encoding Terms and Formulae

In a first step, every logical and non-logical symbol ζ of Peano Arithmetic is assigned a natural number $\#\zeta$ in $\mathbb{N}$, called **Gödel number** of ζ. Since from now on, we will switch ofthen between the meta-level and the formal level, we will always explicitly mention whenever we are reasoning formally, *i.e.*, within PA. Otherwise the proofs will be on the meta-level.

Symbol ζ	Gödel number $\#\zeta$
$\mathsf{0}$	$\mathbf{0}$
s	$\mathbf{2}$
$+$	$\mathbf{4}$
$\cdot$	$\mathbf{6}$
$=$	$\mathbf{8}$
$\neg$	$\mathbf{10}$
$\wedge$	$\mathbf{12}$
$\vee$	$\mathbf{14}$
$\to$	$\mathbf{16}$
$\exists$	$\mathbf{18}$
$\forall$	$\mathbf{20}$
v_0	$\mathbf{1}$
v_1	$\mathbf{3}$
$\vdots$	$\vdots$
v_n	$\mathbf{2} \cdot n + \mathbf{1}$

In the previous section, we have introduced power functions in PA. Since $\mathbb{N} \models \mathsf{PA}$, such functions also exist in $\mathbb{N}$; and we will use the same notation n^k as in PA. By $\mathbb{N}$-conformity we have $\mathsf{PA} \vdash \underline{n^k} = \underline{n}^k$ for all $n, k \in \mathbb{N}$. Note that by THEOREM 1.2 it would already suffice to just gödelize the logical operators $\neg, \wedge$ and $\exists$. Next we encode terms and formulae.

Term τ	Gödel number $\#\tau$
$\mathsf{0}$	$\mathbf{0}$
v_n	$\mathbf{2} \cdot n + \mathbf{1}$
$\mathsf{s}t$	$\mathbf{2}^{\#\mathsf{s}} \cdot \mathbf{3}^{\#t}$
$t_0 + t_1$	$\mathbf{2}^{\#+} \cdot \mathbf{3}^{\#t_0} \cdot \mathbf{5}^{\#t_1}$
$t_0 \cdot t_1$	$\mathbf{2}^{\#\cdot} \cdot \mathbf{3}^{\#t_0} \cdot \mathbf{5}^{\#t_1}$

Formula φ	Gödel number $\#\varphi$
$\tau_0 = \tau_1$	$\mathbf{2}^{\#=} \cdot \mathbf{3}^{\#\tau_0} \cdot \mathbf{5}^{\#\tau_1}$
$\neg\psi$	$\mathbf{2}^{\#\neg} \cdot \mathbf{3}^{\#\psi}$
$\psi_0 \wedge \psi_1$	$\mathbf{2}^{\#\wedge} \cdot \mathbf{3}^{\#\psi_1} \cdot \mathbf{5}^{\#\psi_1}$
$\psi_0 \vee \psi_1$	$\mathbf{2}^{\#\vee} \cdot \mathbf{3}^{\#\psi_0} \cdot \mathbf{5}^{\#\psi_1}$
$\psi_0 \to \psi_1$	$\mathbf{2}^{\#\to} \cdot \mathbf{3}^{\#\psi_0} \cdot \mathbf{5}^{\#\psi_1}$
$\exists x\psi$	$\mathbf{2}^{\#\exists} \cdot \mathbf{3}^{\#x} \cdot \mathbf{5}^{\#\psi}$
$\forall x\psi$	$\mathbf{2}^{\#\forall} \cdot \mathbf{3}^{\#x} \cdot \mathbf{5}^{\#\psi}$

