cse371/ math371
LOGIC

Professor Anita Wasilewska

LECTURE 11

Chapter 11
Formal Theories and Gödel Theorems

PART 1: **Formal Theories:** Definition and Examples

PART 2: **PA:** Formal Theory of **Natural Numbers**

PART 3: Consistency, Completeness, **Gödel Theorems**

PART 4: **Proof** of the Gödel **Incompleteness Theorems**

Chapter 11
Formal Theories and Gödel Theorems

PART 1: **Formal Theories:** Definition and Examples

Inroduction

Formal theories play crucial role in mathematics

They were historically defined for classical predicate and also for other first and higher order logics, classical and non-classical

The idea of formalism in mathematics, which resulted in the **concept** of formal theories, or formalized theories, as they are also called

The concept of **formal theories** was developed in connection with the Hilbert Program

Introduction

Hilbert Program

One of the main objectives of the Hilbert program was to construct a **formal theory** that would cover the whole mathematics and to prove its **consistency** by employing the simplest of logical means

We say that a formal theory is **consistent** if no formal proof can be carried in that theory for a formula A and at the same time for its negation $\neg A$

This part of the program is called the **Consistency** Program .

Introduction

In 1930, while still in his twenties Kurt Gödel made a historic announcement:

Hilbert **Consistency Program** *could not be carried out*

He justified his claim by proving his **Inconsistency Theorem**

The Gödel **Inconsistency Theorem** is called also Gödel **Second Incompleteness Theorem**

Introduction

Gödel Inconsistency Theorem

Roughly speaking the theorem states that

a **proof** *of the* **consistency** *of every* **formal theory** *that contains arithmetic of natural numbers* **can be** *carried out* **only in** *mathematical theory which is more comprehensive than the one whose* **consistency** *is to be proved*

In particular,

Gödel **Inconsistency Theorem** states that

a **proof** *of the* **consistency** *of formal (elementary, first order) arithmetic can be carried out* **only** *in mathematical theory which* **contains** *the whole arithmetic and also other theorems that* **do not** *belong to arithmetic*

It applies to a **formal theory** that would cover the whole mathematics because it would obviously contain the arithmetic of natural numbers

Hence the **Hilbert Consistency Program** fails

Introduction

Gödel's result concerning the **proofs** of the **consistency** of **formal** mathematical theories has had a decisive impact on research in properties of **formal theories**

Instead of looking for **direct proofs** of **inconsistency** of mathematical theories, mathematicians concentrated largely on **relative proofs** that demonstrate that a theory under consideration is **consistent** if a certain **other theory**, for example a **formal** theory of natural numbers, is **consistent**

Introduction

All those **relative proofs** are rooted in a deep conviction that even though it cannot be proved that the theory of natural numbers is **free** of inconsistencies, it is **consistent**

This conviction is **confirmed** by centuries of development of mathematics and experiences of mathematicians

Complete and **Incomplete Theories**

We say that formal theory is called **complete**

if for every sentence (formula without free variables) of

the language of that theory **there is** a formal proof of it

or of its negation

A formal theory is **incomplete** if there is a sentence A of the language of that theory, such that **neither** A **nor** $\neg A$

are **provable** in it

Such sentences are called **undecidable** or **independent**

of the **theory**

Introduction

It might seem that one should be able to **formalize** a formal theory of natural numbers in a way to make it **complete** i.e. **free** of undecidable (independent) sentences

Gödel proved that **it is not** the case in the following

Gödel **Incompleteness Theorem**

Every ***consistent*** *formal theory which contains the arithmetic of natural numbers is* ***incomplete***

Introduction

The Gödel **Inconsistency Theorem** follows from the Gödel **Incompleteness Theorem**

This is why the Incompleteness and Inconsistency **Theorems** are now called

Gödel **First Incompleteness Theorem** and
Gödel **Second Incompleteness Theorem**, respectively

Introduction

The **third part** of the **Hilbert Program** posed and was concerned with the problem of decidability of formal mathematical theories

A formal theory is called **decidable** if there is a method of determining, in a finite number of steps, whether any given formula in that theory **is** its theorem or **not**

Most of mathematical theories are **undecidable**

Gödel proved in 1931 that the arithmetic of natural numbers is **undecidable**

Formal Theories: Definition and Examples

Formal Theories: Definition and Examples

We define here a notion of a **formal theory** based on a predicate (first order) language

Formal theories are also routinely called first order or elementary theories, or formal axiomatic theories, or **theories**, when it is clear from the context that they are formal theories

We will often use the term **theory** for simplicity.

Formal Theories: Definition and Examples

We consider here only **formal theories** based on a complete classical Hilbert style proof system

$$H = (\mathcal{L}, \mathcal{F}, LA, \mathcal{R})$$

with a predicate (first order) language

$$\mathcal{L} = \mathcal{L}_{\{\neg, \cap, \cup, \Rightarrow\}}(\mathbf{P}, \mathbf{F}, \mathbf{C})$$

where the sets **P, F, C** are infinitely enumerable.

A formal theory based on H is a proof system obtained from H by **adding** a special set SA of axioms to it, called the set of specific axioms

Formal Theories: Definition and Examples

Let SA be a certain set of formulas of $\mathcal{L}$ of H, such that

$$SA \subseteq \mathcal{F} \quad \text{and} \quad \mathbf{T}_{\mathcal{L}} \cap SA = \emptyset$$

where $\mathbf{T}_{\mathcal{L}}$ denotes the set of formulas of $\mathcal{L}$ that are classical **tautologies**

We call the set SA a set of specific axioms of a formal **theory** based on H

Formal Theories: Definition and Examples

The **specific axioms** are characteristic descriptions of the universe of the formal theory

The **specific axioms** SA are to be true only in a **certain** structure as opposed to logical axioms LA that are true in **all** structures i.e. that are tautologies

Formal Theories: Definition and Examples

Language $\mathcal{L}_{SA}$

Given a proof system $H = (\mathcal{L}, \mathcal{F}, LA, \mathcal{R})$ and a non-empty set SA of specific axioms

We define a language

$$\mathcal{L}_{SA} \subseteq \mathcal{L}$$

determined by the specific axioms SA by **restricting** the sets **P, F, C** of predicate, functional, and constant symbols of $\mathcal{L}$ to predicate, functional, and constant symbols appearing in the set SA of specific axioms

Both languages $\mathcal{L}_{SA}$ and $\mathcal{L}$ share the same set of propositional **connectives**

Formal Theory

Given a proof system $H = (\mathcal{L}, \mathcal{F}, LA, \mathcal{R})$ and a non-empty set SA of specific axioms

A proof system $T = (\mathcal{L}, \mathcal{F}, LA, SA, \mathcal{R})$ is called a **formal theory** based on H, with its set SA of specific axioms

The language

$$\mathcal{L}_{SA} \subseteq \mathcal{L}$$

determined by the set SA is called the language of the **formal theory** T

Formal Theories: Definition and Examples

Given a theory $T = (\mathcal{L}, \mathcal{F}, LA, SA, \mathcal{R})$

We denote by

$$\mathcal{F}_{SA}$$

the set of formulas of the **language** $\mathcal{L}_{SA}$ of the theory T

We denote by $\mathbf{T}$ the set all **provable** formulas in T, i.e.

$$\mathbf{T} = \{B \in \mathcal{F}_{SA} : \quad SA \vdash B\}$$

We also write $\vdash_T B$ to denote that $B \in \mathbf{T}$

LE- Logic with Equality

Definition

A proof system

$$H = (\mathcal{L}, \mathcal{F}, \; LA, \; \mathcal{R})$$

is called a **Logic with Equality** LE if and only if the language $\mathcal{L}$ has **as one of its predicates**, a two argument

predicate P which we denote by $=$, and the following axioms are provable in H

LE- Logic with Equality

Equality Axioms LE

For any any free variable or constant of $\mathcal{L}$, i.e for any $u, w, u_i, w_i \in (VAR \cup \mathbf{C})$, and any $R \in \mathbf{P}$, and $t \in \mathbf{T}_{\mathcal{L}}$, where $\mathbf{T}_{\mathcal{L}}$ is set of all **terms** of $\mathcal{L}$, the following properties hold

E1 $u = u$

E2 $(u = w \Rightarrow w = u)$

E3 $((u_1 = u_2 \cap u_2 = u_3) \Rightarrow u_1 = u_3)$

LE- Logic with Equality

E4
$$((u_1 = w_1 \cap ... \cap u_n = w_n) \Rightarrow (R(u_1, ..., u_n) \Rightarrow R(w_1, ..., w_n)))$$
E5
$$((u_1 = w_1 \cap ... \cap u_n = w_n) \Rightarrow (t(u_1, ..., u_n) \Rightarrow t(w_1, ..., w_n)))$$

Directly from above definition we have the following

Fact

The Hilbert style proof system **H** defined in chapter 9 is a logic with equality with the set of specific axioms $SA = \emptyset$

Formal Theories: Some Examples

Formal Theories: Some Examples

Formal theories are abstract models of real **mathematical theories** we develop using laws of logic

Hence the theories we present here are based on a **complete** proof system H for classical predicate logic with a language

$$\mathcal{L} = (\mathcal{L}_{\{\neg,\cap,\cup,\Rightarrow\}}(\mathbf{P}, \mathbf{F}, \mathbf{C})$$

The classical, first order (predicate) formal theories are also called first order **elementary theories**

Formal Theories: Some Examples

$T1$. **Theory of Equality**

Language

$$\mathcal{L}_{T1} = \mathcal{L}_{\{\neg,\Rightarrow,\cup,\cap\}}(\mathbf{P} = \{P\},\ \mathbf{F} = \emptyset,\ \mathbf{C} = \emptyset)$$

where # P = 2, i.e. P is a two argument predicate

The **intended** interpretation of P is **equality**, so we use the equality symbol $=$ instead of P

We write $x = y$ instead $=(x, y)$

We write the **language** of $T1$ as

$$\mathcal{L}_{T1} = \mathcal{L}_{\{\neg,\Rightarrow,\cup,\cap\}}(\{=\},\ \emptyset,\ \emptyset)$$

Formal Theories: Some Examples

$T1.$ **Specific Axioms**

e1 $x = x$

e2 $(x = y \Rightarrow y = x)$

e3 $(x = y \Rightarrow (y = z \Rightarrow x = z))$

for any $x, y, z \in VAR$

Observation

We have chosen to write the $T1.$ specific axioms as **open** formulas. Sometimes it is more convenient to write them as **closed** formulas (sentences)

In this case new axioms will be **closures** of axioms that were **open** formulas

Formal Theories: Some Examples

T2. **Theory of Equality (2)**

We adopt a **closure** of the axioms $e1, e2, e3$, i.e. the following new set of axioms.

Specific Axioms

(e1) $\forall x(x = x)$

(e2) $\forall x \forall y(x = y \Rightarrow y = x)$

(e3) $\forall x \forall y \forall z(x = y \Rightarrow (y = z \Rightarrow x = z))$

Formal Theories: Some Examples

*T*3. **Theory of Partial Order**

Partial order relation is also called **order** relation.

Language

$$\mathcal{L}_{T3} = \mathcal{L}_{\{\neg, \Rightarrow, \cup, \cap\}}(\mathbf{P} = \{P, Q\},\ \mathbf{F} = \emptyset,\ \mathbf{C} = \emptyset)$$

where P is a two argument predicate

The **intended** interpretation of P is **equality**, so we use the equality symbol $=$ instead of P

Q is a two argument predicate

The **intended** interpretation of Q is **partial order**

We use the order symbol $\leq$ instead of Q and write $x \leq y$ instead $\leq (x, y)$

We write the language of $T3$ as

$$\mathcal{L}_{T3} = \mathcal{L}_{\{\neg, \Rightarrow, \cup, \cap\}}(\{=,\ \leq\},\ \emptyset,\ \emptyset)$$

Formal Theories: Some Examples

$T3.$ **Specific Axioms**

There are two groups of axioms: Equality and Order

We adopt the LE (logic with equality) axioms to the language $\mathcal{L}_{T3}$ as follows

Equality Axioms

For any $x, y, z, x_1, x_2, y_1, y_2 \in VAR$

e1 $x = x$

e2 $(x = y \Rightarrow y = x)$

e3 $((x = y \cap y = z) \Rightarrow x = z)$

e4 $((x_1 = y_1 \cap x_2 = y_2) \Rightarrow (x_1 \leq x_2 \Rightarrow y_1 \leq y_2))$

Formal Theories: Some Examples

Partial Order Axioms

o1 $x \leq x$ *(reflexivity)*

o2 $((x \leq y \cap y \leq x) \Rightarrow x = y)$ *(antisymmetry)*

o3 $((x \leq y \cap y \leq z) \Rightarrow x \leq z)$ *(trasitivity)*

where $x, y, z \in VAR$

The **model** of $T3$ is called a **partially ordered** structure

Formal Theories: Some Examples

$T4.$ **Theory of Partial Order (2)**

Here is another formalization for partial order

Language

$$\mathcal{L}_{T4} = \mathcal{L}_{\{\neg, \Rightarrow, \cup, \cap\}}(\mathbf{P} = \{P\},\ \mathbf{F} = \emptyset,\ \mathbf{C} = \emptyset)$$

where $\# P = 2$ i.e. P is a two argument predicate

The **intended** interpretation of $P(x, y)$ is $x < y$, so we use the "less" symbol $<$ instead of P

We write $x \not< y$ for $\neg(x < y)$, i.e. for $\neg < (x, y)$

Formal Theories: Some Examples

We write the language of $T4$ as

$$\mathcal{L}_{T4} = \mathcal{L}_{\{\neg,\Rightarrow,\cup,\cap\}}(\{<\},\ \emptyset,\ \emptyset)$$

Specific Axioms

For any $x, y, z \in VAR$

p1 $x \nless x$ *(irreflexivity)*

p2 $((x \leq y \cap y \leq z) \Rightarrow x \leq z)$. *(trasitivity)*

Formal Theories: Some Examples

T5. **Theory of Linear Order**

Linear order relation is also called **total order** relation

Language

$$\mathcal{L}_{T5} = \mathcal{L}_{\{\neg, \Rightarrow, \cup, \cap\}}(\{=, \ \leq\}, \ \emptyset, \ \emptyset)$$

Specific Axioms

We adopt all axioms of theory *T3* of partial order and **add** the following additional axiom

o4 $\quad (x \leq y) \cup (y \leq x)$.

