§9. Gödel’s Incompleteness Theorem I: Representing Primitive Recursive Functions

The following three problems can be considered the basic problems in the foundations of mathematics.

1) Every mathematical truth about the natural numbers should have a meaningful finitistic proof. While it is arguable what a “finitistic proof” is, one precise way of saying this is that there is a natural set of axioms from which we can derive all truths about the natural numbers. For example Peano Arithmetic would be a good candidate.

2) Hilbert’s Conservation Program: If a mathematical truth can be proved by strong methods (say using set theoretic methods), then it can be proved by finitistic methods.

3) Hilbert’s Consistency Program: We should be able to give a finitistic proof of the consistency of our methods.

Gödel showed that all of this is impossible.

Theorem 9.1. i) (First Incompleteness Theorem) There is a sentence \( \phi \) such that \( \mathbb{N} \models \phi \) and \( \text{PA} \nvdash \phi \). Indeed if \( T \) is a recursive theory such that \( T \supseteq \text{PA} \) and \( \mathbb{N} \models T \), then there is a sentence \( \phi \) such that \( T \not\vdash \phi \) and \( T \not\vdash \neg \phi \).

ii) (Second Incompleteness Theorem) Let \( T \) be as above. Then \( T \) does not prove the consistency of \( T \).

The first completeness theorem shows that i) will fail. The second incompleteness theorem shows that iii) fails. Since using set theory we can show that \( \text{PA} \) is consistent, iii) shows that the conservation program fails as well.

We begin by outlining the main steps of Gödel’s proof. Let \( \mathcal{L} \) be a recursive language containing \(+, \cdot, <, 0, 1\) the language of arithmetic and let \( T \) be an \( \mathcal{L} \)-theory. \( T \) will be our candidate for an axiomatization of a large part of mathematics. We assume \( T \) is sufficiently strong by requiring that \( T \) contains \( T_0 \), where \( T_0 \) is a theory which axiomatizes enough number theory for our purposes and \( \mathbb{N} \models T_0 \). We will take \( T_0 = \text{PA} \), though a much weaker theory would suffice. For \( T \) we can take \( \text{PA} \), the Zermelo-Frankel axioms for set theory, or some other strong system.

The following lemma is the key technical idea. For \( n \in \mathbb{N} \), let \( \hat{n} \) denote the \( \mathcal{L} \)-term \( 1 + \ldots + 1 \), if \( m_1, \ldots, m_n \in \mathbb{N} \), we let \( \hat{m} \) denote \( (\hat{m_1}, \ldots, \hat{m_n}) \).

Lemma 9.2: (Representation Lemma) We will show that if \( f(\vec{x}) \) is a primitive recursive function, then there is a formula \( \phi_f(\vec{x}, y) \) such that

i) \( f(\hat{n}) = m \iff T_0 \vdash \phi_f(\hat{n}, \hat{m}) \iff \mathbb{N} \models \phi_f(\hat{n}, \hat{m}) \).
ii) \( T_0 \vdash \forall \vec{x} \exists y \phi_f(\vec{x}, y) \), and
iii) \( T_0 \vdash \forall \vec{x} \forall y \forall z ((\phi_f(\vec{x}, y) \land \phi_f(\vec{x}, z)) \rightarrow y = z) \).

1
Lemma 9.2 and the results of §7 already allow us to deduce the impossibility of i). Suppose $R \subseteq \mathbb{N}^m$ is recursive. Suppose $R = W_e$. Then

$$x \in R \iff \exists s \ T(e, x, s)$$

where $T$ Kleene’s T-predicate.

Suppose $A \subseteq \mathbb{N}$ is $\Sigma_n$. Say

$$x \in A \iff \exists y_1 \forall y_2 \ldots Q y_n \ R(x, \bar{y})$$

where $R$ is recursive and $Q$ is $\exists$ if $n$ is odd and $\forall$ if $n$ is even.

Suppose without loss of generality that $n$ is odd. Then

$$x \in A \iff \exists y_1 \forall y_2 \ldots Q y_n \exists s \ T(e, x, \bar{y}, s)$$

Let $\psi_A(v)$ be the formula

$$\exists y_1 \forall y_2 \ldots Q y_n \exists s \ \phi_T(\bar{e}, v, \bar{y}, s)$$

where $\phi_T$ is the formula which represents $T$.

It is clear that

$$n \in A \iff \mathbb{N} \models \psi_A(\bar{n})$$

Let $Th(\mathbb{N}) = \{\psi : \mathbb{N} \models \psi\}$. The function

$$n \mapsto \psi_A(\bar{n})$$

gives a many-one reduction of $A$ to $Th(\mathbb{N})$. Thus we have proved:

**Lemma 9.3.** $Th(\mathbb{N})$ is not arithmetic.

**Proof.**

Suppose $Th(\mathbb{N})$ were $\Sigma_n$. By 7.15, there is a $\Sigma_{n+1}$ set $A$ which is not $\Sigma_n$. By the above arguments $A \leq_m Th(\mathbb{N})$. But by 7.11 vi), if $Th(\mathbb{N}) \in \Sigma_n$, then $A \in \Sigma_n$, a contradiction.