Observe that by the uniqueness of the prime decomposition up to $\mathsf{5}$, every natural number encodes at most one variable, term or formula. So far, we have only assigned each symbol, term and formula a natural number in the standard model. However, we want to do this within Peano Arithmetic. This can be achieved by stipulating

$$\ulcorner\zeta\urcorner :\equiv \underline{\#\zeta}$$

for an arbitrary symbol, term or formula ζ. This allows us to express in PA that some number is the code of a variable, term or formula. However, we can easily formalize this so-called **Gödelization** process, where $\mathsf{2} :\equiv \mathsf{ss0}$, $\mathsf{3} :\equiv \mathsf{sss0}$, and $\mathsf{5} :\equiv \mathsf{sssss0}$.

$$\begin{aligned}
\mathrm{succ}(n) &:\equiv \mathsf{2}^{\ulcorner\mathsf{s}\urcorner} \cdot \mathsf{3}^{n} & \mathrm{add}(n,m) &:\equiv \mathsf{2}^{\ulcorner+\urcorner} \cdot \mathsf{3}^{n} \cdot \mathsf{5}^{m} \\
\mathrm{mult}(n,m) &:\equiv \mathsf{2}^{\ulcorner\cdot\urcorner} \cdot \mathsf{3}^{n} \cdot \mathsf{5}^{m} & \mathrm{eq}(t,t') &:\equiv \mathsf{2}^{\ulcorner=\urcorner} \cdot \mathsf{3}^{t} \cdot \mathsf{5}^{t'} \\
\mathrm{not}(f) &:\equiv \mathsf{2}^{\ulcorner\neg\urcorner} \cdot \mathsf{3}^{f} & \mathrm{and}(f,f') &:\equiv \mathsf{2}^{\ulcorner\wedge\urcorner} \cdot \mathsf{3}^{f} \cdot \mathsf{5}^{f'} \\
\mathrm{or}(f,f') &:\equiv \mathsf{2}^{\ulcorner\vee\urcorner} \cdot \mathsf{3}^{f} \cdot \mathsf{5}^{f'} & \mathrm{imp}(f,f') &:\equiv \mathsf{2}^{\ulcorner\to\urcorner} \cdot \mathsf{3}^{f} \cdot \mathsf{5}^{f'} \\
\mathrm{ex}(v,f) &:\equiv \mathsf{2}^{\ulcorner\exists\urcorner} \cdot \mathsf{3}^{v} \cdot \mathsf{5}^{f} & \mathrm{all}(v,f) &:\equiv \mathsf{2}^{\ulcorner\forall\urcorner} \cdot \mathsf{3}^{v} \cdot \mathsf{5}^{f}
\end{aligned}$$

Now we are ready to provide a formalised version of the term and formula construction:

$$\begin{aligned}
\mathrm{var}(v) &:\Longleftrightarrow \exists n(v = \mathsf{2} \cdot n + \mathsf{1}) \\
\mathrm{c_term}(c,t) &:\Longleftrightarrow \mathrm{seq}(c) \wedge c_{\mathrm{lh}(c)-1} = t \wedge \forall k < \mathrm{lh}(c)(\mathrm{var}(c_k) \vee c_k = \mathsf{0} \\
&\quad \vee \exists i < k \exists j < k(c_k = \mathrm{succ}(c_i) \vee c_k = \mathrm{add}(c_i, c_j) \\
&\quad \vee c_k = \mathrm{mult}(c_i, c_j)) \\
\mathrm{term}(t) &:\Longleftrightarrow \exists c(\mathrm{c_term}(c,t)) \\
\mathrm{c_fml}(c,f) &:\Longleftrightarrow \mathrm{seq}(c) \wedge c_{\mathrm{lh}(c)-\mathsf{1}} = f \\
&\quad \wedge \forall k < \mathrm{lh}(c)(\exists t, t'(\mathrm{term}(t) \wedge \mathrm{term}(t') \wedge c_k = \mathrm{eq}(t,t')) \\
&\quad \vee \exists i < k \exists j < k \exists v(c_k = \mathrm{not}(c_i) \vee c_k = \mathrm{and}(c_i, c_j) \\
&\quad \vee c_k = \mathrm{or}(c_i, c_j) \vee c_k = \mathrm{imp}(c_i, c_j) \vee c_k = \mathrm{ex}(v, c_i) \\
&\quad \vee c_k = \mathrm{all}(v, c_i). \\
\mathrm{fml}(f) &:\Longleftrightarrow \exists c(\mathrm{c_fml}(c,f)).
\end{aligned}$$

Note that all relations above are defined by an $\exists$-formula.