This axiom says that in linearly ordered sets each two elements are **comparable**

T6. Theory of Dense Order

Language

$$\mathcal{L}_{T6} = \mathcal{L}_{\{\neg, \Rightarrow, \cup, \cap\}}(\{=, \leq\}, \emptyset, \emptyset)$$

We write $x \neq y$ for $\neg(x = y)$, i.e. for the formula $\neg = (x, y)$

Specific Axioms

We adopt **all** axioms of theory $T5$ of **linear** order and **add** the following additional axiom

o5

$$((x \leq y \cap x \neq y) \Rightarrow \exists z((x \leq z \cap x \neq z) \cap (z \leq y \cap z \neq y)))$$

This axiom says that in linearly ordered sets between **any two** different elements there is a **third** element between them, respective to the order

Formal Theories: Some Examples

T7. **Lattice Theory**

Language

$$\mathcal{L}_{T7} = \mathcal{L}_{\{\neg,\Rightarrow,\cup,\cap\}}(\mathbf{P} = \{P, Q\},\ \mathbf{F} = \{f, g\},\ \mathbf{C} = \emptyset)$$

where P is a two argument predicate symbol

The **intended** interpretation of P is **equality**, so we use the equality symbol $=$ instead of P

Q is a two argument predicate symbol

The **intended** interpretation of Q is **partial order**, so we use the order symbol $\leq$ instead of Q

Formal Theories: Some Examples

f, g are a two argument functional symbols

The **intended** interpretation of f, g is the **lattice** intersection $\wedge$ and union $\vee$, respectively

We write $(x \wedge y)$ for $\wedge(x, y)$ and $(x \vee y)$ for $\vee(x, y)$

$(x \cap y)$, $(x \cup y)$ are **atomic** formulas of $\mathcal{L}_{T7}$ and $(x \wedge y)$ and $(x \vee y)$ are **terms** of $\mathcal{L}_{T7}$

We write the language of $T7.$ as

$$\mathcal{L}_{T7} = \mathcal{L}_{\{\neg, \Rightarrow, \cup, \cap\}}(\{=, \leq\},\ \{\wedge, \vee\},\ \emptyset)$$

Formal Theories: Some Examples

Specific Axioms

There are three groups of axioms: equality axioms, order axioms, and lattice axioms

Equality Axioms

We adopt the LE (logic with equality) axioms to the language $\mathcal{L}_{T7}$ as follows

e1 $x = x$

e2 $(x = y \Rightarrow y = x)$

e3 $((x = y \cap y = z) \Rightarrow x = z)$

e4 $((x_1 = y_1 \cap x_2 = y_2) \Rightarrow (x_1 \leq x_2 \Rightarrow y_1 \leq y_2))$

Formal Theories: Some Examples

e5 $((x_1 = y_1 \cap x_2 = y_2) \Rightarrow (x_1 \wedge x_2 \;\Rightarrow\; y_1 \wedge y_2))$

e6 $((x_1 = y_1 \cap x_2 = y_2) \Rightarrow (x_1 \vee x_2 \;\Rightarrow\; y_1 \vee y_2))$

where $x, y, z, x_1, x_2, y_1, y_2 \in VAR$

Remark

We use the symbol $\wedge$ for the lattice set intersection

functional symbol in order to better distinguish it from the

conjunction symbol $\cap$

The same applies to the axiom that involves lattice set union

functional symbol $\vee$ and the **disjunction** symbol $\cup$

Formal Theories: Some Examples

Partial Order Axioms

For any $x, y, z \in VAR$

o1 $\quad x \leq x \qquad$ *(reflexivity)*

o2 $\quad ((x \leq y \cap y \leq x) \Rightarrow x = y) \qquad$ *(antisymmetry)*

o3 $\quad ((x \leq y \cap y \leq z) \Rightarrow x \leq z) \qquad$ *(trasitivity)*

Lattice Axioms

For any $x, y, z \in VAR$

b1 $\quad (x \wedge y) = (y \wedge x), \qquad (x \vee y) = (x \vee y),$

b2 $\quad (x \wedge (y \wedge z)) = ((x \wedge y) \wedge z), \quad (x \vee (y \vee z)) = ((x \vee y) \vee z)$

b3 $\quad (((x \wedge y) \vee y) = y), \qquad ((x \wedge (x \vee y)) = x).$

Formal Theories: Some Examples

T8. **Theory of Distributive Lattices**

Language

$$\mathcal{L}_{T8} = \mathcal{L}_{\{\neg, \Rightarrow, \cup, \cap\}}(\{=, \leq\},\ \{\wedge, \vee\},\ \emptyset)$$

Specific Axioms

We adopt all axioms of the lattice theory *T7* and the following additional axiom

b4 $(x \wedge (y \vee z)) = ((x \wedge y) \vee (x \wedge z))$

Formal Theories: Some Examples

T9. **Theory of Boolean Algebras**

Language

$$\mathcal{L}_{T9} = \mathcal{L}_{\{\neg, \Rightarrow, \cup, \cap\}}(\{=, \leq\},\ \{\wedge, \vee, -\},\ \emptyset)$$

where $-$ is one argument function symbol representing algebra complement

Specific Axioms

We adopt all axioms of **distributive lattices** theory *T8*

and **add** the following axiom that characterizes the algebra complement $-$

b5 $(((x \wedge -x) \vee y) = y), \qquad (((x \vee -x) \wedge y) = y)$

Formal Theories: Some Examples

$T10$. **Theory of Groups**

Language

$$\mathcal{L}_{T10} = \mathcal{L}_{\{\neg,\Rightarrow,\cup,\cap\}}(\mathbf{P} = \{P\},\ \mathbf{F} = \{f, g\},\ \mathbf{C} = \{c\})$$

where P is a two argument **predicate** symbol

The **intended** interpretation of P is equality and we use the equality symbol $=$ instead of P

f is a two argument **functional** symbol

The **intended** interpretation of f is group operation $\circ$

We write $(x \circ y)$ for the term $\circ(x, y)$

Formal Theories: Some Examples

g is a one argument **functional** symbol

$g(x)$ **represents** a group inverse element to a given x

usually denoted it by x^{-1}

We hence use a symbol $^{-1}$ for g

c is a **constant** symbol representing group unit element e

Hence we use a symbol e for c

We write the language of $T10.$ as

$$\mathcal{L}_{T10} = \mathcal{L}_{\{\neg, \Rightarrow, \cup, \cap\}}(\{=\}, \{\circ, {}^{-1}\}, \{e\})$$

Formal Theories: Some Examples

Specific Axioms

There are two groups of axioms: equality and group axioms

Equality Axioms

We adopt the TE (theory with equality) axioms to the language $\mathcal{L}_{T10}$ as follows

For any $x, y, z, x_1, x_2, y_1, y_2, \in VAR$

e1 $x = x$

e2 $(x = y \Rightarrow y = x)$

e3 $((x = y \cap y = z) \Rightarrow x = z)$

e4 $(x = y \Rightarrow x^{-1} = y^{-1})$

e5 $((x_1 = y_1 \cap x_2 = y_2) \Rightarrow (x_1 \circ x_2 \Rightarrow y_1 \circ y_2))$

Formal Theories: Some Examples

Group Axioms

g1 $(x \circ (y \circ z)) = ((x \circ y) \circ z)$

g2 $(x \circ e) = x$

g3 $(x \circ x^{-1}) = e$

$T11$. **Theory of Abelian Groups**

Language is the same as $\mathcal{L}_{T10}$, i.e.

$$\mathcal{L}_{T11} = \mathcal{L}_{\{\neg, \Rightarrow, \cup, \cap\}}(\{=\},\ \{\circ,\ ^{-1}\},\ \{e\})$$

We adopt **all** axioms of theory $T10$ of groups and **add** the following axiom

g4 $(x \circ y) = (y \circ x)$

Elementary Theories

Observe that all what we can prove in the **formal** axiomatic theories defined and presented here **represents** only small fragments of corresponding **axiomatic theories** developed in **mathematics**

For example, Groups Theory, Lattices or Boolean Algebras Theories are whole, often large **fields** in **mathematics**

Elementary Theories

The theorems developed in the **axiomatic theories** in **mathematics** like for example the Representation Theorem for Boolean algebras, can not be even **expressed**, not to mention to be **proved** in the languages of respective formal theories

This is a reason why we also call the formal axiomatic theories **elementary theories**

For example, we say **Elementary Group Theory** to distinguish it from the Group Theory as a much lager and complicated field of mathematics

Chapter 11
Formal Theories and Gödel Theorems

PART 2: **PA:** Formal Theory of **Natural Numbers**

Peano Arithmetic PA

Next to geometry, the **theory of natural numbers** is the most intuitive and intuitively known of all branches of mathematics

This is why the first attempts to **formalize mathematics** begin with arithmetic of natural numbers.

The first attempt of axiomatic formalization was given by Dedekind in **1879** and by Peano in **1889**

The Peano formalization became known as **Peano Postulates** and can be written as follows.

Peano Arithmetic PA

Peano Postulates

p1 0 is a natural number

p2 If n is a natural number, there is another number which we denote by n'
We call the number n' a **successor** of n and the intuitive meaning of n' is $n+1$

p3 $0 \neq n'$, for any natural number n

p4 If $n' = m'$, then $n = m$, for any natural numbers n, m

Peano Arithmetic PA

p5 **If** W is is a property that may or may not hold for natural numbers, and
if **(i)** 0 has the property W and
(ii) whenever a natural number n has the property W, then n' has the property W,
then all natural numbers have the property W

The postulate **p5** is called Principle of Induction

Peano Arithmetic PA

The **Peano Postulates** together with certain amount of set theory are sufficient to develop **not only** theory of natural numbers, **but also** theory of rational and even real numbers

But **Peano Postulates** can't act as a fully formal theory as they include **intuitive** notions like "property" and "has a property"

A **formal theory** of natural numbers based on the Peano Postulates is referred in literature as **Peano Arithmetic**, or simply PA

We present here formalization by Mendelson (1973)

It is included and worked out in the smallest **details** in his book *Intoduction to Mathematical Logic* (1987)

We refer the reader to this excellent book for details and further reading

Peano Arithmetic PA

We assume, as we did for all other formal theories, that the Peano Arithmetic PA is based on a **complete** Hilbert style proof system

$$H = (\mathcal{L}, \mathcal{F}, LA, \mathcal{R})$$

for classical predicate logic with a language

$$\mathcal{L} = (\mathcal{L}_{\{\neg,\cap,\cup,\Rightarrow\}}(\mathbf{P}, \mathbf{F}, \mathbf{C})$$

We **additionally** assume that the system H has as one of the inference rules a generalization rule

$$(G) \quad \frac{A(x)}{\forall x A(x)}$$

We do so to facilitate the use of the Mendelson's book as a supplementary reading to the material included here and for additional reading for material not covered here

PA **Peano Arithmetic**

Language is

$$\mathcal{L}_{PA} = \mathcal{L}(\mathbf{P} = \{P\},\ \mathbf{F} = \{f, g, h\},\ \mathbf{C} = \{c\})$$

where the predicate P represents the equality $=$ and we write $x \neq y$ for the formula $\neg(x = y)$

the functional symbol f represents the successor $'$

the functional symbols g, h represent addition $+$ and the multiplication $\cdot$, respectively

c is a constant symbol representing zero and we use a symbol 0 to denote c

We write the language of PA as

$$\mathcal{L}_{PA} = \mathcal{L}_{\{\neg, \Rightarrow, \cup, \cap\}}(\{=\},\ \{\ ',\ +,\ \cdot\},\ \{0\})$$