In particular we can now deduce the first incompleteness theorem.

**Corollary 9.4:** If $\mathbb{N} \models T$ and $T$ is recursive, then there is a sentence $\phi$ such that $\mathbb{N} \models \phi$ and $T \nvdash \phi$.

**Proof:**

If $T$ is recursive, then $\{\phi : T \vdash \phi\}$ is a recursively enumerable subset of $Th(\mathbb{N})$. Since $Th(\mathbb{N})$ is not recursively enumerable, there is a sentence $\phi$ such that $\mathbb{N} \models \phi$ but $T \nvdash \phi$.

Indeed, if $T \subseteq Th(\mathbb{N})$ is $\Sigma_n$, then $\{\phi : T \vdash \phi\}$ is $\Sigma_{n+1}$ and the above argument shows that there is a true sentence $\phi$ such that $T \nvdash \phi$. 

2
Our first task then is to prove the representation lemma. This will require yet another method of coding finite sequences. The goal is to have a method of coding finite sequences so that one can express properties of the codes easily in the language of arithmetic. We will use Gödel’s beta-function.

Let $Seq$ be the set of finite sequences of elements of $\mathbb{N}$. We define $\beta : \mathbb{N}^3 \to Seq$, such that $\beta(u, v, w)$ is the sequence $(a_0, \ldots, a_{w-1})$ where $a_i = u(\text{mod } (i + 1)v + 1)$ for $i = 0, \ldots, w - 1$ (and $\beta(u, v, w)$ is the empty sequence for $w = 0$).

Let $\Psi(u, v, w, i, x)$ be the formula

\[ i < w \land 0 \leq x < (i + 1)v + 1 \land \exists y \leq u \ y((i + 1)v + 1) + x = u. \]

Then $\Psi(u, v, w, i, x)$ expresses that $x$ is the $i^{th}$ element in the sequence $\beta(u, v, w)$. We will write $\beta(u, v, w)_i = x$ for $\Psi(u, v, w, i, x)$. While it is easy to express that $\beta(u, v, w)_i = x$, it is not so easy to see that every sequence is coded in this way.

The proof uses the following simple lemma from number theory.

**Lemma 9.5.** (Chinese Remainder Theorem) Suppose $m_1, \ldots, m_n$ are relatively prime. Then for any $a_1, \ldots, a_n$ there is an $x$ such that $x \equiv a_i \pmod{m_i}$ for $i = 1, \ldots, n$.

**Proof.**

Let $M_i = \prod_{j \neq i} m_j$.

Since $M_i$ and $m_i$ are relatively prime, we can find $b_i$ such that $b_i M_i \equiv 1 \pmod{m_i}$. Let

\[ x = \sum_{i=1}^{n} a_i b_i M_i. \]

Since $m_i | M_j$ for $j \neq i$, $x \equiv a_i b_i M_i \pmod{m_i}$. Thus $x \equiv a_i \pmod{m_i}$ for $i = 1, \ldots, n$.

**Lemma 9.6.** For any sequence $\sigma = (a_0, \ldots, a_{w-1})$ there are $u$ and $v$ such that $\beta(u, v, w) = \sigma$.

**Proof.**

Let $n = \max(w, a_0, \ldots, a_{w-1})$ and let $v = n!$. We claim that $v + 1, 2v + 1, \ldots, wv + 1$ are relatively prime. Suppose $p$ is prime and $p|iv + 1$ and $p|jv + 1$ where $j > i > 0$ Then $p|(j - i)v$ Thus $p|(j - i)$ or $p|v$ and, since $(j - i)|v$, $p|v$. But then $p \not| iv + 1$. Thus $v + 1, \ldots, wv + 1$ are relatively prime.

By the Chinese remainder theorem there is a number $u$ such that $u \equiv a_i \pmod{(i + 1)v + 1}$ for $i = 0, \ldots, w - 1$. 

3
The proof above is a very simple number theoretic argument that can easily be formalized in Peano Arithmetic (in particular the usual proof of the Chinese remainder theorem can be formalized in $PA$). Thus for any $w$

$$PA \vdash \forall a_0 \ldots \forall a_{w-1} \exists u \forall v < w \bigwedge_{i=0}^{w-1} \beta(u, v, w)_i = a_i.$$ 

We will use $PA$ as the weak base theory $T_0$. If we wanted to use a weaker base theory we would need to use a more subtle coding.

We can now prove the Representation Lemma. If $f : \mathbb{N}^m \to \mathbb{N}$ is a primitive recursive function we say that $\phi(x_1, \ldots, x_m, y)$ represents $f$ in $PA$ if and only if

i) $f(\bar{m}) = n$ if and only if $PA \vdash \phi(\bar{m}, n)$ if and only if $\mathbb{N} \models \phi(\bar{m}, n)$ for $\bar{m}, n \in \mathbb{N}$.

ii) $PA \vdash \forall x \exists y \phi(x, y)$, and

iii) $PA \vdash \forall x \forall y \forall z ((\phi(x, y) \land \phi(x, z)) \to y = z)$.