Example 9.1 Consider the term $\tau \equiv \mathtt{s}v_n + \mathtt{0}$. In the standard model $\mathbb{N}$, the sequence code $c \equiv \langle \#v_n, \#\mathtt{s}v_n, \#\mathtt{0}, \#\mathtt{s}v_n + \mathtt{0} \rangle$ codes τ, *i.e.*, $\mathbb{N} \vDash \mathrm{c\_term}(c, \#\tau)$. Using PROPOSITION 9.2 this implies $\mathsf{PA} \vdash \mathrm{c\_term}(\underline{c}, \ulcorner \tau \urcorner)$.

LEMMA 9.10. *For $n \in \mathbb{N}$ we have*

(a) $\mathbb{N} \vDash \mathrm{var}(n)$ *if and only if* $n \equiv \#\nu$ *for some variable* ν.

(b) $\mathbb{N} \vDash \mathrm{term}(n)$ *if and only if* $n \equiv \#\tau$ *for some* $\mathscr{L}_{\mathsf{PA}}$*-term* τ.

(c) $\mathbb{N} \vDash \mathrm{fml}(n)$ *if and only if* $n \equiv \#\varphi$ *for some* $\mathscr{L}_{\mathsf{PA}}$*-formula* φ.

Proof. Condition (a) is obvious. For (b), we prove first that $\mathbb{N} \vDash \mathrm{term}(\#\tau)$ for every term τ. We proceed by induction on the term construction of τ. If $\tau \equiv \mathtt{0}$ or τ is a variable, then clearly $\mathbb{N} \vDash \mathrm{c\_term}(\langle \#\tau \rangle, \#\tau)$ and hence the claim follows. Now if $\tau \equiv \mathtt{s}\tau'$ for some term τ' with $\mathbb{N} \vDash \mathrm{term}(\#\tau')$, and $c \in \mathbb{N}$ is a code with $\mathbb{N} \vDash \mathrm{c\_term}(c, \#\tau')$, then $\mathbb{N} \vDash \mathrm{succ}(\#\tau', \#\tau)$ and hence $\mathbb{N} \vDash \mathrm{c\_term}(c^\frown \langle \#\tau \rangle, \#\tau)$. The other cases are similar. For the converse, we use the principle of strong induction in $\mathbb{N}$. Suppose that the claim holds for all $m < n$ in $\mathbb{N}$ and let $\mathbb{N} \vDash \mathrm{term}(n)$. If $n \equiv \mathbf{0}$ then $n \equiv \#\mathtt{0}$ and if $n \equiv 2m + 1$ for some m, then $n \equiv \#v_m$. Let $\mathbb{N} \vDash \mathrm{c\_term}(c, n)$ for some $c \in \mathbb{N}$ with $\mathrm{lh}(c) > 1$. Now in $\mathbb{N}$ we either have $n \equiv \mathrm{succ}^{\mathbb{N}}(c_i)$ for some $i < \mathrm{lh}(c)$, $n \equiv \mathrm{add}^{\mathbb{N}}(c_i, c_j)$ or $n \equiv \mathrm{mult}^{\mathbb{N}}(c_i, c_j)$ for $i, j < \mathrm{lh}(c)$. In the first case, note that $\mathbb{N} \vDash \mathrm{c\_term}(\langle c_k \mid k < \mathbf{s}i \rangle, c_i)$. By our inductive hypothesis, we can take a term τ such that $c_i \equiv \#\tau$. But then by $\mathbb{N}$-conformity we have $n \equiv \mathrm{succ}^{\mathbb{N}}(c_i) \equiv (2^{\ulcorner \mathtt{s} \urcorner} \cdot 3^{\underline{c_i}})^{\mathbb{N}} \equiv 2^{\#\mathtt{s}} \cdot 3^{c_i} \equiv \#\mathtt{s}\tau$ and hence n codes $\mathtt{s}\tau$. The other cases are similar.