Peano Arithmetic PA

Specific Axioms

P1 $(x = y \Rightarrow (x = z \Rightarrow y = z))$,

P2 $(x = y \Rightarrow x' = y')$,

P3 $0 \neq x'$,

P4 $(x' = y' \Rightarrow x = y)$,

P5 $x + 0 = x$,

P6 $x + y' = (x + y)'$

P7 $x \cdot 0 = 0$,

P8 $x \cdot y' = (x \cdot y) + x$,

P9 $(A(0) \Rightarrow (\forall x(A(x) \Rightarrow A(x')) \Rightarrow \forall x A(x))))$,

for all formulas $A(x)$ of $\mathcal{L}_{PA}$ and all $x, y, z \in VAR$

Peano Arithmetic PA

The axiom P9 is called Principle of Mathematical Induction

It **does not** fully corresponds to Peano Postulate **p5** which refers intuitively to all (uncountably many) possible properties of natural numbers

The axiom P7 applies only to properties defined by infinitely countably many formulas of $A(x)$ of $\mathcal{L}_{PA}$

Axioms P3, P4 correspond to Peano Postulates **p3, p4**

The Peano Postulates **p1, p2** are taken care of by presence of 0 and successor function

Peano Arithmetic PA

Axioms P1, P2 deal with some needed properties of equality that were probably assumed as intuitively obvious by Peano and Dedekind

Axioms P5 - P8 are the recursion equations for addition and multiplication

They are **not stated** in the Peano Postulates as Dedekind and Peano allowed the use of intuitive **set theory** within which the **existence** of addition and multiplication and their properties P5 - P8 can be **proved** (Mendelson, 1973)

Peano Arithmetic PA

Observe that while axioms P1 - P9 of Peano Arithmetic PA are particular formulas of $\mathcal{L}_{PA}$ and the axiom P9 is an **axiom schema** providing an infinite number of axioms

This means that the set of axioms P1 - P9 **do not** provide a **finite** axiomatization for Peano Arithmetic

The following was **proved** formally by Czeslaw Ryll-Nardzewski in 1952 and again by Rabin in 1961

Peano Arithmetic PA

Ryll-Nardzewski Theorem

Peano Arithmetic is **is not** finitely axiomatizable

That is there **is no** theory K having only a finite number of proper axioms, whose theorems are the same as those of PA

Observe that the theory PA is **one** of many formalizations of the Peano Arithmetic

We denoted by $\mathbf{T}$ the set all provable formulas in T

In particular, $\mathbf{PA}$ denotes the set of all formulas provable in theory PA and we adopt the following definition

Peano Arithmetic PA

Definition

Any theory T such that **T = PA** is called a Peano Arithmetic

For example, taking **closure** of axioms P1 - P8 of T14 we obtain new theory *CPA*

The axiom P9 is a **sentence** (closed formula) already

Peano Arithmetic CPA

Theory CPA

Language is $\mathcal{L}_{CPA} = \mathcal{L}_{PA} = \mathcal{L}_{\{\neg,\Rightarrow,\cup,\cap\}}(\{=\}, \{', +, \cdot\}, \{0\})$

Specific Axioms

C1 $\forall x \forall y \forall z (x = y \Rightarrow (x = z \Rightarrow y = z))$

C2 $\forall x \forall y (x = y \Rightarrow x' = y')$

C3 $\forall x (0 \neq x')$

C4 $\forall x \forall y (x' = y' \Rightarrow x = y)$

C5 $\forall x (x + 0 = x)$

C6 $\forall x \forall y (x + y' = (x + y)')$

C7 $\forall x (x \cdot 0 = 0)$

C8 $\forall x \forall y (x \cdot y' = (x \cdot y) + x)$

C9 $(A(0) \Rightarrow (\forall x (A(x) \Rightarrow A(x')) \Rightarrow \forall x A(x)))$

for all formulas $A(x)$ of $\mathcal{L}_{CPA}$

Fact 1

Theory CPA is a Peano Arithmetic

Proof

We have to show that **PA = CPA**

As both theories are based on the same language $\mathcal{L}_{PA}$ we have to show that for any formula B

$$\vdash_{PA} B \quad \text{if and only if} \quad \vdash_{CPA} B$$

Both theories are also based on the same proof system H, so we have to prove that

(1) all axioms $C1 - C8$ of CPA are provable in PA and

(2) all axioms $P1 - P8$ of PA are provable in CPA

Peano Arithmetic CPA

Here are detailed **proofs** for axioms P1, and C1

The proofs for other axioms follow the same pattern

(1) We prove that the axiom

C1 $\forall x \forall y \forall z (x = y \Rightarrow (y = z \Rightarrow x = z))$

is **provable** in *PA* as follows

Observe that axioms of CPA are **closures** of respective axioms of *PA*

Consider axiom

P1 $(x = y \Rightarrow (y = z \Rightarrow x = z))$

Peano Arithmetic CPA

As the proof system H has a **generalization rule**

$$(G)\quad \frac{A(x)}{\forall x A(x)}$$

we obtain a **formal proof**

$$B1,\ B2,\ B3,\ B4$$

of C1 as follows

B1: $(x = y \Rightarrow (x = z \Rightarrow y = z))$, P1

B2: $\forall z(x = y \Rightarrow (x = z \Rightarrow y = z))$ (G) rule

B3: $\forall y \forall z(x = y \Rightarrow (x = z \Rightarrow y = z))$ (G) rule

B4: $\forall x \forall y \forall z(x = y \Rightarrow (x = z \Rightarrow y = z))$ C1

This **ends** the proof of (1) for axioms P1, and C1

Peano Arithmetic CPA

(2) We prove now that the axiom

P1 $(x = y \Rightarrow (y = z \Rightarrow x = z))$

is provable in CPA

By completeness of H we know that the predicate tautology

$$(**) \quad (\forall x A(x) \Rightarrow A(t))$$

where term t is free for x in $A(x)$

is **provable** in H for any formula $A(x)$ of $\mathcal{L}$ and hence for any formula $A(x)$ of its particular **sublanguage** $\mathcal{L}_{PA}$

So for its particular case of

A(x) $= (x = y \Rightarrow (x = z \Rightarrow y = z))$ and $t = x$

$$(*) \quad \vdash_{CPA} (\forall x \forall y \forall z (x = y \Rightarrow (x = z \Rightarrow y = z))$$

$$\Rightarrow \forall y \forall z (x = y \Rightarrow (x = z \Rightarrow y = z)))$$

Peano Arithmetic CPA

We construct a formal proof $B1,\ B2,\ B3,\ B4,\ B5,\ B6,\ B7$ of P1 in CPA in as follows

B1 $\forall x \forall y \forall z (x = y \Rightarrow (x = z \Rightarrow y = z))$ C1

B2 $(\forall x \forall y \forall z (x = y \Rightarrow (x = z \Rightarrow y = z))$
$\Rightarrow \forall y \forall z (x = y \Rightarrow (x = z \Rightarrow y = z)))$ (ast)

B3 $\forall y \forall z (x = y \Rightarrow (x = z \Rightarrow y = z))$ MP on B1, B2

B4 $(\forall y \forall z (x = y \Rightarrow (x = z \Rightarrow y = z))$
$\Rightarrow \forall z (x = y \Rightarrow (x = z \Rightarrow y = z)))$ (ast)

Peano Arithmetic CPA

B5 $\forall z(x = y \Rightarrow (x = z \Rightarrow y = z))$, MP on B3, B4

B6 $(\forall z(x = y \Rightarrow (x = z \Rightarrow y = z))$
$\Rightarrow (x = y \Rightarrow (x = z \Rightarrow y = z)))$, (ast)

B7 $(x = y \Rightarrow (x = z \Rightarrow y = z))$ MP on B5, B6

This **ends** the proof of (2) for axioms P1, and C1

The proofs for other axioms is similar and are left as homework assignment

Peano Arithmetic PA

Here are some basic facts about *PA*

Fact 2

The following formulas are provable in *PA* for any terms t, s, r of $\mathcal{L}_{PA}$

P1' $(t = r \Rightarrow (t = s \Rightarrow r = s))$

P2' $(t = r \Rightarrow t' = r')$

P3' $0 \neq t'$

P4' $(t' = r' \Rightarrow t = r)$

P5' $t + 0 = t$

P6' $t + r' = (t + r)'$

P7' $t \cdot 0 = 0$

P8' $t \cdot r' = (t \cdot r) + t$

Peano Arithmetic PA

We named the **Fact 1** properties as P1'- P8' to stress the fact that they are **generalizations** of axioms P1 - P8 of PA to the set of all terms of the language $\mathcal{L}_{PA}$

Proof

We write the proof for P1' as an example

Proofs of all other formulas follow the same pattern

Consider axiom

P1: $(x = y \Rightarrow (y = z \Rightarrow x = z))$

By the **Fact 1** its closure is **provable** in PA, i.e.

$$(*) \quad \vdash_{PA} \forall x \forall y \forall z (x = y \Rightarrow (x = z \Rightarrow y = z))$$

Peano Arithmetic PA

By completeness of H we know that the predicate tautology

$$(PT) \quad (\forall x A(x) \Rightarrow A(t))$$

where term t is free for x in $A(x)$

is **provable** in H for any formula $A(x)$ of $\mathcal{L}$

So it is also provable for a formula

A(x) $= \forall y \forall z (x = y \Rightarrow (x = z \Rightarrow y = z))$

Observe that any term t is free for x in this **particular** $A(x)$ so we get that for any term t the following holds

$$(**) \quad \vdash_{PA} (\forall x \forall y \forall z (x = y \Rightarrow (x = z \Rightarrow y = z))$$
$$\Rightarrow \forall y \forall z (t = y \Rightarrow (t = z \Rightarrow y = z)))$$

Peano Arithmetic PA

Applying MP to $(*)$ and $(**)$ we get that for any term t

$$(a) \vdash_{PA} \forall y \forall z(t = y \Rightarrow (t = z \Rightarrow y = z))$$

Observe that any term r is free for for y in

$$\forall z(t = y \Rightarrow (t = z \Rightarrow y = z))$$

so we have that for all terms r

$$(aa) \vdash_{PA} (\forall y \forall z(t = y \Rightarrow (t = z \Rightarrow y = z))$$
$$\Rightarrow \forall z(t = r \Rightarrow (t = z \Rightarrow r = z)))$$

as a particular case of the tautology (PT)

Peano Arithmetic PA

Applying MP to (a) and (aa) we get that for any terms t, r

$$(b) \quad \vdash_{PA} \forall z(t = r \Rightarrow (t = z \Rightarrow r = z))$$

Observe that any term s is free for z in the formula

$$(t = r \Rightarrow (t = z \Rightarrow r = z))$$

and so we have that

$$(bb) \quad \vdash_{PA} (\forall z(t = y \Rightarrow (t = z \Rightarrow y = z))$$

$$\Rightarrow (t = r \Rightarrow (t = s \Rightarrow r = s)))$$

for all terms r, t, s as a particular case of the tautology (PT)

Peano Arithmetic PA

Applying MP to (b) and (bb) we get that for any terms t, r

$$\vdash_{PA} \ (t = r \Rightarrow (t = s \Rightarrow r = s))$$

This **ends** the proof of $P1'$

The proofs of properties P2' - P8' follow the same pattern and are left as an exercise

As the next step we use **Fact 1** and **Fact 2**, the axioms of PA, and the **completeness** of the proof system H to prove the following **Fact 3**

The details of the steps in the proof, similar to the proof of the **Fact 2** are left to the reader as an exercise

Peano Arithmetic PA

Fact 3

The following formulas are **provable** in *PA* for any terms t, s, r of $\mathcal{L}_{PA}$

a1 $t = t$

a2 $(t = r \Rightarrow r = t)$

a3 $(t = r \Rightarrow (r = s \Rightarrow t = s))$

a4 $(r = t \Rightarrow (t = s \Rightarrow r = s))$

a5 $(t = r \Rightarrow (t + s = r + s))$

a6 $t = 0 + t$

Proof

The full details of the steps in the proof, similar to the proof of the **Fact 2** are left to the reader as an exercise

Peano Arithmetic PA

a1 $t = t$

We construct a formal proof

$$B1,\ B2,\ B3,\ B34$$

of $t = t$ in PA in as follows

B1 $t + 0 = t$ P5' in Fact 2

B2 $(t + 0 = t \Rightarrow (t + 0 = t \Rightarrow t = t))$

P1' in Fact 2 for $t = t + 0,\ r = t,\ s = t$

B3 $(t + 0 = t \Rightarrow t = t)$ MP on B1, B2

B4 $t = t.$ MP on B1, B3

Peano Arithmetic PA

a2 $\quad (t = r \Rightarrow r = t)$

We construct a formal proof

$$B1,\ B2,\ B3,\ B4$$

of a2 as follows.

B1 $\quad (t = r \Rightarrow (t = t \Rightarrow r = t))$

P1' in Fact 2 for $r = t, \quad s = t$

B2 $\quad (t = t \Rightarrow (t = r \Rightarrow r = t))$ tautology, B1

B3 $\quad t = t$ already proved a1

B4 $\quad (t = r \Rightarrow r = t)$ MP on B2, B3

Peano Arithmetic PA

a3 $\quad (t = r \Rightarrow (r = s \Rightarrow t = s))$

We construct a formal proof

$$B1,\; B2,\; B3$$

of a3 as follows.