Conditions ii) and iii) assert that in any model of $PA$ the formula $\phi$ defines the graph of a function. Condition i) asserts that on the standard part of the model that function agrees with $f$.

We will prove that any primitive recursive function is represented in $PA$. This will be done by induction on the complexity of the definition of $f$. We begin with the basic functions.

- If $z$ is the zero function, let $\phi_z(x, y)$ be the formula

- If $s$ is the successor function, let $\phi_s(x, y)$ be the formula $y = x + 1$.

- If $\pi^n_i$ is the function $(x_1, \ldots, x_n) \mapsto x_i$, then $\phi_{\pi^n_i}(x_1, \ldots, x_n, y)$ is the formula $y = x_i$.

It should be clear if $f$ is a basic function then $\phi_f$ represents $\phi$.

- Suppose $g_1, \ldots, g_n : \mathbb{N}^m \to \mathbb{N}$ and $h : \mathbb{N}^n \to \mathbb{N}$ are primitive recursive and $f : \mathbb{N}^m \to \mathbb{N}$ by $f(\bar{x}) = h(g_1(\bar{x}), \ldots, g_n(\bar{x}))$. Suppose by induction that there are formulas $\phi_{g_1}, \ldots, \phi_{g_n}, \phi_h$ representing $g_1, \ldots, g_n$, and $h$ respectively. Let $\phi_f(\bar{x}, y)$ be the formula

$$\exists z_1 \ldots \exists z_n \bigwedge_{i=1}^{n} \phi_{g_i}(\bar{x}, z_i) \land \phi_h(z_1, \ldots, z_n, y).$$

Suppose $a_1, \ldots, a_m \in \mathbb{N}$ and $h(\bar{a}) = b$. There are $c_1, \ldots, c_n \in \mathbb{N}$ such that $g_i(\bar{a}) = c_i$ for $i = 1, \ldots, n$ and $h(\bar{c}) = b$. Then

$$PA \vdash \bigwedge_{i=1}^{n} \phi_{g_i}(\bar{a}, \bar{c}_i) \land \phi_h(\bar{c}, \bar{b})$$

so $PA \vdash \phi_f(\bar{a}, \bar{b})$. On the other hand suppose $a_1, \ldots, a_m, b \in \mathbb{N}$ and $PA \vdash \phi_f(\bar{a}, \bar{b})$. Then $\mathbb{N} \models \phi_f(\bar{a}, \bar{b})$. Thus there are $c_1, \ldots, c_n \in \mathbb{N}$ such that

$$\mathbb{N} \models \phi_{g_i}(\bar{a}, \bar{c}_i) \land \phi_h(\bar{c}, \bar{b}).$$
Since \( PA \) proves that \( \phi_{g_i} \) and \( \phi_h \) define graphs of functions which agree on \( \mathbb{N}^m \) and \( \mathbb{N}^n \) with \( g_i \) and \( h \) respectively, \( g_i(\bar{a}) = c_i \) and \( h(\bar{c}) = b \) thus \( f(\bar{a}) = b \).

Since \( PA \) proves that \( \phi_{g_i} \), for \( i = 1, \ldots, n \) and \( \phi_h \) define the graphs of a functions, \( PA \) proves that \( \phi_f \) defines the graph of a function.

- Suppose \( g : \mathbb{N}^{m-1} \rightarrow \mathbb{N} \) and \( h : \mathbb{N}^{m+1} \rightarrow \mathbb{N} \) are primitive recursive \( f(0, \bar{x}) = g(\bar{x}) \) and \( f(n+1, \bar{x}) = h(n, \bar{x}, f(n, \bar{x})) \). Let \( \phi_g \) represent \( g \) and \( \phi_h \) represent \( h \).

Let \( \phi_f(z, \bar{x}, y) \) be the formula asserting "there is \( \sigma \) coding a sequence of length \( z + 1 \), say \( \sigma = (u_0, \ldots, u_z) \) where \( \phi_g(i, \bar{x}, u_0) \) and \( \phi_h(i, \bar{x}, u_i, u_{i+1}) \) for \( i < z + 1 \) and \( y = u_z \)."

To make this precise we use the beta-function. Let \( \phi_f(z, \bar{x}, y) \) be the formula

\[
\exists u, u \left[ (\forall w \leq u \beta(u, v, z + 1)_0 = w \rightarrow \phi_g(\bar{x}, w)) \land (\forall i \leq z \forall w_1, w_2 \leq u \ (\beta(u, v, z + 1)_i = w_1 \land \beta(u, v, z + 1)_{i+1} = w_2) \rightarrow \phi_h(i, \bar{x}, w_1, w_2) \right].
\]

Using the facts above it is easy to see that \( \phi_f \) has the desired interpretation.

**Exercise:** Show that \( \phi_f \) represents \( f \).

This concludes the proof of the Representation Lemma.

We can get a much sharper version of the representation lemma.

**Exercise:** We say that a formula in the language of arithmetic is a \( \Delta_0 \) formula if it is in the smallest collection of formulas containing the atomic formulas and closed under \( \land, \lor, \neg \) and bounded quantification \( \forall x < t \) and \( \exists x < t \) where \( t \) is a term. For example \( \forall x < y^2 \exists z < 2xy \ z = y - x \) is a \( \Delta_0 \) formula.