The corresponding statement for formulae is proved in the same way and is therefore left as an exercise. ⊣

Note that the relations var, term and formula are $\exists$-formulae, so by combining LEMMA 9.10 and PROPOSITION 9.1 we obtain: If ν is a variable, τ is a term and φ is a formula, then

$$\begin{aligned} &\mathsf{PA} \vdash \mathrm{var}(\ulcorner \nu \urcorner) \\ &\mathsf{PA} \vdash \mathrm{term}(\ulcorner \tau \urcorner) \\ &\mathsf{PA} \vdash \mathrm{formula}(\ulcorner \varphi \urcorner). \end{aligned}$$

Before we proceed to gödelise logical axioms, the axioms of Peano Arithmetic and formal proofs, we have to deal with substitution.

Firstly, we introduce new relations which check wether a code for a variable appears in the code of a term or formula respectively.

$$\begin{aligned} \mathrm{var\_in\_term}(v, t) :\Longleftrightarrow \mathrm{var}(v) \wedge \exists c \big(&\mathrm{c\_term}(c, t) \wedge \exists i < \mathrm{lh}(c)(c_i = v \wedge \\ &\forall c' < c \neg \mathrm{c\_term}(c', t)) \big). \end{aligned}$$

Note that the minimality of c is necessary since otherwise any code for a variable could appear in the sequence of codes of the term construction. The same holds for the following relation var_in_fml.

$$\begin{aligned}\mathrm{var\_in\_fml}(v,f) :\Longleftrightarrow \exists c(\mathrm{c\_fml}(c,f) \wedge \exists i < \mathrm{lh}(c) \exists t_0 \exists t_1 \big(\, &\mathrm{term}(t_0) \wedge \mathrm{term}(t_1) \\ &\wedge c_i = \mathrm{eq}(t_0,t_1) \\ &\wedge (\mathrm{var\_in\_term}(v,t_0) \vee \mathrm{var\_in\_term}(v,t_1)) \\ &\wedge \forall c' < c \neg\, \mathrm{c\_fml}(c',f)\end{aligned}$$

$$\begin{aligned}\mathrm{free}(v,f) :\Longleftrightarrow \exists c\big(&\mathrm{c\_fml}(c,f) \wedge \mathrm{var\_in\_fml}(v,f) \wedge \forall i < \mathrm{lh}(c) \forall j < i \\ &(c_i \neq \mathrm{ex}(v,c_j) \wedge c_i \neq \mathrm{all}(v,c_j))\big)\end{aligned}$$

For the sake of simplicity, we only permit the substitution $\varphi(x/\tau)$, if it is admissible and x as well as all variables in τ appear only free in φ. This does not impose a restriction since by renaming of variables every formula is equivalent to one in which no variable occurs both bound and free. We can thus define

$$\begin{aligned}\mathrm{sb\_adm}(v,t,f) :\Longleftrightarrow\ &\mathrm{var\_in\_fml}(v,f) \wedge \mathrm{free}(v,f) \\ &\wedge \forall v' < t(\mathrm{var\_in\_term}(v',t) \to \mathrm{free}(v',f)).\end{aligned}$$

Note that the relations $\mathrm{var\_in\_term}$, $\mathrm{var\_in\_fml}$, free and $\mathrm{sb\_adm}$ are all $\exists$-formulae: The only unbounded universal quantifier appears in $\mathrm{sb\_adm}$, where $\mathrm{var\_in\_term}$ occurs negated, since $\mathrm{var\_in\_term}(v',t) \to \mathrm{free}(v',f)$. The existential quantifier in the definition of $\mathrm{var\_in\_term}(v',t)$ can, however, be replaced by a bounded one, since the code of t has to be smaller than the code of f.