B1 $\quad (r = t \Rightarrow (r = s \Rightarrow t = s))$ P1' in Fact 2

B2 $\quad (t = r \Rightarrow r = t)$ already proved a2

B3 $\quad (t = r \Rightarrow (r = s \Rightarrow t = s))$ tautology, B1, B2

Peano Arithmetic PA

a4 $\quad (r = t \Rightarrow (t = s \Rightarrow r = s))$

We construct a formal proof

$$B1,\ B2,\ B3,\ B4,\ B5$$

of a4 as follows.

B1 $\quad (r = t \Rightarrow (t = s \Rightarrow r = s)) \quad$ a3 for $t = r,\ r = t$

B2 $\quad (t = s \Rightarrow (r = t \Rightarrow r = s)) \quad$ B1, tautology

B3 $\quad (s = t \Rightarrow t = s) \quad$ a2

B4 $\quad (s = t \Rightarrow (r = t \Rightarrow r = s)) \quad$ B1, B2, tautology

B5 $\quad (r = t \Rightarrow (t = s \Rightarrow r = s)) \quad$ B4, tautology

Peano Arithmetic PA

a5 $(t = r \Rightarrow (t + s = r + s))$

We **prove** *a5* by the Principle of Mathematical Induction

$$P9 \quad (A(0) \Rightarrow (\forall x(A(x) \Rightarrow A(x') \Rightarrow \forall x A(x))))$$

The proof uses the **Deduction Theorem** which holds for the proof system H and so it can be used in PA

We **first** apply the Induction Rule P9 to the formula

$$A(z): \quad (x = y \Rightarrow x + z = y + z)$$

to prove

$$\vdash_{PA} \forall z(x = y \Rightarrow x + z = y + z)$$

Peano Arithmetic PA

Proof of the formula $\forall z(x = y \Rightarrow x + z = y + z)$
by the Principle of Mathematical Induction

$$P9 \quad (A(0) \Rightarrow (\forall x(A(x) \Rightarrow A(x') \Rightarrow \forall xA(x))))$$

applied to the forrmula

$$A(z): \quad (x = y \Rightarrow x + z = y + z)$$

(i) We prove initial step $A(0)$, i.e. we prove that

$$\vdash_{PA} \ (x = y \Rightarrow x + 0 = y + 0)$$

Here the steps in the proof

B1 $\quad x + 0 = x \quad$ P5'

B2 $\quad y + 0 = y \quad$ P5'

Peano Arithmetic PA

B3 $\quad x = y \quad$ Hyp

B4 $\quad (x + 0 = x \Rightarrow (x = y \Rightarrow x + 0 = y) \quad$ a3 for $t = x + 0,\ r = x, s = y$

B5 $\quad (x = y \Rightarrow x + 0 = y) \quad$ MP on B1, B4

B6 $\quad x + 0 = y \quad$ MP on B3, B5

B7 $\quad (x + 0 = y \Rightarrow (y + 0 = y \Rightarrow x + 0 = y + 0), \quad$ a4 for $r = x + 0, t = y, s = y = 0$

B8 $\quad (y + 0 = y \Rightarrow x + 0 = y + 0) \quad$ MP on B6, B7

B9 $\quad x + 0 = y + 0) \quad$ MP on B2, B8

B10 $\quad (x = y \Rightarrow x + 0 = y + 0) \quad$ B1- B9, Deduction Theorem

Thus, $\quad \vdash_{PA} \ A(0)$

Peano Arithmetic PA

(ii) We prove inductive step $\forall z(A(z) \Rightarrow A(z'))$

i.e. prove that

$$\vdash_{PA} \forall z((x = y \Rightarrow x+z = y+z) \Rightarrow (x = y \Rightarrow x+z' = y+z'))$$

Here the steps in the proof

C1 $(x = y \Rightarrow x + z = y + z)$ Hyp

C2 $x = y$ Hyp

C3 $x + z' = (x + z)'$ P6'

C4 $y + z' = (y + z)'$ P6'

Peano Arithmetic PA

C5 $(x + z = y + z)$ MP on C1, C2

C6 $(x + z = y + z \Rightarrow (x + z)' = (y + z)')$ P2' for $t = x + z,\ r = y + z$

C7 $(x + z)' = (y + z)'$ MP on C5, C6

C8 $x + z' = y + z'$, a3 substitution, MP on C3, C7

C9 $((x = y \Rightarrow x + z = y + z) \Rightarrow x + z' = y + z')$ C1- C8, Deduction Theorem

This proves $\vdash_{PA}\ A(z) \Rightarrow A(z')$

Peano Arithmetic PA

C10 $(((x = y \Rightarrow x + 0 = y + 0) \Rightarrow ((x = y \Rightarrow x + z = y + z) \Rightarrow x + z' = y + z')) \Rightarrow \forall z(x = y \Rightarrow x + z = y + z))$

P9 for $A(z):\ (x = y \Rightarrow x + z = y + z)$

C11 $((x = y \Rightarrow x + z = y + z) \Rightarrow x + z' = y + z')) \Rightarrow \forall z(x = y \Rightarrow x + z = y + z)$ MP on C10 and B10

C12 $\forall z(x = y \Rightarrow x + z = y + z)$ MP on C11, C9

C13 $\forall y \forall z(x = y \Rightarrow x + z = y + z)$ (G) rule

C14 $\forall x \forall y \forall z(x = y \Rightarrow x + z = y + z)$ (G) rule

Peano Arithmetic PA

Now we **repeat** here the proof of P1' of **Fact 2**

We apply it step by step to C14

We **eliminate** the quantifiers $\forall x \forall y \forall z$ and **replace** variables x, y, z by terms t, r, s using the tautology

$$(\forall x A(x) \Rightarrow A(t))$$

and Modus Ponens (MP) rule

Finally, we obtain the proof of a5, i.e. we prove that

$$\vdash_{PA} \ (t = r \Rightarrow (t + s = r + s))$$

Peano Arithmetic PA

We go on proving other basic properties of addition and multiplication including for example the following

Fact

The following formulas are provable in PA for any terms $t, , s, r$ of $\mathcal{L}_{PA}$

(i) $t \cdot (r + s) = (t \cdot r) + (t \cdot s)$ distributivity

(ii) $(r + s) \cdot t = (r \cdot t) + (s \cdot t)$ distributivity

(iii) $(r \cdot t) \cdot s = r \cdot (t \cdot s)$ associativity

(iv) $(t + s = r + s \Rightarrow t = r)$ cancelation

Numerals in PA

Numerals Definition

The terms $0,\ 0',\ 0'',\ 0''',\ldots$ are called **numerals** and denoted by

$$\overline{0},\ \ \overline{1},\ \ \overline{2},\ \ \overline{3},\ \ \ldots\ \ \ldots$$

More precisely,

(1) the term $\overline{0}$ is number 0

(2) for any natural number n,

$$\overline{n+1} \quad \text{is} \quad (\overline{n})'$$

In general, if n is a natural number,

$\overline{n}$ stands for the corresponding **numeral** $0^{'''\ldots''}$, i.e. by 0 followed by n strokes

Numerals in PA

The **numerals** can be defined recursively as follows

(1) 0 is a numeral

(2) if u is a numeral, then u' is also a numeral

Here are some more of many properties, intuitively obvious, that **provable** in PA

We give in the chapter some **proofs** and an example , and leave the others as an exercise

Reminder

We use $\overline{n}$, $\overline{m}$ as un **abbreviation** of the terms r, s they represent

Peano Arithmetic PA

Fact

The following formulas are **provable** in *PA* for any terms t, s of $\mathcal{L}_{PA}$

1. $t + \overline{1} = t'$
2. $t \cdot \overline{1} = t$
3. $t \cdot \overline{2} = t + t$
4. $(t + s = 0 \Rightarrow (t = 0 \cap s = 0))$
5. $(t \neq 0 \Rightarrow (s \cdot t = 0 \Rightarrow s = 0))$

Proof

Major steps in the proof of 1. - 5. are presented in the chapter

Peano Arithmetic PA

For example, we construct the **proof** of

4. $(t + s = 0 \Rightarrow (t = 0 \cap s = 0))$

in the following sequence of steps

(s1) We apply the Principle of Mathematical Induction P9 to

$$A(y) : \; (x + y = 0 \Rightarrow (x = 0 \cap y = 0))$$

and prove

$$(*) \quad \forall y(x + y = 0 \Rightarrow (x = 0 \cap y = 0))$$

(s2) We apply the generalization rule (G) to $(*)$ and get

$$(**) \quad \forall x \forall y(x + y = 0 \Rightarrow (x = 0 \cap y = 0))$$

Peano Arithmetic PA

(s3) We now repeat here the proof of P1' of **Fact 2**

We apply it step by step to $(**)$ as follows

We **eliminate** the quantifiers $\forall x \forall y$ and **replace** variables x, y by terms t, s using (MP) rule and the tautology

$$(\forall x A(x) \Rightarrow A(t))$$

Finally, we obtain the **proof** of 4., i.e. we have proved that

$$\vdash_{PA} \ (t + s = 0 \Rightarrow (t = 0 \cap s = 0))$$

Peano Arithmetic PA

We also prove in the chapter, as an example, the following

Fact

Let n, m be any natural numbers

(1) If $m \neq n$, then $\overline{m} \neq \overline{n}$

(2) $\overline{m+n} = \overline{m} + \overline{n}$ and $\overline{m \cdot mn} = \overline{m} \cdot \overline{n}$

are **provable** in PA

(3) Any **model** for PA is infinite

Order Relation in PA

An **order relation** can be introduced in PA as follows

Order Relation **Definition**

Let t, s be any terms of $\mathcal{L}_{PA}$

We write $t < s$ for a formula $\exists w(w \neq 0 \cap w + t = s)$

where we choose w to be the first variable **not** in t or s

We write $t \leq s$ for a formula $(t < s \cup t = s)$

We write $t > s$ for a formula $s < t$ and

$t \geq s$ for a formula $s \leq t$

$t \not< s$ for a formula $\neg(t < s)$

and so on...

Order Relation in PA

Then we prove properties of **order** relation, for example the following.

Fact

For any terms t, r, s of $\mathcal{L}_{PA}$, the following formulas are **provable** in PA

o1 $t \leq t$

o2 $(t \leq s \Rightarrow (s \leq r \Rightarrow t \leq r))$

o3 $((t \leq s \cap s \leq t) \Rightarrow t = s)$

o4 $(t \leq s \Rightarrow (t + r \leq s + r))$

o5 $(r > 0 \Rightarrow (t > 0 \Rightarrow r \cdot t > 0))$.

Complete Inductionin PA

There are several **stronger forms** of the the

Principle of Mathematical Induction

P9 $\quad (A(0) \Rightarrow (\forall x(A(x) \Rightarrow A(x') \Rightarrow \forall x A(x))))$

that are **provable** in *PA*. Here is one of them

Fact

The following formula, called Complete Induction Principle

(PCI) $\quad (\forall x \forall z(z < x \Rightarrow A(z)) \Rightarrow A(x)) \Rightarrow \forall x A(x))$

is **provable** in *PA*

In plain English, the (PCI) says:

consider a property **P** such that, for any x, if **P** holds for for all natural numbers **less then** x, then **P** holds for x also.

Then **P** holds for all natural numbers

Mendelson Book

We **proved** and cited only some of the **basic properties** corresponding to properties of arithmetic of natural numbers

There are many more of them, developed in many Classical Logic textbooks

We **refer** the reader especially to Mendelson (1997) book:
Introduction to Mathematical Logic, Fourth Edition, Wadsworth&Brooks/Cole Advanced Books &Software

We found this book the most rigorous and complete

The **proofs** included in this chapter are detailed versions of few of Mendelson's proofs.

Peano Arithmetic PA

We selected and **proved** some direct consequences Peano Arithmetic axioms **not only** because they are needed as the starting point for a **strict development** of the formal theory of arithmetic of natural numbers **but also** because they are good examples of how one **develops** any formal theory

From this point on one can generally **translate** onto the language $\mathcal{L}_{PA}$ and prove in the PA the **results** from any textbook on elementary number theory

Robinson Arithmetic RR

We know by Ryll Nardzewski Theorem that the Peano Arithmetic PA **is not finitely** axiomatizable

We want now to bring reader's attention a Robinson Arithmetic RR that is a proper sub-theory of PA and which **is finitely** axiomatizable

Moreover, the Robinson Arithmetic RR has the same **expressive power** as PA with respect to the Gödel Theorems discussed and **proved** in the next section

Here it is, as **formalized** and discussed in detail in the Mendelson's book.