We say that \( \psi(\bar{x}) \) is a \( \Sigma_1 \)-formula if it is of the form \( \exists \bar{y} \phi(\bar{x}, \bar{y}) \) where \( \phi \) is \( \Delta_0 \). We say that a formula \( \theta \) is \( \Sigma^1_{PA} \) if \( PA \vdash \theta \rightarrow \psi \) for some \( \Sigma_1 \)-formula \( \Psi \).

i) If \( \phi_0 \) and \( \phi_1 \) are \( \Sigma^1_{PA} \)-formulas show that there \( \phi \land \psi \) and \( \phi \lor \psi \) are \( \Sigma^1_{PA} \).

ii) Suppose \( \phi(\bar{x}, y) \) is \( \Sigma^1_{PA} \) then so are \( \exists \bar{y} \phi(\bar{x}, \bar{y}) \) and \( \forall \bar{y} < t(\bar{x}) \phi(\bar{x}, y) \) for any term \( t \).

iii) Show that any primitive recursive function is represented by a \( \Sigma^1_{PA} \)-formula.

We can say a bit more. Suppose \( P(\bar{x}) \) is a primitive recursive predicate. Then there is a formula \( \psi_P(\bar{x}) \) such that \( P(\bar{n}) \) if and only if \( PA \vdash \psi_P(\bar{n}) \) for all \( \bar{n} \in \mathbb{N} \). [Let \( \Psi_P(\bar{x}) \) be \( \phi_f(\bar{x}, 1) \) where \( f \) is the primitive recursive characteristic function for \( P \).] The same is true for any recursively enumerable set.

**Corollary 9.7.** Let \( A \subseteq \mathbb{N} \) be recursively enumerable. There is a formula \( \psi_A(x) \) such that \( n \in A \) if and only if \( PA \vdash \psi_A(n) \) for all \( n \in \mathbb{N} \).

**Proof.**

Let \( A = W_e \). Then \( x \in A \) if and only if \( \exists s T(e, x, s) \) where \( T \) is Kleene’s \( T \)-predicate. Let \( \psi_T(u, v, w) \) be a formula representing the primitive recursive predicate \( T \). Let \( \psi_A(x) \) be the formula

\[
\exists y \ \psi_T(\hat{e}, x, y).
\]

If \( n \in A \), then there is an \( s \in \mathbb{N} \) such that \( T(e, n, s) \). Since \( \psi_T \) represents \( T \), \( PA \vdash \psi_T(e, n, s) \). But then \( PA \vdash \exists y \ \psi_T(\hat{e}, \hat{n}, y) \).
On the other hand if PA ⊨ ∃y ψ_T(⌜e, n, y⌝), then N ⊨ ∃y ψ_T(⌜e, n, y⌝). Choose s ∈ N such that N ⊨ ψ_T(⌜e, n, s⌝). But then T(e, n, s) holds and n ∈ A.

We conclude this section with a stronger version of the first incompleteness theorem. We will replace the assumption that N ⊨ T by the weaker assumption that T ⊨ PA is consistent.

Let P(T) = {φ : T ⊨ φ} and let R(T) = {φ : T ⊨ ¬φ}. Intuitively P(T) are the sentences provable from T and R(T) are the sentences refutable from T.

**Theorem 9.8.** The sets P(PA) and R(PA) are recursively inseparable.

**Proof.**
Since PA is consistent P(PA) ∩ R(PA) = ∅ and since PA is recursively axiomatized they are recursively enumerable.

For i = 0, 1 let A_i(e, x, s) be the primitive recursive predicate asserting “the computation of φ_e(x) halts in at most s steps with output i”. Let ψ_i(e, x, s) represent A_i.

Let θ_0(x) be the formula

$$\exists y \psi_0(x, x, s) \land \forall z < y \neg \psi_1(x, x, z)$$

and let θ_1(x) be the formula

$$\exists y \psi_1(x, x, s) \land \forall z \leq y \neg \psi_0(x, x, z)$$

**claim i)** PA ⊨ ∀x ¬(φ(x) ∧ ψ(x))

**ii)** PA ⊨ ∀x [(∃y(ψ_0(x, x, y) ∨ ψ_1(x, x, y))] → (θ_0(x) ∨ θ_1(x))].

For any x, if there is a y such that one of ψ_0(x, x, y) or ψ_1(x, x, y) holds, then, by induction, there is y_0 such that one of the ψ_i(x, x, y_0) holds and ¬ψ_0(x, x, z) and ¬ψ(x, x, z) for z < y_0. If ψ_0(x, x, y_0) holds, then θ_0(x); otherwise θ_1(x).

Let A = {e : φ_e(e) = 0} and B = {e : φ_e(e) = 1} be the recursively inseparable sets of 8.4.

Suppose C is a recursive set of sentences such that P(PA) ⊆ C and R(PA) ⊆ C. Let D = {e : the sentence θ_0(⌜e⌝) ∈ C}. Clearly D is recursive. We claim that D separates A and B.