The next relation expresses that c' encodes the construction of the term obtained from the term with code t by replacing every occurrence of the code v of a variable by the code t_0.

$$\begin{aligned}&\mathrm{c\_sb\_term}(c,c',c'',v,t_0,t,t') \\ &\quad :\Longleftrightarrow \mathrm{var}(v) \wedge \mathrm{c\_term}(c,t) \wedge \mathrm{c\_term}(c'',t_0) \wedge \mathrm{lh}(c') = \mathrm{lh}(c'') + \mathrm{lh}(c) \\ &\qquad \wedge \forall k < \mathrm{lh}(c'')(c'_k = c''_k) \wedge \forall k < \mathrm{lh}(c)(c_k = \mathsf{0} \to c'_{\mathrm{lh}(c'')+k} = \mathsf{0} \\ &\qquad \wedge c_k = v \to c'_{\mathrm{lh}(c'')+k} = t_0 \wedge (\mathrm{var}(c_k) \wedge c_k \neq v \to c'_k = c_k) \\ &\qquad \wedge \forall i < k \forall j < k(c_k = \mathrm{succ}(c_i) \to c'_{\mathrm{lh}(c'')+k} = \mathrm{succ}(c'_{\mathrm{lh}(c'')+i}) \\ &\qquad \wedge c_k = \mathrm{add}(c_i,c_j) \to c_{\mathrm{lh}(c'')+k} = \mathrm{add}(c'_{\mathrm{lh}(c'')+i}, c'_{\mathrm{lh}(c'')+j}) \\ &\qquad \wedge \mathrm{mult}(c_i,c_j) \to c_{\mathrm{lh}(c'')+k} = \mathrm{mult}(c'_{\mathrm{lh}(c'')+i}, c'_{\mathrm{lh}(c'')+j})).\end{aligned}$$

By omitting the codes c, c', c'' we can describe term substitution by

$$\mathrm{sb\_term}(v,t_0,t,t') :\Longleftrightarrow \exists c \exists c' \exists c''(\mathrm{c\_sb\_term}(c,c',c'',v,t_0,t,t')).$$

For formulae we proceed in the same fashion except that we have to make sure first that the substitution is admissible.

$$\begin{aligned}&\mathrm{c\_sb\_fml}(c,c',v,t_0,f,f') \\ &\quad :\Longleftrightarrow \mathrm{sb\_adm}(v,t_0,f) \wedge \mathrm{c\_fml}(c,f) \wedge \mathrm{c\_fml}(c',f') \wedge \mathrm{lh}(c') = \mathrm{lh}(c) \wedge\end{aligned}$$

$$\begin{aligned}&\forall k < \mathrm{lh}(c)(\forall t\forall t'\forall s\forall s'(c_k = \mathrm{eq}(t,t') \wedge \mathrm{sb\_term}(v,t_0,t,s)\wedge\\ &\mathrm{sb\_term}(v,t_0,t',s') \to c'_k = \mathrm{eq}(s,s')) \wedge \forall i < k\forall j < k\forall v\\ &(c_k = \mathrm{not}(c_i) \to c'_k = \mathrm{not}(c'_i) \wedge c_k = \mathrm{and}(c_i,c_j) \to c'_k = \mathrm{and}(c'_i,c'_j)\wedge\\ &c_k = \mathrm{or}(c_i,c_j) \to c'_k = \mathrm{or}(c'_i,c'_j) \wedge c_k = \mathrm{imp}(c_i,c_j) \to c'_k = \mathrm{imp}(c'_i,c'_j)\\ &\wedge c_k = \mathrm{ex}(v,c_i) \to c'_k = \mathrm{ex}(v,c'_i) \wedge c_k = \mathrm{all}(v,c_i) \to c'_k = \mathrm{all}(v,c'_i)).\end{aligned}$$

Again by leaving out the sequence codes, we define

$$\mathrm{sb\_fml}(v,t_0,f,f') :\Longleftrightarrow \exists c\exists c'(\mathrm{c\_sb\_fml}(c,c',v,t_0,f,f').$$