Robinson Arithmetic RR

RR Robinson Arithmetic

Language

The language of *RR* is the same as the language of *PA*, i.e.

$$\mathcal{L}_{RR} = \mathcal{L}_{\{\neg, \Rightarrow, \cup, \cap\}}(\{=\}, \{{}', +, \cdot\}, \{0\})$$

Specific Axioms

r1 $x = x$

r2 $(x = y \Rightarrow y = x)$

r3 $(x = y \Rightarrow (y = z \Rightarrow x = z))$

r4 $(x = y \Rightarrow x' = y')$

r5 $(x = y \Rightarrow (x + z = y + z \Rightarrow z + x = z + y))$

r6 $(x = y \Rightarrow (x \cdot z = y \cdot z \Rightarrow z \cdot x = z \cdot y))$

Robinson Arithmetic RR

r7 $(x' = y' \Rightarrow x = y)$

r8 $0 \neq x'$

r9 $(x \neq 0 \Rightarrow \exists y\ x = y')$

r10 $x + 0 = x$

r11 $x + y' = (x + y)'$

r12 $x \cdot 0 = 0$

r13 $x \cdot y' = x \cdot y + x$

r14 $(y = x \cdot z + p \cap ((p < x \cap y < x \cdot q + r) \cap r < x) \Rightarrow p = r)$

for any $x, y, z, p, q, r \in VAR$

Robinson Arithmetic RR

Axioms r1 - r13 are due to Robinson (1950)

Axiom r14 is due to Mendelson (1973)

It expresses the uniqueness of remainder

The relation $<$ is the order relation as defined in PA

Gödel showed in his famous **Incompleteness Theorem** that there are closed formulas of the language $\mathcal{L}_{PA}$ of the Peano Arithmetic PA that **are** neither **provable** nor **disprovable** in PA, if PA is **consistent**

Robinson Arithmetic RR

Hence, the Gödel **Incompleteness Theorem** also says that there is a formula that is **true** under standard interpretation but is **not provable** in *PA*

We also see that the **incompleteness** of *PA* cannot be attributed to omission of some essential axiom but has **deeper** underlying causes that apply to other theories as well

Robinson proved in 1950, that the **Gödel Theorems** hold in his system *RR* and that *RR* has the same incompleteness property as *PA*

Chapter 11
Formal Theories and Gödel Theorems

PART 3: Consistency, Completeness, **Gödel Theorems**

PART 4: **Proof** of the **Gödel Incompleteness Theorems**

Chapter 11
Formal Theories and Gödel Theorems

PART 3: Consistency, Completeness, Gödel Theorems

Consistency, Completeness, Gödel Theorems

Formal theories, because of their precise structure, became themselves an **object** of of mathematical research

The mathematical theory concerned with the study of **formalized** mathematical theories is called, after Hilbert, **metamathematics**

The most important open problems of **metamathematics** were introduced by Hilbert as a part of the **Hilbert Program**

They were concerned with notions of **consistency**, **completeness**, and **decidability**

Consistency, Completeness, Gödel Theorems

The answers to Hilbert problems of **consistency** and **completeness** of formal theories were given by Gödel in 1930 in a form of the two Gödel **Theorems**

They are some of the most important and influential results in twentieth century **mathematics**

Consistency

There are two definitions of **consistency**: semantical and syntactical

The **semantical** definition is based on the notion of a **model** and says, in plain English:

" a theory is **consistent** *if the set of its specific axioms has a* **model** *"*

The syntactical definition uses the notion of **provability** and says:

"a theory is **consistent** *if one can't* **prove** *a contradiction in it "*

Consistency

We have used the syntactical definition in chapter 5 in the proof the **completeness theorem** for the propositional logic
In chapter 9 we used the semantical one

We **extend** now these propositional definitions to the predicate language and **formal theories**

In order to distinguish these two definitions of **consistency** we call the semantical one **model-consistent**, and we call the syntactical one just **consistent**

Model - Consistency

Model for a Theory

Given a first order theory

$$(\mathcal{L},\ \mathcal{F},\ LA,\ SA,\ \mathcal{R})$$

Any structure $\mathcal{M} = [M, I]$ that is a model for the set SA of the specific axioms of T is called a **model** for the theory T

Model - Consistent Theory

A first order theory $T = (\mathcal{L},\ \mathcal{F},\ LA,\ SA,\ \mathcal{R})$ is model - consistent if and only if it has a **model**

Standard Model for PA

Consider the Peano Arithmetics PA and a structure $\mathcal{M} = [M, I]$ for its language

$$\mathcal{L}_{PA} = \mathcal{L}_{\{\neg, \Rightarrow, \cup, \cap\}}(\{=\}, \{', +, \cdot\}, \{0\})$$

such that the universe M is the set N of natural numbers (nonnegative integers) and the interpretation I is defined as follows

(1) the constant symbol 0 is interpreted as a natural number 0

Standard Model for PA

(2) the one argument function symbol $'$ (successor) is interpreted as successor operation (addition of 1) on natural numbers;

$$succ(n) = n + 1$$

(3) the two argument function symbols $+$ and $\cdot$ are interpreted as ordinary addition and multiplication in N

(4) the predicate symbol "=" is interpreted as equality relation in N

Standard Model for PA

Standard Model for PA

We denote $\mathcal{M} = [N, I]$ for I defined by (1) - (4) as

$$\mathcal{M} = [N, \ =, \ succ, \ +, \ \cdot \ ]$$

and call it a **standard model** for *PA*

The interpretation I is called a **standard interpretation**

Any **model** for *PA* in which the predicate symbol "=" is interpreted as equality relation in N that **is not** isomorphic to the standard model is called a **nonstandard model** for *PA*

Standard Model for PA

Observe that if we recognize that the set N of natural numbers with the **standard interpretation** i.e. the structure

$$\mathcal{M} = [N, =, succ, +, \cdot]$$

to be a **model** for PA, then, of course, PA is **consistent**

However, semantic methods, involving a fair amount of set-theoretic reasoning, are regarded by many (and were regarded as such by Gödel) as too **precarious** to serve as **basis** of consistency proofs

Standard Model and Consistency

Moreover, we have **not proved** formally that the axioms of *PA* are **true** under standard interpretation

We only have taken it as intuitively **obvious**

Hence for this and other reasons it is common practice to take the **model-consistency** of *PA* as un explicit, unproved **assumption** and to adopt, after Gödel the following syntactic definition of **consistency**

Consistent Theory

Given a theory $T = (\mathcal{L}, \mathcal{F}\ LA, SA, \mathcal{R})$

Let $\mathbf{T}$ be the set of all **provable** formulas in T

The theory T is **consistent** if and only if **there is no** formula A of the language $\mathcal{L}_{SA}$ such that

$$\vdash_T A \quad \text{and} \quad \vdash_T \neg A$$

i.e. **there is no** formula A such that

$$A \in \mathbf{T} \quad \text{and} \quad \neg A \in \mathbf{T}$$

Inconsistent Theory

Inconsistent Theory

The theory $T = (\mathcal{L}, \mathcal{F}, LA, SA, \mathcal{R})$ is **inconsistent**

if and only if

there is a formula A of the language $\mathcal{L}_{SA}$ such that

$$\vdash_T A \quad \text{and} \quad \vdash_T \neg A$$

i.e. **there is** a formula A such that

$$A \in \mathbf{T} \quad \text{and} \quad \neg A \in \mathbf{T}$$

Consistency Theorem

Here is a basic characterization of consistent theories

Theorem

A theory $T = (\mathcal{L}, \mathcal{F}, LA, SA, \mathcal{R})$ based on a complete proof system $H = (\mathcal{L}, \mathcal{F}, LA, \mathcal{R})$ is **consistent** if and only if **there is** a formula A of the language $\mathcal{L}_{SA}$ such that

$$A \notin \mathbf{T}$$

Proof

Let denote by CD the consistency condition in the consistency **definition** and by CT consistency condition in the **theorem**

Consistency Theorem Proof

1. We prove implication " if CD, then CT "

Assume **not** CT

This means that $A \in \mathbf{T}$ for all formulas A

In particular **there is** a formula B such that

$$B \in \mathbf{T} \quad \text{and} \quad \neg B \in \mathbf{T}$$

and **not** CD holds

2. We prove implication " if CT, then CD "

Assume **not** CD

This means that **there is** A of $\mathcal{L}_{SA}$, such that $A \in \mathbf{T}$

$$(*) \quad A \in \mathbf{T} \quad \text{and} \quad \neg A \in \mathbf{T}$$

Consistency Theorem Proof

By definition of formal theory T, all tautologies of $\mathcal{L}_{SA}$ are provable on T, i.e. are in $\mathbf{T}$ and so

$$(((A \cap B) \Rightarrow C) \Rightarrow ((A \Rightarrow (B \Rightarrow C))) \in \mathbf{T}$$

and

$$(**) \quad ((A \cap \neg A) \Rightarrow C) \in \mathbf{T}$$

for all A, B, C of $\mathcal{L}_{SA}$

In particular, when $B = \neg A$ we get that

$$(***) \quad (((A \cap \neg A) \Rightarrow C) \Rightarrow ((A \Rightarrow (\neg A \Rightarrow C))) \in \mathbf{T}$$

Consistency Theorem Proof

Applying MP to $(**)$ and $(***)$ we get

$$((A \Rightarrow (\neg A \Rightarrow C))) \in \mathbf{T}$$

Applying MP twice to $(***)$ and $(*)$ we get that

$$C \in \mathbf{T} \quad \text{for all formulas } C$$

We proved **not** CT

This **ends** the proof of 2. and of the **Theorem**

The **Theorem** often serves a following definition of consistency

Consistency Definition

Definition

A theory T is **consistent** if and only if $\mathbf{T} \neq \mathcal{F}_{SA}$, i.e. there is A of $\mathcal{L}_{SA}$, such that

$$A \notin \mathbf{T}$$

The next important characterization of a formal theory T is the one of its **completeness** understood as the **ability** of proving or disapproving any of its statements, provided it is correctly formulated in its language $\mathcal{L}_{SA}$

Complete Theory

Definition

A theory $T = (\mathcal{L}, \mathcal{F}, LA, SA, \mathcal{R})$ is **complete**

if and only if

for **any** closed formula (sentence) A of the language $\mathcal{L}_{SA}$,

$$\vdash_T A \quad \text{or} \quad \vdash_T \neg A$$

We also write the above as

$$A \in \mathbf{T} \quad \text{or} \quad \neg A \in \mathbf{T}$$

Incomplete Theory

Definition

A theory T is **incomplete** if and only if **there is** a closed formula (sentence) A of the language $\mathcal{L}_{SA}$, such that

$$\nvdash_T A \quad \text{and} \quad \nvdash_T \neg A$$

We also write the above condition as

$$(*) \quad A \notin \mathbf{T} \quad \text{and} \quad \neg A \notin \mathbf{T}$$

Definition

Any sentence A with the property $(*)$ is called an **independent**, or **undecidable** sentence of the theory T

Gödel Theorems

The **incompleteness** definition says that in order to prove that a given theory T is **incomplete** we have to construct a sentence A of $\mathcal{L}_{SA}$ and be able to prove that **neither** A **nor** $\neg A$ has a proof in it

We are now almost ready to discuss Gödel Theorems

One of the most comprehensive development and proofs of Gödel Theorems can be found the Mendelson (1984) book

The Gödel Theorems chapter in Mendelson book is over 50 pages long, technically sound and beautiful

Gödel Theorems

We present here a short, high level approach **adopting** style of Smorynski's chapter in the

Handbook of Mathematical Logic, *Studies in Logic and Foundations of Mathematics, Volume 20 (1977)*

The chapter is over 40 pages long what seems to be a norm when one wants to prove Gödel's results

Smorynski's chapter is written in a very **condensed** and **general** way and concentrates on presentation of modern results

Gödel Theorems

We also want to bring to readers attention that the **introduction** to the Smorynski's chapter contains an excellent discussion of Hilbert Program and its relationship to Gödel's results

The chapter also provides an **explanation** why and how devastating Gödel Theorems were to the optimism reflected in Hilbert's Consistency and Conservation Programs

Hilbert's Conservation and Consistency Programs

Hilbert's Conservation and Consistency Programs

Hilbert proposed his Conservation and Consistency Programs as **response** to Brouwer and Weyl propagation of their **theory** that existence of Zermello's paradoxes free axiomatization of set theory makes the need for investigations into **consistency** of mathematics superfluous

Hilbert wrote:

" .... they (Brouwer and Weil) would **chop** *and* **mangle** *the science. If we would follow such a reform as the one they suggest, we would run the risk of losing a great part of our most valuable treasures! "*

Hilbert's Conservation Programs

Hilbert stated his **Conservation Program** as follows:

To **justify** *the use of abstract techniques he would show - by as simple and concrete a means as possible - that the* **use** *of abstract techniques was* **conservative** *- i.e. that any concrete assertion one could derive by means of such abstract techniques would be derivable without them*

Hilbert's Conservation Programs

We follow Smorynski's clarification of some of Hilbertian jargon whose exact meaning was never **defined** by Hilbert

We hence talk about **finitistically** meaningful statements and **finitistic** means of proof

By the **finitistically** meaningful statements we mean for example identities of the form

$$\forall x(f(x) = g(x))$$

where f, g are reasonably simple functions, for example primitive recursive

We will call them **real statements**

Hilbert's Conservation Programs

Finitistic proofs correspond to computations or combinatorial manipulations
More complicated statements are called **ideal** ones and, as such, have **no** meaning, but can be manipulated abstractly

The use of **ideal** statements and abstract reasoning about them would not allow one to derive any new **real** statements, i.e. none which were **not** already derivable

To **refute** Weyl and Brouwer, Hilbert required that his conservation property itself be **finitistically** provable

Hilbert's Consistency Programs

Hilbert's **Consistency Program** asks to devise a finitistic means of proving the consistency of various formal systems **encoding** abstract reasoning with **ideal** statements