Let e ∈ A. Then there is an s ∈ N such that A_0(e, e, s). Thus PA ⊨ ψ_0(⌜e, hate, s⌝) and for all t < s PA ⊨ ¬ψ_1(⌜e, hate, s⌝) Since

$$PA \vdash t < s \rightarrow \bigwedge_{i=0}^{s-1} t = i,$$

PA ⊨ ψ_0(⌜e, hate, s⌝) ∧ ∀z < s ¬ψ_1(⌜e, hate, s⌝). Thus PA ⊨ θ_0(⌜e⌝). So θ(⌜e⌝) ∈ R(PA) ⊆ C and e ∈ D.

Similarly if e ∈ B, then PA ⊨ θ_1(⌜e⌝). Thus PA ⊨ ¬θ_0(⌜e⌝). Thus θ_0(⌜e⌝) ∈ R(PA) ⊆ ¬C and e ∈ ¬D. Since D separates A and B, D is not recursive, a contradiction.
Corollary 9.9. (Rosser’s Incompleteness Theorem) Suppose $T \supseteq PA$ is a consistent recursively axiomatized theory. Then $P(T)$ and $R(T)$ are recursively inseparable. In particular $T$ is incomplete.

Proof.

Since $T$ is consistent and recursively axiomatized, $P(T)$ and $R(T)$ are disjoint recursively enumerable sets. Since $T \supseteq PA$, $P(T) \supseteq P(PA)$ and $R(T) \supseteq R(PA)$. If we could recursively separate $P(T)$ and $R(T)$ we would also separate $P(PA)$ and $R(PA)$ contradicting 9.8.

If $T$ were complete, then $P(T)$ would itself be recursive (by 4.8) contradicting 9.8.
§10. Gödel’s Incompleteness Theorem II: Arithmetization of Syntax:

In §9 we gave a proof of the first incompleteness theorem based on basic recursion theoretic ideas. In this section we give a second proof which follows Gödel more closely. We will also sketch the ideas behind the proof of the second incompleteness theorem. The new idea of this section is the idea of Gödel codes for formulas.

We will assign a number $[\phi]$ to each formula $\phi$. We call $[\phi]$ the Gödel code for $\phi$. Gödel coding allows us to talk about properties of formulas in the language of arithmetic. Gödel showed that there are amazing possibilities for self reference. In particular he proved the following striking lemma.

Lemma 10.1. (Diagonalization Lemma) Let $\phi(v)$ be a formula in the language of arithmetic with one free variable $v$. There is a sentence $\psi$ such that

$$PA \vdash \psi \iff \phi([\psi]).$$

Intuitively the sentence $\psi$ says “My code has property $\phi$”. Strictly speaking we should write $PA \vdash \psi \iff \phi([\phi])$ but we will drop the $\neg$ when no confusion arises.

We will begin shortly the work needed to prove the diagonalization lemma and to deduce the incompleteness theorem from it, but first let us deduce one simple and important corollary.

A formula $\Gamma(v)$ is called a truth definition if and only if

$$N \models \psi \text{ if and only if } N \models \Gamma([\psi]).$$

for all sentences $\psi$

Corollary 10.2. (Tarski’s Undeﬁnability of Truth) There are no truth definitions.

Proof.

Suppose $\Gamma(v)$ is a truth definition. Apply the diagonalization lemma to $\neg \Gamma$ to obtain a sentence $\psi$ such that $PA \vdash \psi \iff \neg \Gamma([\psi])$. Clearly $\psi$ shows that $\Gamma$ is not a truth definition.

We now begin the mechanics of coding. We fix a primitive recursive method of coding finite sequences. We let $\langle a_1, \ldots, a_m \rangle$ be the code for the sequence $(a_1, \ldots, a_m)$. We choose the coding so that:

i) every natural number codes a sequence,
ii) $n \mapsto l(n)$ is primitive recursive, where $l(n)$ is the length of the sequence coded by $n$, and
iii) $(n, i) \mapsto (n)_i$ is primitive recursive, where $(n)_i$ is the $i^{th}$-element of the sequence coded by $n$ if $i \leq l(n)$ and $(n)_i = 0$ if $i > l(n)$.

For example we could use the coding $\tau$ described in §6.
Let us assume that our language is $\mathcal{L} = \{+, \cdot, <, 0, 1\}$ and that we use only the connectives $\land$ and $\neg$, the quantifier $\exists$ and variables $v_0, v_1, \ldots$. We assign each symbol a code as follows.

$$[0] = \langle 0, 0 \rangle \quad [1] = \langle 0, 1 \rangle \quad [v_i] = \langle 1, i \rangle$$
$$[+] = \langle 2, 0 \rangle \quad [.] = \langle 2, 1 \rangle \quad [\langle] = \langle 3, 0 \rangle \quad [-] = \langle 4, 1 \rangle$$
$$[\exists] = \langle 5, 0 \rangle$$

We inductively define coding of terms as follows. If $t_1$ and $t_2$ are terms then

$$[t_1 + t_2] = \langle [+] , [t_1], [t_2] \rangle \quad \text{and} \quad [t_1 \cdot t_2] = \langle [\cdot] , [t_1], [t_2] \rangle .$$