LEMMA 9.11. *Let τ and τ_0 be two terms, φ a formula and ν a variable such that the substitution $\varphi(\nu/\tau_0)$ is admissible. Then one has*

(a) $\mathsf{PA} \vdash \mathrm{sb\_term}(\ulcorner\nu\urcorner, \ulcorner\tau_0\urcorner, \ulcorner\tau\urcorner, t) \leftrightarrow t = \ulcorner\tau(\nu/\tau_0)\urcorner$

(b) $\mathsf{PA} \vdash \mathrm{sb\_fml}(\ulcorner\nu\urcorner, \ulcorner\tau_0\urcorner, \ulcorner\varphi\urcorner, f) \leftrightarrow f = \ulcorner\varphi(\nu/\tau_0)\urcorner$.

Proof. This follows from the definition of the relations $\mathrm{sb\_term}$ and $\mathrm{sb\_fml}$ using induction on the term construction of τ and the formula construction of φ. ⊣

Example 9.4. Let $\tau \equiv \mathtt{s}x+y$ and $\tau_0 \equiv \mathtt{0}$. Then the sequence $\langle\ulcorner x\urcorner, \ulcorner\mathtt{s}x\urcorner, \ulcorner y\urcorner, \ulcorner\mathtt{s}x+y\urcorner\rangle$ codes $\ulcorner\tau\urcorner$ and $\langle\ulcorner\mathtt{0}\urcorner\rangle$ codes τ_0. Now when coding $\tau(x/\tau_0)$, we take first the code of τ_0 and then replace every occurrence of x in the code of τ by τ_0. This gives

$$\langle\ulcorner\mathtt{0}\urcorner, \ulcorner\mathtt{s0}\urcorner, \ulcorner y\urcorner, \ulcorner\mathtt{s0}+y\urcorner\rangle$$

which codes $\tau(x/\tau_0)$.

Encoding Formal Proofs

Finally, we can apply the machinery developed in the previous section to encode axioms and formal proofs. We first show how to achieve this in $\mathbb{N}$: Recall that a formal proof of some $\mathscr{L}_{\mathsf{PA}}$-formula φ is a finite sequence $\varphi_0,\ldots,\varphi_n$ with $\varphi_n \equiv \varphi$ such that each φ_i is an instance of a logical axiom, an axiom of PA or it is obtained from preceding elements of the sequence using MODUS PONENS or GENERALISATION. Hence we can code a formal proof of φ by $\langle\#\varphi_0,\ldots,\#\varphi_n\rangle$. Conversely, from such a code we can recover the sequence $\varphi_0,\ldots,\varphi_n$ and hence reconstruct a formal proof of φ.

As for terms and formulae, we proceed to code formal proofs in PA. The goal is to define a relation prv with the property that $\mathbb{N} \vDash \mathrm{prv}(\ulcorner\varphi\urcorner)$ for some formula φ if and only if there is a formal proof of φ, *i.e.*, $\mathsf{PA} \vdash \varphi$. The following examples illustrate how axioms can be formalised in PA:

$$\mathrm{ax\_L_1}(f) :\Longleftrightarrow \exists f'\exists f''(\mathrm{fml}(f') \wedge \mathrm{fml}(f'') \wedge f = \mathrm{imp}(f',\mathrm{imp}(f'',f')))$$

$$\mathrm{ax\_L_{10}}(f) :\Longleftrightarrow \exists f'\exists f''\exists v\exists t(\mathrm{sb\_fml}(v,t,f',f'') \wedge f = \mathrm{imp}(\mathrm{all}(v,f'),f''))$$
$$\mathrm{ax\_L_{14}}(f) :\Longleftrightarrow \exists t(\mathrm{term}(t) \wedge f = \mathrm{eq}(t,t)).$$