The **Consistency Program** is a natural outgrowth and successor to the **Conservation Program**

There are two reasons for this

Hilbert's Consistency Programs

R1 Consistency is the assertion that some string of symbols **is not** provable

Since derivations are simple combinatorial manipulations, this is a **finitistically** meaningful and ought to have a **finitistic** proof

R2 Proving a consistency of a formal system **encoding** the abstract concepts already **establishes** the conservation **result**

Reason **R1** is straightforward

We will discuss **R2** as it is particularly important

Hilbert's Consistency Programs

Let's denote by **R** a formal system encoding **real** statements with their **finitistic** proofs

Denote by **I** the **ideal** system with its abstract reasoning

Let A be a **real** statement $\forall x(f(x) = g(x))$

Assume $\vdash_I A$

Then there is a derivation d of A in **I**

But, derivations are concrete objects and, for some **real** formula $P(x, y)$ **encoding** derivations in **I**,

$$\vdash_R P(d, \ulcorner A \urcorner)$$

where $\ulcorner A \urcorner$ is some **code** for A

Hilbert's Consistency Programs

Now, if A were **false**, one would have

$$f(a) \neq g(a)$$

for some a and hence

$$\vdash_R P(c, \ulcorner \neg A \urcorner)$$

for some c being a derivation of $\neg A$ in **I**

In fact, one would have a **stronger** assertion

$$\vdash_R (f(x) \neq g(x) \Rightarrow P(c_x, \ulcorner \neg A \urcorner))$$

for some c_x depending on x

Hilbert's Consistency Programs

But, if **R** proves consistency of **I**, we have

$$\vdash_R \neg(P(d, \ulcorner A \urcorner) \cap P(c, \ulcorner \neg A \urcorner))$$

whence $\vdash_R f(x) = g(x)$, with free variable x, i.e.

$$\vdash_R \forall x(f(x) = g(x))$$

To make the above argument rigorous, one has to define and explain the basics of **encoding**, develop the assumptions on the formula $P(x, y)$ and to **deliver** the whole argument in a formal rigorous way

Hilbert's Consistency Programs

To make the above argument rigorous, one also has to **develop** rigorously the whole apparatus **developed** originally by Gödel, which is **needed** for the proofs of his **theorems**

We **bring** it here at this stage because the above argument clearly invited Hilbert to establish his Consistency Program

Hilbert's Consistency Programs

Since Consistency Program was as **broad** as the general Conservation Program and, since it was more **tractable**, Hilbert **fixed** on it asserting:

"if the arbitrary given axioms **do not** *contradict each other through their consequences, then they are* **true***, then the objects defined through the axioms* **exist**

That, for me, is the criterion of **truth** *and* **existence***"*

Hilbert's Consistency Programs

The Consistency Program had as its **goal** the proof, by **finitistic** means of the consistence of strong systems

The solution would completely **justify** the use of abstract concepts and would **repudiate** Brouwer and Weyl

Gödel proved that it **couldn't** work

Gödel Incompleteness Theorems

Gödel Incompleteness Theorems

In 1931, while in his twenties, Kurt Gödel **announced** that Hilbert's **Consistency Program** could not be carried out

He had proved two theorems which gave a blow to the Hilbert's Program but on the other hand **changed** the face of mathematics establishing mathematical logic as strong and rapidly developing discipline

Loosely stated these **theorems** are as follows

Gödel Incompleteness Theorems

First Incompleteness Theorem

Let T be a formal theory containing arithmetic
Then **there is** a sentence A in the language of T which **asserts** its own unprovability and is such that:

(i) If T is consistent, then $\nvdash_T A$

(ii) If T is ω- consistent, then $\nvdash_T \neg A$

Gödel Incompleteness Theorems

Second Incompleteness Theorem

Let T be a consistent formal theory containing arithmetic

Then

$$\nvdash_T \ Con_T$$

where Con_T is the sentence in the language of T asserting the consistency of T

Observe that the **Second Incompleteness Theorem** destroys the Consistency Program

It **states** that **R** can't prove its own consistency, so obviously it can't prove consistency of **I**

Gödel Incompleteness Theorems

Smorynski's argument that the **First Incompleteness Theorem** destroys the Conservation Program is as follows

The Gödel sentence A is **real** and is easily seen to be **true**

It **asserts** its own unprovability and **is** indeed unprovable

Thus the Conservation Program **cannot** be carried out and, hence, the same must **hold** for the Consistency Program

Gödel Incompleteness Theorems

M. Detlefsen in the **Appendix** of his book

"Hilbert Program: An Essay on Mathematical Instrumentalism", Springer, 2013

argues that Smorynski's argument is **ambiguous**, as he doesn't tell us whether **it is** unprovability in **R** or unprovability in **I**

Gödel Incompleteness Theorems

We **recommend** to the reader interested a philosophical discussion of Hilbert Program to read this **Appendix**, if not the whole **book**

We will now formulate the **Incompleteness Theorems** in a more precise formal way and describe the main **ideas** behind their proofs

Arithmetization and Encoding

Arithmetization and Encoding

Observe that in order to formalize the **Incompleteness Theorems** one has **first** to "translate" the Gödel sentences A and Con_T into the language of T

For the **First Incompleteness Theorem** one needs to " translate " a self-referring sentence

"I am not provable in a theory T"

and for the **Second Incompleteness Theorem** one needs to
" translate " the self-referring sentence

"I am consistent"

Arithmetization and Encoding

The assumption in both theorems is that T contains arithmetic **means** usually it contains the Peano Arithmetic PA, or even its **sub-theory** RR called Robinson System

In this case the **final** product of such "translation" must be a sentence A or sentence Con_T of the language $\mathcal{L}_{PA}$ of PA, usually written as

$$\mathcal{L}_{PA} = \mathcal{L}(\{=\}, \{', +, \cdot\}, \{0\})$$

Arithmetization and Encoding

This ”translation” process into the **language** of some formal system containing arithmetic is called **arithmetization** and **encoding**, or just encoding for short

We **define** a notion of *arithmetization* as follows

An **arithmetization** of a theory T is a one-to-one function g from the **set** of symbols of the **language** of T, expressions (formulas) of T, and finite sequences of expressions of T (proofs) into the **set** of positive integers

Arithmetization and Encoding

The function g must **satisfy** the following conditions

(1) g is effectively computable

(2) there is an effective procedure that **determines** whether any given positive integer n is in the range of g and, if n is in the range of g, the procedure **finds** the object x such that $g(x) = n$

Arithmetization was originally devised by Gödel in 1931 in order to *arithmetize* Peano Arithmetic PA and **encode** the **arithmetization** process in PA in order to formulate and to prove his **Incompleteness Theorems**

Arithmetization and Encoding

Functions and relations whose arguments and values are natural numbers are called the **number-theoretic** functions and relations

In order to **arithmetize** and **encode** in a formal system we have to

1. **associate** numbers with symbols of the language of the system, **associate** numbers with expressions (formulas), and with sequences of expressions of the language of the system

This is **arithmetization** of basic syntax, and **encoding** of syntax in the system

Arithmetization and Encoding

2. **replace** assertions about the system by **number-theoretic** statements, and **express** these number-theoretic statements **within** the formal system itself

This is arithmetization and encoding in the system

We want the number - theoretic function to be **representable** in PA and we want the predicates to be **expressible** in PA, i.e. their characteristic functions to be **representable** in PA

Functions Representable in PA

The study of **representability** of functions in PA leads to the **class** of number-theoretic functions that turn out to be of great importance in mathematical logic, namely the **primitive recursive** and **recursive functions**

Their definition and study in a form of a **Recursion Theory** is an important field of mathematics and of computer science which developed out of the Gödel proof of the **Incompleteness Theorems**

Primitive Recursive and Recursive Functions

We **prove** that the class of recursive functions is **identical** with the class of functions representable in PA, i.e. we prove:

every **recursive** *function is representable in PA* and
every function representable in PA is **recursive**

The **representability** of primitive recursive and recursive functions in a formal system S in general and in PA in particular plays crucial role in the **encoding** process and consequently in the **proof** of Gödel Theorems

Arithmetization and Encoding

The details of **arithmetization** and **encoding** are as complicated and tedious as **fascinating** but are out of scope of our book

We recommend Mendelson's book:
Introduction to Mathematical Logic, Chapman & Hall (1997)
as the one with the **most** comprehensive and detailed presentation

Theories T and S

Principles of **Encoding** for T and S

Theories T and S

We **assume** at this moment that T is some fixed, but for a moment unspecified **consistent** formal theory

We also **assume** that **encoding** is done in some fixed theory S and that the theory T **contains** S, i.e. the language of T is an **extension** of the language of S and

$$\mathbf{S} \subseteq \mathbf{T}$$

i.e. for any formula A,

$$\text{if } \vdash_S A, \text{ then } \vdash_T A$$

Theories T and S

Moreover, we also **assume** that theories T and S **contain** as constants **only** numerals

$$\overline{0},\ \overline{1},\ \overline{2},\ \overline{3},\ \ldots,\ \ldots$$

and T **contains** infinitely countably many functional and predicate symbols

Usually S is taken to be a formal theory of arithmetic, but sometimes S can be a weak set theory

But in any case S always **contains** numerals

We also assume that theories T and S are such that the following Principles of Encoding hold

Principles of Encoding

The mechanics, conditions and details of **encoding** for T and S being Peano Arithmetic PA or its **sub-theory** Robinson Arithmetic RR are beautifully presented in the smallest **detail** in Mendelson's book

The Smorynski's approach we discuss here **covers** a larger class of formal theories and **uses** a more general and modern approach
We can't include all details but we are convinced that at this stage the reader will be able to **follow** Smorynski's chapter in the Encyclopedia

Principles of Encoding

Smorynski's chapter is very well and clearly written and is now **classical**

We wholeheartedly **recommend** it as a future reading

We also follow Smorynski's approach explaining **what** is to be encoded, **where** it is to be encoded, and which are the most **important** encoding and provability **conditions** needed for the proofs of the **Incompleteness Theorems**

Principles of Encoding

We first **encode** the syntax of T in S

Since **encoding** takes place in S, we assumed that it has a sufficient supply of constants, namely a countably infinite set of **numerals**

$$\overline{0},\ \overline{1},\ \overline{2},\ \overline{3},\ \ldots,\ \ldots$$

and closed terms to be used as **codes**

We assign to each formula A of the language of T a **closed term**

$$\ulcorner A \urcorner$$

called the **code** of A

Principles of Encoding

If $A(x)$ is a formula with a free variable x, then the **code**

$$\ulcorner A(x) \urcorner$$

is a **closed term** encoding the formula $A(x)$, with x viewed as a syntactic object and **not** as a parameter

We do it **recursively**

First we assign **codes** (unique closed terms from S) to its **basic** syntactic objects, i.e. elements of the **alphabet** of the language of T

Principles of Encoding

Terms and formulas are finite sequences of the **basic** syntactic objects and **derivations** (formal proofs) are also finite sequences of formulas

It means that S have to be able to **encode** and **manipulate** finite sequences

In the next **recursive step** we use for such **encoding** a class of primitive recursive functions and relations

We **assume** S admits a **representation** of the primitive recursive functions and relations and we **finish encoding** syntax

Principles of Encoding

S will also have to have certain important function symbols and we have to be able to **encode** them

1. S **must** have functional symbols

$$neg, \quad impl, \quad \ldots \text{ etc.}$$

corresponding to the logical connectives and quantifiers, such that, for all formulas A, B of the language of T

$$\vdash_S \; neg(\ulcorner A \urcorner) = \ulcorner \neg A \urcorner,$$

$$\vdash_S \; impl(\ulcorner A \urcorner, \ulcorner B \urcorner) = impl(\ulcorner A \Rightarrow B \urcorner), \quad \ldots \text{ etc.}$$

Principles of Encoding

An operation of **substitution** of a variable x in a formula A(x) by a term t is of a special **importance** in logic, so it **must** be represented in S, i.e.