If $t_1$ and $t_2$ are terms, we code atomic formulas as follows.

$$[t_1 = t_2] = \langle [=] , [t_1], [t_2] \rangle \quad \text{and} \quad [t_1 < t_2] = \langle [<] , [t_1], [t_2] \rangle .$$

Finally if $\phi$ and $\psi$ are formulas then

$$[-\phi] = \langle [\neg] , [\phi] \rangle ,$$
$$[\phi \land \psi] = \langle [\land] , [\phi], [\psi] \rangle \quad \text{and} \quad [\exists v_i \phi] = \langle [\land] , [v_i], [\phi] \rangle .$$

We will see that all basic syntactic properties of formulas are primitive recursive. It is easy to see for example that the maps $[\phi] \mapsto [-\phi]$ and $( [\phi], [\psi] ) \mapsto [\phi \land \psi]$ are primitive recursive.

**Lemma 10.3.** The predicates “$n$ codes a term” and “$n$ codes a formula” are primitive recursive.

**Proof.**

Let

$$T(x) = \begin{cases} 1 & x = [0] \text{ or } x = [1] \\ 1 & l(x) = 3, (x)_1 = \langle 2, 0 \rangle \text{ or } \langle 2, 1 \rangle, \ T((x)_2) = 1 \text{ and } T((x)_3) = 1 . \\ 0 & \text{otherwise} \end{cases}$$

Clearly $T$ is primitive recursive and $T(n) = 1$ if and only if $n$ codes a term. Let

$$F(x) = \begin{cases} 1 & l(x) = 3, (x)_1 = [\langle] \text{ or } [\langle] , T((x)_1) = 1 \text{ and } T((x)_2) = 1 \\ 1 & l(x) = 2, (x)_1 = [\neg] \text{ and } F((x)_2) = 1 . \\ 1 & l(x) = 3, (x)_1 = [\land] \text{ and } F((x)_2) = F((x)_3) = 1 . \\ 1 & l(x) = 3, (x)_1 = [\exists] , \exists i < x \ (x)_2 = \langle 1, i \rangle \text{ and } F((x)_3) = 1 . \\ 0 & \text{otherwise} \end{cases} .$$

Then $F$ is primitive recursive and $F(n) = 1$ if and only if $n$ is the code for a formula.
The next lemmas will be the key to proving the diagonalization lemma.

**Lemma 10.4.** There is a primitive recursive function \( s \) such that if \( t \) is a term and \( i, y \in \mathbb{N} \), then \( s([t], i, y) \) is the code for the term obtained by replacing all occurrences of \( v_i \) in \( t \) by the term \( \hat{y} \) (where \( \hat{y} \) is the term \( 1 + \ldots + 1 \)).

**Proof.** We define \( s \) by:

\[
 s(x, i, y) = \begin{cases} 
 x & x = [0], x = [1] \text{ or } x = [v_j] \text{ where } i \neq j \\
 [\hat{y}] & x = [v_i] \\
 \langle+, s(t_1, i, y), s(t_2, i, y)\rangle & x = \langle+, t_1, t_2\rangle \\
 \langle\cdot, s(t_1, i, y), s(t_2, i, y)\rangle & x = \langle\cdot, t_1, t_2\rangle \\
 0 & \text{otherwise}
\end{cases}
\]

Clearly \( s \) is primitive recursive and \( s \) is the desired function.

**Lemma 10.5.** There is a primitive recursive function \( \text{sub} \) such that \( \text{sub}([\phi], i, y) = [\psi] \) where \( \psi \) is the formula obtained by substituting \( \hat{y} \) for each free occurrence of \( v_i \) in \( \phi \).

**Proof.**

We may define \( \text{sub} \) by

\[
 \text{sub}(x, i, y) = \begin{cases} 
 \langle[=], s(t_1, i, y), s(t_2, i, y)\rangle & x = \langle[=], t_1, t_2\rangle \\
 \langle[<], s(t_1, i, y), s(t_2, i, y)\rangle & x = \langle[<], t_1, t_2\rangle \\
 \langle[\neg], \text{sub}([\phi], i, y)\rangle & x = \langle[\neg], [\phi]\rangle \\
 \langle[\land], \text{sub}([\phi], i, y), \text{sub}([\psi], i, y)\rangle & x = \langle[\land], [\phi], [\psi]\rangle \\
 \langle[\exists], [v_j], \text{sub}([\phi], i, y)\rangle & x = \langle[\exists], [v_j], [\phi]\rangle \text{ and } i \neq j \\
 \langle[\exists], [v_i], [\phi]\rangle & x = \langle[\exists], [v_i], [\phi]\rangle \\
 0 & \text{otherwise}.
\end{cases}
\]

We are now ready to prove the diagonalization lemma 10.1. Let \( \phi(v_0) \) be an \( \mathcal{L} \)-formula with one free variable \( v_0 \). Let \( S(x, y, z, w) \) be an \( \mathcal{L} \)-formula representing the primitive function \( \text{sub} \).

Let \( \theta(v_0) = \exists y \ (S(v_0, 0, v_0, y) \land \phi(y)) \) That is, \( \theta(v_0) \) asserts \( \phi(\text{sub}(v_0, 0, v_0)) \). Let \( m = [\theta(v_0)] \) and let \( \psi = \theta(m) \).