PA_0 and the induction schema are gödelised as follows:

$$\begin{aligned}\mathrm{ax\_PA_0}(f) :\Longleftrightarrow & f = \ulcorner \forall v_0 \neg(\mathsf{s}v_0 = \mathsf{0})\urcorner\\ \mathrm{ax\_PA_6}(f) :\Longleftrightarrow & \exists f'\exists f''\exists f'''\exists v\exists g(\mathrm{free}(v,f') \wedge \mathrm{sb\_fml}(v,\ulcorner \mathsf{0}\urcorner,f',f'')\\ & \wedge \mathrm{sb\_fml}(v,\mathrm{succ}(v),f',f''') \wedge g = \mathrm{all}(v,\mathrm{imp}(f',f'''))\\ & \wedge f = \mathrm{imp}(\mathrm{and}(f'',g),\mathrm{all}(v,f')).\end{aligned}$$

We leave it to the reader to formalize the other axioms. We further define

$$\begin{aligned}\mathrm{log\_ax}(f) &:\Longleftrightarrow \mathrm{ax\_L_0}(f) \vee \cdots \vee \mathrm{ax\_L_{16}}(f)\\ \mathrm{peano\_ax}(f) &:\Longleftrightarrow \mathrm{ax\_PA_0}(f) \vee \cdots \vee \mathrm{ax\_PA_6}(f)\\ \mathrm{ax}(f) &:\Longleftrightarrow \mathrm{log\_ax}(f) \vee \mathrm{peano\_ax}(f).\end{aligned}$$

Next we formalize the inference rules MODUS PONENS and GENERALISATION by stipulating

$$\begin{aligned}\mathrm{mp}(f',f'',f) &:\Longleftrightarrow \mathrm{fml}(f') \wedge \mathrm{fml}(f) \wedge f'' = \mathrm{imp}(f',f)\\ \mathrm{gen}(v,f',f) &:\Longleftrightarrow \mathrm{var}(v) \wedge \mathrm{fml}(f') \wedge f = \mathrm{all}(v,f').\end{aligned}$$

Finally, we encode formal proofs as sequences of codes of formulae which are either axioms or are produced by one of the inference rules. Thus we define the predicates $\mathrm{c\_prv}(c,f)$ in order to specify that c encodes a proof of the formula coded by f and prv to express provability.

$$\begin{aligned}\mathrm{c\_prv}(c,f) :\Longleftrightarrow & \mathrm{seq}(c) \wedge c_{\mathrm{lh}(c)-1} = f \wedge \forall k < \mathrm{lh}(c)\big(\mathrm{ax}(c_k) \vee \exists i < k\exists j < k\\ & \mathrm{mp}(c_i,c_j,c_k)) \vee \exists v(\mathrm{gen}(v,c_i,c_k))\big)\\ \mathrm{prv}(f) :\Longleftrightarrow & \exists c(\mathrm{prv}(c,f)).\end{aligned}$$

Note that in the standard model $\mathbb{N}$ we have $\mathbb{N} \vDash \mathrm{c\_prv}(c,\#\varphi)$, if and only if c encodes a sequence $\langle \#\varphi_0,\ldots,\#\varphi_n\rangle$, where $\varphi_0,\ldots,\varphi_n$ is a formal proof of φ.

LEMMA 9.12. *Let $n,c \in \mathbb{N}$ be natural numbers. Then $\mathbb{N} \vDash \mathrm{c\_prv}(c,n)$ if and only if c encodes a formal proof of some $\mathscr{L}_{\mathsf{PA}}$-formula φ with $\#\varphi = n$. In particular, $\mathbb{N} \vDash \mathrm{prv}(\#\varphi)$ if and only if $\mathsf{PA} \vdash \varphi$.*

Proof. Note first that $\mathbb{N} \vDash \mathrm{ax}(m)$ for some $m \in \mathbb{N}$ if and only if $m = \#\psi$ encodes some (instance) of a logical axiom or an axiom of PA. The proof is the same as the proof of LEMMA 9.10, where in the forward direction, one proceeds by induction on $\mathrm{lh}(c)$ and for the converse by induction on the length of the formal proof of φ.