2. S **must** have in a functional symbol sub that **represents** the substitution operator, such that for any formula $A(x)$ and term t with **codes**

$$\ulcorner A(x) \urcorner, \quad \ulcorner t \urcorner$$

respectively, we have that

$$\vdash_S \ sub(\ulcorner A(x) \urcorner, \ulcorner t \urcorner) = \ulcorner A(t) \urcorner$$

Principles of Encoding

Iteration of *sub* allows one to define

$$sub_3, \quad sub_4, \quad sub_5, \quad \ldots$$

such that

$$\vdash_S \; sub_n(\ulcorner A(x_1, \ldots, x_n) \urcorner, \ulcorner t_1 \urcorner, \ldots, \ulcorner t_n \urcorner) = \ulcorner A(t_1, \ldots, t_n) \urcorner$$

Finally, we have to **encode** derivations in S

To do so we proceed as follows

Principles of Encoding

3. S **must** have in a binary relation $Prov_T(x, y)$, such that for **closed terms** t_1, t_2

$\vdash_S$ $Prov_T(t_1, t_2)$ if and only if t_1 is a **code** of a derivation in T of the formula with a **code** t_2

We read $Prov_T(x, y)$ as *"x* **proves** *y in T"* or as *"x* **is** *a proof y in T"*

It follows that for some **closed term** t,

$$\vdash_T A \quad \text{if and only if} \quad \vdash_S Prov_T(t, \ulcorner A \urcorner)$$

Principles of Encoding

We **define**

$$Pr_T(y) \Leftrightarrow \exists x Prov_T(x, y)$$

and obtain a predicate **asserting** provability

However, it is not always **true**

$$\vdash_T A \text{ if and only if } \vdash_S Pr_T(\ulcorner A \urcorner)$$

unless S is fairly **sound** (to be defined separately)

The **encoding** can be carried out, however, in such a way that the following **conditions** essential to the proofs of the **Incompleteness Theorems** hold for any sentence A of T

Derivability Conditions

Derivability Conditions (Hilbert-Bernays, 1939)

For sentence A of T

D1 $\vdash_T A$ implies $\vdash_S Pr_T(\ulcorner A \urcorner)$

D2 $\vdash_S ((Pr_T(\ulcorner A \urcorner) \Rightarrow Pr_T(\ulcorner Pr_T(\ulcorner A \urcorner) \urcorner)))$

D3 $\vdash_S ((Pr_T(\ulcorner A \urcorner) \cap Pr_T(\ulcorner (A \Rightarrow B) \urcorner)) \Rightarrow Pr_T(\ulcorner B \urcorner))$

Chapter 11
Formal Theories and Gödel Theorems

Slides Set 2

PART 4: **Proof** of the Gödel **Incompleteness Theorems**

Diagonalization Lemma

The following theorem, called historically by the name **Diagonalization Lemma** is essential to the **proof** of the **Incompleteness Theorems**

It is also called **Fixed Point Theorem** and both names are used interchangeably

The fist name as is historically older, important for convenience of references and the second name is routinely **used** in computer science community

Diagonalization Lemma

Mendelson (1977) believes that the **central idea** was first explicitly mentioned by Carnap who pointed out in 1934 that the result was **implicit** in the work of Gödel (1931)

Gödel was **not aware** of Carnap work until 1937

The name **Diagonalization Lemma** is used because the main argument in its proof has some resemblance to the **diagonal** arguments used by Cantor in 1891

Diagonalization Lemma

In mathematics, a Fixed-point Theorem is a **name** of a theorem saying that a function f under some conditions, will have at least one fixed point, ı.e. a point x such that

$$f(x) = x$$

The **Diagonalization Lemma** says that for any formula A in the language of theory T with one free variable **there is** a sentence B such that the formula

$(B \Leftrightarrow A(\ulcorner B \urcorner))$ is provable in T

Diagonalization Lemma

Intuitively, the **Diagonalization Lemma** sentence B such that

$$\vdash_T \quad (B \Leftrightarrow A(\ulcorner B \urcorner))$$

is a **self-referential** sentence saying that B has property A

The sentence B can be viewed as a **fixed point** of the operation assigning to each formula A the sentence $A(\ulcorner B \urcorner)$

Hence the name **Fixed Point Theorem**.

Diagonalization Lemma

Let T, S be theories as defined

Let $A(x)$ be a formula in the language of T with x as the only free variable

Then **there is** a sentence B such that

$$\vdash_S \ (B \Leftrightarrow A(\ulcorner B \urcorner))$$

NOTE: If A, B are not in the language of S, then by $\vdash_S \ (B \Leftrightarrow A(\ulcorner B \urcorner))$ we mean that the equivalence is proved in the theory S' in the language of T whose only non-logical axioms are those of S

Proof of **Diagonalization Lemma**

Given $A(x)$, let the formula $(C(x) \Leftrightarrow A(sub(x,x))$

be a *diagonalization* of $A(x)$

Let $m = \ulcorner C(x) \urcorner$ and $B = C(m)$, i.e. $B = C(\ulcorner C(x) \urcorner)$

Then we claim

$$\vdash_S (B \Leftrightarrow A(\ulcorner B \urcorner))$$

For, in S, we see that

$B \Leftrightarrow C(m) \Leftrightarrow A(sub(m,m))$

$\Leftrightarrow A(sub(\ulcorner C(x) \urcorner, m)$ (since $m = \ulcorner C(x) \urcorner$)

$\Leftrightarrow A(\ulcorner C(m) \urcorner) \Leftrightarrow A(\ulcorner B \urcorner)$

by sub definition and $B = C(m)$

This **proves** (we leave details to the reader as exercise)

$$\vdash_S (B \Leftrightarrow A(\ulcorner B \urcorner))$$

First Incompleteness Theorem

First Incompleteness Theorem

Let T, S be theories as defined

Then **there is** a sentence G in the language of T such that:

(i) $\nvdash_T \ G$

(ii) under an additional assumption, $\nvdash_T \ \neg G$

Proof

We apply **Diagonalization Lemma** for a formula $A(x)$ being $\neg Pr_T(x)$, where $Pr_T(x)$ is defined as

$$Pr_T(x) \ \Leftrightarrow \ \exists y Prov_T(y, \ x)$$

and $Prov_T(y, \ x)$ reads as "y **is** *a proof* x *in* T "

We get that **there is** a sentence G such that

$$\vdash_S \ (G \Leftrightarrow \ \neg Pr_T(\ulcorner G \urcorner))$$

Proof of First Incompleteness Theorem

We have assumed about theories T, S that T is **consistent** and $\mathbf{S} \subseteq \mathbf{T}$, i.e. for any formula A,

$$\text{if } \vdash_S A, \text{ then } \vdash_T A$$

So we have that also

$$(*) \quad \vdash_T (G \Leftrightarrow \neg Pr_T(\ulcorner G \urcorner))$$

Now we are ready to prove (i)

We conduct the proof of (i) $\nvdash_T G$ by **contradiction**

Assume

$$\vdash_T G$$

Proof of First Incompleteness Theorem

Observe that by the Derivability Condition **D1**: $\vdash_T A$ implies $\vdash_S Pr_T(\ulcorner A \urcorner)$ for $A = G$ we get that

$$\vdash_T G \quad \text{implies} \quad \vdash_S Pr_T(\ulcorner G \urcorner)$$

Hence by assumption $\vdash_T G$ we get

$$\vdash_S Pr_T(\ulcorner G \urcorner)$$

By the assumption $\mathbf{S} \subseteq \mathbf{T}$ we get

$$\vdash_T Pr_T(\ulcorner G \urcorner)$$

This, the assumption $\vdash_T G$, and already proved

$$(*) \quad \vdash_T (G \Leftrightarrow \neg Pr_T(\ulcorner G \urcorner))$$

contradicts the consistency of T

Proof of First Incompleteness Theorem

Now we are ready to prove

(ii) under an additional assumption, $\nvdash_T \neg G$

The *additional assumption* is a **strengthening** of the converse implication to **D1** namely,

$$\vdash_T Pr_T(\ulcorner G \urcorner) \quad \text{implies} \quad \vdash_T G$$

We conduct the proof by **contradiction**

Assume $\vdash_T \neg G$

Hence $\vdash_T \neg\neg Pr_T(\ulcorner G \urcorner))$ so we have that $\vdash_T Pr_T(\ulcorner G \urcorner))$

By the *additional assumption* it implies that $\vdash_T G$ what

contradicts the consistency of T

This ends the **proof**

First Incompleteness Theorem

Observe that the sentence G is **equivalent** in T to an assertion that G is **unprovable** in T

In other words the sentence G says

" I am not provable in T"

Hence the just proved **Second Incompleteness Theorem** provides a strict mathematical formalization of its previously intuitively stated version that said:

" **there is** *a sentence A in the language of T which asserts its own unprovability* "

We call G the Gödel's sentence

Second Incompleteness Theorem

Second Incompleteness Theorem

Let T, S be theories as defined

Let Con_T be a sentence $\neg Pr_T(\ulcorner C \urcorner))$, where is C is any contradictory statement

Then

$$\nvdash_T \; Con_T$$

Proof

Let G the Gödel's sentence of the First Incompleteness Theorem. We prove that

$$\vdash_T \; (Con_T \Leftrightarrow G)$$

and use it to prove that

$$\nvdash_T \; Con_T$$

Proof of Second Incompleteness Theorem

Assume that we have already proved the property

$$(*) \quad \vdash_T \ (Con_T \Leftrightarrow G)$$

We conduct the proof of

$$\nvdash_T \ Con_T$$

by **contradiction**

Assume $\vdash_T \ Con_T$

By $(*)$ we have that $\vdash_T \ (Con_T \Leftrightarrow G)$, so by the assumption we get $\vdash_T \ G$ what **contradicts** the First Incompleteness Theorem.

Proof of Second Incompleteness Theorem

To complete the **proof** we have to prove now the property

$$(*) \quad \vdash_T \; (Con_T \Leftrightarrow G)$$

In the proof of $(*)$ we use some logic facts, called **Logic 1, 2, 3, 4** that are listed and proved after this **proof**

We know by **Logic 1** that

$$\vdash_T \; (Con_T \Leftrightarrow G)$$

if and only if

$$\vdash_T \; (Con_T \Rightarrow G) \quad \text{and} \quad \vdash_T \; (G \Rightarrow Con_T)$$

Proof of Second Incompleteness Theorem

1. We prove the implication

$$\vdash_T \ (G \Rightarrow Con_T)$$

By definition of Con_T we have to prove now

$$\vdash_T \ (G \Rightarrow \neg Pr_T(\ulcorner C \urcorner))$$

The formula C is a contradiction, so the formula $(C \Rightarrow G)$ is a predicate tautology

Hence

$$\vdash_T \ (C \Rightarrow G)$$

By the Derivability Condition **D1**: $\vdash_T \ A$ implies $\vdash_S \ Pr_T(\ulcorner A \urcorner)$ for $A = (C \Rightarrow G)$ we get that

$$\vdash_S \ Pr_T(\ulcorner (C \Rightarrow G) \urcorner)$$

Proof of Second Incompleteness Theorem

We write **D3** for $A = Pr_T(\ulcorner C \urcorner)$ and $B = \vdash_S Pr_T(\ulcorner (C \Rightarrow G) \urcorner)$ and obtain that

$$(*) \quad \vdash_S ((Pr_T(\ulcorner C \urcorner) \cap Pr_T(\ulcorner (C \Rightarrow G) \urcorner)) \Rightarrow Pr_T(\ulcorner G \urcorner))$$

We have by **Logic 2**

$$(**) \quad \vdash_S (Pr_T(\ulcorner C \urcorner) \Rightarrow (Pr_T(\ulcorner C \urcorner) \cap Pr_T(\ulcorner (C \Rightarrow G) \urcorner)))$$

We get from $(*)$, $(**)$, and **Logic 3**

$$\vdash_S (Pr_T(\ulcorner C \urcorner) \Rightarrow Pr_T(\ulcorner G \urcorner))$$

We apply **Logic 4** (contraposition) to the above and get

$$(***) \quad \vdash_S (\neg Pr_T(\ulcorner G \urcorner) \Rightarrow \neg Pr_T(\ulcorner C \urcorner))$$

Proof of Second Incompleteness Theorem

Observe that by the property $\vdash_S (G \Leftrightarrow \neg Pr_T(\ulcorner G \urcorner))$ proved in in the proof of the **First Incompleteness Theorem** we have

$$\vdash_S (G \Rightarrow \neg Pr_T(\ulcorner G \urcorner))$$

We put $(* * *)$ and the property above the together and get

$$\vdash_S (G \Rightarrow \neg Pr_T(\ulcorner G \urcorner)) \quad \text{and} \quad \vdash_S (\neg Pr_T(\ulcorner G \urcorner) \Rightarrow \neg Pr_T(\ulcorner C \urcorner))$$

Applying **Logic 4** to the above we get

$$\vdash_S (G \Rightarrow \neg Pr_T(\ulcorner C \urcorner))$$

But C is by definition Con_T and hence we have **proved** the

$$\vdash_S (G \Rightarrow Con_T)$$

and hence also

$$\vdash_T (G \Rightarrow Con_T)$$

Proof of Second Incompleteness Theorem

2 . We prove now $\vdash_T \ (Con_T \Rightarrow G)$, i.e. the implication

$$\vdash_T \ (\neg Pr_T(\ulcorner C \urcorner) \Rightarrow G)$$

Here is a concise **proof**

We leave it to the reader as an exercise to write a detailed version that **develops** and lists needed **Logic** properties in a similar way as we did in the part **1.**

By the Derivability Condition **D2** for $A = G$ we get

$$\vdash_S \ ((Pr_T(\ulcorner G \urcorner) \Rightarrow Pr_T(\ulcorner Pr_T(\ulcorner G \urcorner) \urcorner)))$$

Proof of Second Incompleteness Theorem

The property $\vdash_S \ ((Pr_T(\ulcorner G \urcorner) \Rightarrow Pr_T(\ulcorner Pr_T(\ulcorner G \urcorner) \urcorner)))$ implies

$$\vdash_S \ (Pr_T(\ulcorner G \urcorner) \Rightarrow Pr_T(\ulcorner \neg G \urcorner))$$

by **D1**, **D3**, since $\vdash_S \ (G \Rightarrow \ \neg Pr_T(\ulcorner G \urcorner))$

