AMBIGUOUS METALOGICAL TERMS

	Argument	Sentence	Proof procedure	Theory (Formal system)
'Valid'	An **argument** is *valid* exactly if it is necessary that if all of its premises are true then its conclusion is also true.	A **sentence** is *valid* exactly if it is true in every interpretation.		
'Sound'	An **argument** is *sound* exactly if it is valid and all of its premises are true.		A **proof procedure** is *sound* exactly if every sequent that is derivable according to the procedure is secure.	
'Complete', 'Incomplete'			A **proof procedure** is *complete* exactly if every secure sequent is derivable according to the procedure.	A **theory** is *complete* exactly if, for every sentence of its language, either the sentence or its negation is in the theory.
'Decidable', 'Undecidable'		A **sentence** is *undecidable in* (or *by* or *for*) *a theory* exactly if neither the sentence nor its negation is in the theory.		A **theory** is *decidable* exactly if there is an effective procedure for determining, for every sentence, whether or not it is in the theory.

INTRODUCTION TO LOGIC

I. SENTENCES VERSUS PROPOSITIONS

1. Different sentence-tokens can belong to the same sentence-type: I say "Good morning" and you say "Good morning".

2. A sentence is always a sentence of some (natural or formal) language: "Καλημέρα".

3. Different sentences can express the same proposition: "Good morning" and "Καλημέρα".

4. Propositions are true or false, but sentences are not true or false: "He is married" is not true or false.

II. ARGUMENTS VERSUS ARGUMENT FORMS

1. An *argument* is an ordered pair whose first member is a set of propositions (the *premises* of the argument) and whose second member is a proposition (the *conclusion* of the argument).

2. An *argument form* is an ordered pair whose first member is a set of sentences in some *formal* language and whose second member is a sentence in that language.

3. An argument *instantiates* an argument form exactly if (roughly) the sentences which constitute the argument form can express the propositions which constitute the argument.

4. An argument in general instantiates more than one argument form:

Sam and Jill are parents		Q & R		Ps & Pj		$\exists xPsx$ & $\exists xPjx$
----------	instantiates	--------	but also	---------	and	----------
Jill is a parent		R		Pj		$\exists xPjx$

III. VALIDITY VERSUS LOGICAL VALIDITY

1. An **argument** is *valid* exactly if it is necessary that if all of its premises are true then its conclusion is also true.

2. An **argument form** is *valid* exactly if _____. Filling in the blank is one main object of logic. There are two ways to fill in the blank: a *syntactic* way, corresponding to the notion of *derivability*, and a *semantic* way, corresponding to the notion of *logical consequence*.

3. Important definition: An argument is *logically valid* exactly if it instantiates **at least one** valid argument form.

	Smith drank water		Smith and Jones are male
Compare	----------	with	----------
	Smith drank H_2O		Smith is male

4. An argument can be logically valid even if it instantiates some invalid argument form:

Every human is mortal and Socrates is human		Q & R
----------	instantiates	--------, an invalid
Socrates is mortal		S

argument form, but is logically valid because it *also* instantiates some valid argument form.

5. We have no good method for proving the logical *invalidity* (as opposed to the logical *validity*) of arguments (as opposed to proving the invalidity of argument forms): it is not enough to show that the argument instantiates many invalid argument forms, because the argument may *also* instantiate some *other*, valid argument form and thus be logically valid.

SYNTAX FOR FIRST-ORDER LOGIC

I. SYMBOLS AND LANGUAGES

1. Logical symbols

(a) *Connective symbols*:
- Tilde: ~ ('not')
- Ampersand: & ('and')
- Wedge: ∨ ('or')
- Arrow: → ('only if')
- Double arrow: ↔ ('exactly if')

(b) *Variables*: *x*, *y*, *z*, ...

(c) *Identity symbol*: = [not always present]

(d) *Quantifier symbols*:
- Inverted ay: ∀ ('for every')
- Reversed ee: ∃ ('for at least one')

(e) *Punctuation symbols*:
- Left parenthesis: (
- Right parenthesis:)
- Comma: ,

2. Nonlogical symbols (no symbol is of more than one kind)

(a) *Constants (individual symbols)*: *a*, *b*, *c*, ...