For the second part, suppose that $\mathbb{N} \vDash \mathrm{prv}(\#\varphi)$. Then there is $c \in \mathbb{N}$ which encodes a formal proof of φ and hence $\mathsf{PA} \vdash \varphi$. ⊣

Note that LEMMA 9.12 does not hold, if we replace $\mathbb{N}$ by a non-standard model $\mathbf{M}$ of PA: If $\mathbf{M} \vDash \mathrm{c\_prv}(c, \ulcorner\varphi\urcorner)$ and $c^{\mathbf{M}}$ is a non-standard number, then we the "proof" encoded by $c^{\mathbf{M}}$ is of non-standard length and hence we cannot conclude that $\mathsf{PA} \vdash \varphi$.

COROLLARY 9.13. *Let φ be an $\mathscr{L}_{\mathsf{PA}}$-formula. If $\mathsf{PA} \vdash \varphi$ then there is $n \in \mathbb{N}$ such that $\mathsf{PA} \vdash \mathrm{c\_prv}(\underline{n}, \ulcorner\varphi\urcorner)$. In particular, if $\mathsf{PA} \vdash \varphi$ then $\mathsf{PA} \vdash \mathrm{prv}(\ulcorner\varphi\urcorner)$.*

Proof. Suppose that $\mathsf{PA} \vdash \varphi$. Then by LEMMA 9.12 we have $\mathbb{N} \vDash \mathrm{c\_prv}(c, \#\varphi)$ for some $c \in \mathbb{N}$. Observe that the relation c_prv and prv are defined by an $\exists$-formula. Hence it follows from PROPOSITION 9.2 that $\mathsf{PA} \vdash \mathrm{c\_prv}(\underline{c}, \ulcorner\varphi\urcorner)$ and in particular $\mathsf{PA} \vdash \mathrm{prv}(\ulcorner\varphi\urcorner)$. ⊣

NOTES

Kurt Gödel used in his proof of the incompleteness theorems [17] the β-function and unique prime decomposition in order to code finite sequences by a single number. There are, however, other ways to achieve this; *e.g.*, the coding provided by Smullyan in [**?**]. However, in our presentation we followed Shoenfield [40], or Boolos?

Finish this.

EXERCISES

9.0 Prove N_2.

9.1 (a) Show that $\mathsf{PA} \vdash \forall x \geq \mathtt{2}\, \exists y\big(\mathrm{prime}(y) \wedge y \mid x\big)$, *i.e.*, every $x \geq 2$ has a prime divisor.
Hint: Use the LEAST NUMBER PRINCIPLE.

(b) Show that $\mathsf{PA} \vdash \forall x(\mathrm{prime}(x) \leftrightarrow \forall y \forall z(x \mid yz \to x \mid y \vee x \mid z))$.
Hint: Use the PRINCIPLE OF DIVISION WITH REMAINDER.

9.2 Introduce a factorial function ! such that $n! = 1 \cdot \ldots \cdot n$ and show that it is $\mathbb{N}$-conform.

9.3 Introduce a function $\mathrm{lcm}_{i<k} F(i)$ for a function F definable in PA such that $\mathrm{lcm}_{i<k} F(i)$ is the least common multiple of $F(\mathtt{0}), \ldots, F(k - \mathtt{1})$ and show that it is $\mathbb{N}$-conform.

9.4 State and prove in PA that every number has a unique prime decomposition, *i.e.*, prove that every number is a product of primes, and show that this is unique up to permutations of the factors.

9.5 Give the Gödel code of the formula $\forall x(\mathtt{0} \neq x)$.

9.6 An alternative way to is to use the existence and uniqueness of base b notation for $b \geq 2$.

(a) Show that it suffices to gödelise only b symbols for some $b \in \mathbb{N}$.

(b) Prove (in PA) that for every number n there are $n_0, \ldots, n_k$ such that

$$n = n_k b^k + \ldots + n_1 b + n_0 \equiv: (n_k \ldots n_0)_b.$$