This yields

$$\vdash_S \ ((Pr_T(\ulcorner G \urcorner) \Rightarrow Pr_T(\ulcorner (G \cap \neg G) \urcorner))$$

by **D1**, **D3**, and logic properties

This in turn implies

$$\vdash_S \ ((Pr_T(\ulcorner G \urcorner) \Rightarrow Pr_T(\ulcorner C \urcorner))$$

by again **D1**, **D3**, and logic properties

Proof of Second Incompleteness Theorem

By **Logic 4** (contraposition) we get

$$\vdash_S \ (\neg Pr_T(\ulcorner G \urcorner) \Rightarrow \neg Pr_T(\ulcorner C \urcorner))$$

which is

$$\vdash_S \ (Con_T \Rightarrow G)$$

and hence by assumption **S** $\subseteq$ **T** we get that also

$$\vdash_T \ (Con_T \Rightarrow G)$$

This ends the **proof**

Second Incompleteness Theorem

Observation

We proved, a part of proof of the **Second Incompleteness Theorem** the equivalence

$$\vdash_T \ (Con_T \Leftrightarrow G)$$

which says that the **self-referential** Gödel sentence G which asserts its own unprovability is **equivalent** to the sentence asserting consistency

Hence, the sentence G **is unique** up to provable equivalence $(Con_T \Leftrightarrow G)$ and we can say that G is **the sentence** that asserts its own unprovability

ω-consistency

We used, in the part (ii) of the **First Incompleteness Theorem**, an additional assumption that $\vdash_T \; Pr_T(\ulcorner G \urcorner)$ implies $\vdash_T \; G$, instead of a **habitual** assumption of ω-consistency

The concept of ω-consistency was introduced by Gödel for purpose of stating assumption **needed** for the proof of his **First Incompleteness Theorem**

The modern researchers **proved** that the assuption of the ω-consistency can be **replaced**, as we did, by other more general better suited for new proofs **conditions**

ω - consistency

Informally, we say that T is ω- consistent if the following two conditions are not simultaneously satisfied for any formula A:

(i) $\vdash_T \exists x A(x)$

(ii) $\vdash_T \neg A(\overline{n})$ for every natural number n

Formally, ω-consistency can be **represented** in varying degrees of generality by (modification of) the following formula

$$(Pr_T\,(\ulcorner \exists x A(x) \urcorner) \Rightarrow \exists x\, \neg Pr_T(\ulcorner \neg A(x) \urcorner))$$

Logic Properties

Logic Properties

We **prove** now, as an exercise the logic based steps in the **proof** of part **1.** of the proof the **Second Incompleteness Theorem** that follow the predicate logic properties, hence we named them **Logic**

The **discovery** and formalization of needed logic properties and their proofs for the part **2.** is left as a homework exercise

Logic Properties

Remark

All formulas belonging to the languages of of T, S belong to the language of H

By the monotonicity of classical consequence everything provable in T, S is provable in H

By definition of T, S, they are based on a **complete** proof system H for predicate logic and so all predicate **tautologies** are provable in H

In particular, all predicate **tautologies** formulated on the languages of T, S are **provable** in T and in S, respectively

Logic 1

Given a complete proof system H, for any formulas A, B of the language of H,

$$\vdash (A \Leftrightarrow B) \text{ if and only if } \vdash (A \Rightarrow B) \text{ and } \vdash (B \Rightarrow A)$$

Proof

1. We prove implication

$$\text{if } \vdash (A \Leftrightarrow B), \quad \text{then } \vdash (A \Rightarrow B) \text{ and } \vdash (B \Rightarrow A)$$

Directly from provability of a tautology

$$((A \Leftrightarrow B) \Rightarrow ((A \Rightarrow B) \cap (B \Rightarrow A)))$$

and assumption $\vdash (A \Leftrightarrow B)$, and MP we get

$$\vdash ((A \Rightarrow B) \cap (B \Rightarrow A))$$

Logic Properties

Consequently, from

$$\vdash ((A \Rightarrow B) \cap (B \Rightarrow A))$$

and provability of tautologies $((A \cap B) \Rightarrow A)$ and $((A \cap B) \Rightarrow B)$, for any formulas A, B, i.e. from fact that in a particular case

$$\vdash \ (((A \Rightarrow B) \cap (B \Rightarrow A) \Rightarrow (A \Rightarrow B))$$

and

$$\vdash \ (((A \Rightarrow B) \cap (B \Rightarrow A) \Rightarrow (B \Rightarrow A))$$

and MP applied twice we get

$$\vdash (A \Rightarrow B) \quad \text{and} \quad \vdash (B \Rightarrow A)$$

Logic Properties

2. We prove implication

$$\text{if } \vdash (A \Rightarrow B) \text{ and } \vdash (B \Rightarrow A), \text{ then } \vdash (A \Leftrightarrow B)$$

Directly from provability of tautology

$$\vdash ((A \Rightarrow B) \Rightarrow ((B \Rightarrow A) \Rightarrow (A \Leftrightarrow B)))$$

and assumptions

$$\vdash (A \Rightarrow B) \text{ and } \vdash (B \Rightarrow A)$$

and MP applied twice we get

$$\vdash (A \Leftrightarrow B)$$

Logic 2

For any formulas A, B of the language of H,

$$\vdash (A \Rightarrow (A \cup B)) \quad \text{and} \quad \vdash (A \Rightarrow (B \cup A))$$

Proof

Follows directly from predicate **tautologies**

$$(A \Rightarrow (A \cup B)) \quad \text{and} \quad (A \Rightarrow (B \cup A))$$

and **completeness** of H

Logic 3

For any formulas A, B of the language of H,

$$\text{if } \vdash (A \Rightarrow B) \quad \text{and} \quad \vdash (B \Rightarrow C), \quad \text{then} \quad \vdash (A \Rightarrow C)$$

Proof

From completeness of H and predicate tautology we get

$$(*) \quad \vdash ((A \Rightarrow B) \Rightarrow ((B \Rightarrow C) \Rightarrow (A \Rightarrow C)))$$

Assume $\vdash (A \Rightarrow B)$ and $\vdash (B \Rightarrow C)$

Applying MP to $(*)$ twice we get the proof of $(A \Rightarrow C)$, i.e.

$$\vdash (A \Rightarrow C)$$

Logic Properties

Logic 4

For any formulas A, B of the language of H,

$$\vdash (A \Rightarrow B) \quad \text{if and only if} \quad \vdash (\neg B \Rightarrow \neg A)$$

Proof

From completeness of H, predicate tautology, and **Logic 1**

$$\vdash ((A \Rightarrow B) \Leftrightarrow (\neg B \Rightarrow \neg A))$$

if and only if

$(*)$ $\vdash ((A \Rightarrow B) \Rightarrow (\neg B \Rightarrow \neg A))$ and $\vdash ((\neg B \Rightarrow \neg A) \Rightarrow (A \Rightarrow B))$

Logic Properties

Assume $\vdash (A \Rightarrow B)$

By $(*)$ and MP we get

$$\vdash (\neg B \Rightarrow \neg A)$$

Assume $\vdash (\neg B \Rightarrow \neg A)$

By $(*)$ and MP we get

$$(A \Rightarrow B)$$

This **ends** the proof

The Formalized Completeness Theorem

The Formalized Completeness Theorem - Introduction

Proving completeness of a proof system with respect to a given semantics is the **first** and most important **goal** while developing a logic and was the central focus of our study

So we now **conclude** our book with presentation the **formalized completeness theorem**

We discuss its proof and show how to use it to give new type of proofs, called model-theoretic proofs, of the **incompleteness theorems** for Peano Arithmetic PA, i.e. for the case when S= PA

The Formalized Completeness Theorem

Formalizing the proof of **completeness theorem** for classical predicate logic from chapter 9 within PA we get the following

Hilbert-Bernays Completeness Theorem

Let U be a theory with a primitive recursive set of axioms
There is a set Tr_M of formulas such that in $PA + Con_U$ one can **prove** that this set Tr_M **defines** a model M of U:

$$\vdash_{PA+Con_U} \forall x(Pr_U(x) \Rightarrow Tr_M(x))$$

Moreover, the set Tr_M is of type Δ_2

The Formalized Completeness Theorem

The Hilbert-Bernays **Completeness Theorem** asserts that modulo Con_U, one can **prove** in PA the **existence** of a **model** of U whose truth definition is of type Δ_2

The **proof** of the **Completeness Theorem** is just an arithmetization of the Henkin proof presented in chapter 9

The **proof** proceeds as follows

The Formalized Completeness Theorem

Following the Henkin proof one **adds** to the language of U an infinite primitive recursive set of new constants

$$c_0,\ c_1,\ c_2\ \ldots,\ldots$$

Then one **adds** for **each** formula $A(x)$ the corresponding Henkin Axiom

$$(\exists x A(x) \Rightarrow A(c_{A[x]}))$$

and enumerates sentences

$$A_0,\ A_1,\ A_2,\ \ldots,\ldots$$

in this augmented language

The Formalized Completeness Theorem

As next step one defines a **complete theory** by starting with U and **adding** at each step n a sentence

$$A_n, \quad \text{or} \quad \neg A_n$$

according to whether A_n **is** consistent with what has been chosen before **or not**

The construction is then **described** within PA

Assuming Con_U one can also **prove** that the construction never terminates

The Formalized Completeness Theorem

The resulting set of sentences forms a **complete theory** which by Henking Axioms forms a **model** of U

Inspection shows that the **truth definition** Tr_M is of type Δ_2

This **ends** the proof

The Hilbert-Bernays **completeness** makes possible to **conduct** new type of proofs of the Gödel **incompleteness theorems**, model- theoretic proofs

The Formalized Completeness Theorem

Gödel chose as the self-referring sentence a **syntactic** statement

" I do not have a proof"

He did not **want** (and saw difficulties with) to use the sentence involving the notion of truth, i.e. the sentence

"I am not true"

The new proofs use exactly this **semantic** statement and this is why they are called model-theoretic proofs

Model-theoretic Proof

Dana Scott was the first to **observe** that one can give a model- theoretic proof of the **First Incompleteness Theorem**

Here is the **theorem** and its Dana Scott's short **proof**

First Incompleteness Theorem

Let PA be a Peano Arithmetic

There is a sentence G of PA, such that

(i) $\nvdash_{PA} \; G$

(ii) $\nvdash_{PA} \; \neg G$

Model-theoretic Proof

Proof

Assume PA is **complete**

Then, since PA is **true**,

$$\vdash_{PA} Con_{PA}$$

and we can apply the Hilbert-Bernays **Completeness Theorem** to obtain a formula Tr_M which gives a truth definition for the **model** of PA

We choose G by

$$(*) \quad \vdash_{PA} (G \Leftrightarrow \neg Tr_M(\ulcorner G \urcorner))$$

Model-theoretic Proof

We claim

$$\nvdash_{PA} \ G \quad \text{and} \quad \nvdash_{PA} \ \neg G$$

For if $\vdash_{PA} \ G$, then $\vdash_{PA} \ Tr_M(\ulcorner G \urcorner))$

By $(*)$ and logic properties we get $\vdash_{PA} \ \neg G$

Contradiction

Similarly, $\vdash_{PA} \neg G$ implies $\vdash_{PA} \ G$

This **ends** the proof

Observe that the sentence G as defined by $(*)$ **asserts**

"I am not true"

G Sentences

Scott 's proof **differs** from the Gödel proof **not only** by the choice of the model- theoretic method, **but also** by be a choice of the model- theoretic sentence G

Let's compare these two **independent** sentences G: the classic **syntactic** one of Gödel proof representing statement

" I do not have a proof"

and the **semantic** one of Scott proof representing statement

"I am not true"

G- Sentences Property

The sentence G_S of the Gödel **Incompleteness Theorem** asserting its own unprovability is

(i) **unique** up to provable equivalence $(Con_T \Leftrightarrow G)$

(ii) the sentence is Π_1 and hence **true**

The sentence G of the Scott **Incompleteness Theorem** asserting its own falsity in the model constructed is

(iii) **not unique** - for the following implication holds

$$\text{if } (G \Leftrightarrow \neg Tr_M(\ulcorner G \urcorner)), \text{ then } (\neg G \Leftrightarrow \neg Tr_M(\ulcorner \neg G \urcorner))$$

(iv) the sentence is Δ_2 and, by (iii) there **is no** obvious way of deciding its **truth** or falsity

Model-theoretic Proof

Georg Kreisler was the **first** to present a model- theoretic proof of the following

Second Incompleteness Theorem

Let PA be a Peano Arithmetic

$$\nvdash_{PA} \quad Con_{PA}$$

The **proof** uses, as did the proof of the Hilbert-Bernays Completeness Theorem the arithmetization of Henkin proof of completeness theorem presented in Chapter 9

Model-theoretic Proof

Proof

The proof is carried by **contradiction**

We assume

$$\vdash_{PA} \quad Con_{PA}$$

Then we show, for any presentation of the Henkin proof of completeness theorem construction (as given by encoding, the enumeration of sentences ...etc.) **there is** a number m, such that, for any **model** N of PA, the sequence of models **determined** by the given presentations must **stop** after fewer then m steps with a **model** in which

Con_{PA} is **false**