Then

\[
 PA \vdash \psi \leftrightarrow \theta(m) \\
 \leftrightarrow \exists y \ S(m, 0, m, y) \land \phi(y) \\
 \leftrightarrow \exists y \ S([\theta(v_0)], 0, m, y) \land \phi(y) \\
 \leftrightarrow \exists y \ y = [\theta(m)] \land \phi(y) \\
 \leftrightarrow \phi([\theta(m)]) \\
 \leftrightarrow \phi([\psi])
\]

as desired.
Our coding will be slightly easier if we assume that the theory $T$ has a primitive recursive axiomatization. The next lemma shows that this is no loss of generality.

**Lemma 10.6.** (Craig’s trick) Suppose $T$ is a recursively axiomatized $L$-theory. Then there is a primitive recursively axiomatized $L$-theory $T^*$ such that $T$ and $T^*$ have the same consequences (i.e. $T \vdash \phi \leftrightarrow T^* \vdash \phi$ for any $L$-sentence $\phi$).

**Proof.**

Suppose $W_e = \{[\phi] : \phi \in T\}$. Let

$$T^* = \left\{ \phi \land \ldots \land \phi : [\phi] \in W_e^s \right\},$$

It is easy to see that $T$ and $T^*$ have the same logical consequences. On the other hand, since “$x \in W_e^s$” is a primitive recursive predicate, $\{[\psi] : \psi \in T^*\}$ is primitive recursive.

Note that in fact Craig’s trick works just as well for recursively enumerable theories $T$.

**Lemma 10.7.** Let $T$ be a primitive recursive $L$-theory. Let $Prov_T(x, y)$ be the predicate “$x$ is a proof from $T$ if the formula with Gödel code $y$”. The predicate $Prov_T$ is primitive recursive.

**Proof.**

Basicly, $Prov_T(x, y)$ if and only if $\forall i \leq l(x) \ (x)_i \in T$ or $(x)_i$ follows from previous $(x)_j$ be an inference rule.

This can be coded in a primitive recursive way. We leave the details to the reader.

Let $\psi_{Prov_T}(x, y)$ be an $L$-formula representing $Prov_T$ in PA and let

$$Pr_T(y) \iff \exists x \psi_{Prov_T}(x, y)$$

Using $Pr_T$ we can give Gödel’s proof of the first incompleteness theorem. Let $T$ be a consistent primitive recursive theory extending $PA$. By the diagonalization lemma there is a sentence $\phi$ such that $PA \vdash \phi \leftrightarrow \neg Pr_T(\phi)$. We call $\phi$ the Gödel sentence for $T$.

**Theorem 10.8** (First Incompleteness theorem) Let $\phi$ be the Gödel sentence for $T$. Then $T \not\vdash \phi$. Moreover if $N \models T$, then $T \not\vdash \neg \phi$.

**Proof.**

If $T \vdash \phi$, then there is such an $n \in N$ such that $Prov_T(n, [\phi])$. But then $PA \vdash \psi_{Prov_T}(n, [\phi])$ and $PA \vdash Pr_T([\phi])$ and $PA \vdash \neg \phi$. Thus $T \vdash \neg \phi$, contradicting the consistency of $T$. Hence $T \not\vdash \phi$.

If $N \not\models \phi$, then $N \models Pr_T([\phi])$ and hence there is $m \in N$ such that $N \models \psi_{Prov_T}(m, [\phi])$ and $m$ really is the code for a proof of $\phi$ from $T$. But then $T \vdash \phi$ and $N \models \phi$ a contradiction, thus $N \not\models \phi$. Hence if $N \models T$, then $T \not\vdash \neg \phi$. 


A slightly different diagonalization gives a different proof of the incompleteness result from corollary 9.9.

**Theorem 10.9.** Let $T$ be a recursively axiomatized consistent extension of $PA$, then $T$ is incomplete.

**Proof.**

Let $\theta(x,y)$ be an $L$-formula representing the primitive recursive relation “$x$ and $y$ are Gödel codes for formulas and $x$ codes the negation of the formula coded by $y$”.

Let $Pr_T^*(v)$ be the formula

$$\exists y (Prov_T(y,v) \land \forall z (\theta(z,v) \rightarrow \forall x < y \neg Prov_T(x,z))).$$

Thus $Pr_T^*([\phi])$ asserts that there is $x$ coding a proof of $\phi$ and no $y < x$ codes a proof of $\neg \phi$.

By the diagonalization lemma there is a sentence $\phi$ such that

$$PA \vdash \phi \iff \neg Pr_T^*([\phi]).$$

We call $\phi$ a Rosser sentence.

Suppose $T \vdash \phi$. Then there is a natural number $n$ coding a proof of $\phi$ and since $T$ is consistent if $m < n$, then $m$ does not code a proof of $\neg \phi$. But then if $M \models T$, then $M \models Pr_T^*([\phi])$ and $M \not\models \neg \phi$ a contradiction. Thus $T \not\models \phi$.

Suppose $T \vdash \neg \phi$. Then there is a natural number $n$ coding a proof of $\neg \phi$ and if $m < n$, then $m$ does not code a proof of $\phi$. Thus if $M \models T$, then $M \models \neg Pr_T^*([\phi])$, so $M \models \phi$, a contradiction. Thus $T \not\models \neg \phi$.

The next lemma summarizes the facts about provability that one must verify in $PA$ to prove the second incompleteness theorem.

**Lemma 10.10.** Let $T \supseteq PA$ be a primitive recursive theory and let $\phi$ and $\psi$ be $L$-sentences. Then the following derivability conditions hold

D1. If $T \vdash \phi$, then $PA \vdash Pr_T([\phi])$.

D2. If $PA \vdash Pr_T([\phi])$, then $PA \vdash Pr_T([Pr_T([\phi])])$.

D3. $PA \vdash (Pr_T([\phi]) \land Pr_T([\phi \rightarrow \psi]) \rightarrow Pr_T([\psi])$.

D4. $PA \vdash (Pr_T([\phi]) \land Pr_T([\psi]) \rightarrow Pr_T([\phi])$.

D5. $PA \vdash Pr_{T+\psi}([\phi]) \iff Pr_T([\psi \rightarrow \phi])$.

**Proof.**

D1 is easy. If $n$ codes a proof of $\phi$, then $PA \vdash Prov_T(n,[\phi])$, so $PA \vdash Pr_T([\phi])$.

D3, D4 and D5 are proved by constructing a primitive recursive function which takes the two given proofs and produces the desired proof. D2 requires more work. See Boolos and Jeffries, *Computability and Logic* for details.
**Theorem 10.11** (Second Incompleteness Theorem) Let \( T \) be a consistent recursively axiomatized theory such that \( T \supseteq PA \). Let \( Con(T) \) be the sentence \( \neg Pr_T([0 = 1]) \). Then \( T \nvdash Con(T) \).

**Proof.**

Let \( \phi \) be the Gödel sentence such that \( PA \vdash \phi \iff \neg Pr_T([\phi]). \)

We will show that \( PA \vdash \phi \iff Con(T) \). Then, by 10.8, \( PA \nvdash \phi \), so \( PA \nvdash Con(T) \).

Since we can derive anything from a contradiction, \( T \vdash 0 = 1 \iff \phi \). Thus by D1,

\[
PA \vdash Pr_T([0 = 1 \iff \phi]).
\]

Thus by D3, \( PA \vdash \neg Con(T) \rightarrow Pr_T([\phi]). \) By choice of \( \phi \), we have

\[
PA \vdash \neg Con(T) \rightarrow \neg \phi
\]

and taking the contrapositive

\[
PA \vdash \phi \rightarrow Con(T).
\]

On the other hand, by D2

\[
PA \vdash Pr_T([\phi]) \rightarrow Pr_T([Pr_T([\phi])]).
\]

By choice of \( \phi \), \( PA \vdash Pr_T(\phi) \rightarrow \neg \phi \). Thus by D1 and D3

\[
PA \vdash Pr_T([Pr_T([\phi])]) \rightarrow Pr_T([\neg \phi]).
\]

Thus

\[
PA \vdash Pr_T([\phi]) \rightarrow Pr_T([\neg \phi]).
\]

Using D4 we see that

\[
PA \vdash Pr_T([\phi]) \rightarrow Pr_T([\phi \land \neg \phi]).
\]

But then by D1 and D2

\[
PA \vdash Pr_T([\phi]) \rightarrow Pr_T([0 = 1]).
\]

So \( PA \vdash Con(T) \rightarrow \phi \), as desired.

By the diagonalization lemma there are sentences \( \phi \) such that \( PA \vdash \phi \iff Pr_T(\phi) \). Henkin asked if such a sentence is provable? The following result shows that it is.
**Corollary 10.12.** (Löb’s Theorem) Let $T$ be a consistent recursively axiomatized theory extending $PA$ and let $\phi$ be any sentence. Then

$$ T \vdash Pr_T([\phi]) \rightarrow \phi \iff T \vdash \phi. $$

**Proof.**

$(\Leftarrow)$ This is clear since if $T \vdash \phi$, then $T \vdash \psi \rightarrow \phi$ for any sentence $\psi$.

$(\Rightarrow)$. Suppose $T \nvDash \phi$. Then $T + \neg \phi$ is consistent and by the second incompleteness theorem

$$ T + \neg \phi \nvDash Con(T + \neg \phi). $$

By D5,

$$ T + \neg \phi \nvDash \neg Pr_T([\neg \phi \rightarrow 0 = 1]). $$

Since $T \vdash (\neg \phi \rightarrow 0 = 1) \rightarrow \phi$, by D1,

$$ T \vdash Pr_T([\neg \phi \rightarrow 0 = 1] \rightarrow \phi). $$

Thus, by D3,

$$ T \vdash Pr_T([\neg \phi \rightarrow 0 = 1]) \rightarrow Pr_T(\phi). $$

Thus, $T + \neg \phi \nvDash \neg Pr_T(\phi)$. So $T \nvDash Pr_T(\phi) \rightarrow \phi$, as desired.