(b) *Predicates (relation symbols)*: *P*, *Q*, *R*, ...

(c) *Function symbols*: ′ (accent), + (plus sign), • (times sign), ...

3. A *language* is an enumerable [i.e., finite or denumerable] set of nonlogical symbols. The *language of arithmetic*, *L**, is {0, <, ′, +, •}.

II. TERMS

1. An *atomic term* is a variable or a constant.

2. A *term* is either an atomic term or any string of symbols that can be built up from atomic terms in a sequence of finitely many steps—called a *formation sequence*—by applying the rule: If *f* is an *n*-place function symbol and t_1, t_2, ..., t_n are terms, then $f(t_1, ..., t_n)$ is a term.

3. A *closed* term is a term that contains no variable, and an *open* term is a term that contains at least one variable.

III. FORMULAS

1. An *atomic formula* is a string of symbols (e.g., '$R(t_1, ..., t_n)$') consisting of an *n*-place predicate (including the identity symbol), followed by '(', followed by *n* terms separated by commas, followed by ')'.

2. A *formula* is either an atomic formula or any string of symbols that can be built up from atomic formulas in a sequence of finitely many steps—called a *formation sequence*—by applying the rules: (a) If *F* is a formula, its *negation* ~*F* is also a formula. (b) If *F* and *G* are formulas, their *conjunction* (*F* & *G*) is also a formula, and so is their *disjunction* (*F* ∨ *G*). (c) If *F* is a formula and *x* is a variable, the *universal quantification* ∀*xF* and the *existential quantification* ∃*xF* are formulas. (*F* is the *scope* of ∀ or of ∃.)

3. A *subformula* of a given formula is any string of consecutive symbols within the given formula which is itself a formula. (Similarly for *subterm*.)

4. An *occurrence* of a variable *x in a formula* is *bound* exactly if it is part of a subformula beginning with '∀*x*' or '∃*x*'; otherwise, the occurrence of the variable is *free* in the formula.

5. An *instance* of a formula *F*(*x*) [in which *x* is the only *free variable* (i.e., the only variable having at *least one* free occurrence)] is any formula of the form *F*(*t*) [in which *t* is substituted for all free occurrences of *x* in *F*] for *t* a closed term.

6. A *sentence* is a formula in which there is no free (occurrence of any) variable. A *subsentence* of a given sentence is any subformula of the given sentence which is itself a sentence.

IV. OFFICIAL AND UNOFFICIAL NOTATION

Official notation	Unofficial notation
$<(x, y)$	$x<y$
$(F \,\&\, G)$	$F \,\&\, G$
$(F \,\&\, (G \,\&\, H))$	$F \,\&\, G \,\&\, H$
$(\sim F \vee G)$	$F \rightarrow G$
$((\sim F \vee G) \,\&\, (\sim G \vee F))$	$F \leftrightarrow G$
$=(x, y)$	$x=y$
$\sim=(x, y)$	$x\neq y$
$'(x)$	x'
$+(x, y)$	$x+y$
$\bullet(x, y)$	$x\bullet y$
$+(x, \bullet(y, z))$	$x+y\bullet z$
$'(0)$	$0'$ or 1
$'('(0))$	$0''$ or 2
$P(a)$	Pa
$P(x, y)$	Pxy

SEMANTICS FOR FIRST-ORDER LOGIC

I. INTERPRETATIONS (OF LANGUAGES)

1. Informally, an interpretation of a language is a way of specifying which propositions the sentences of the language express.

2. Formally, an *interpretation* $\mathcal{M}$ *for (or of) a language* L is an ordered pair $\langle|\mathcal{M}|, f\rangle$, where:

(a) $|\mathcal{M}|$ is a nonempty set, the *domain* or *universe of discourse* of the interpretation (the set of things the interpretation takes the language to be about).

(b) f is a function assigning to each member (i.e., nonlogical symbol) of L a *denotation* as follows:

(i) The denotation of a *constant* is a *member* of the domain.

(ii) The denotation of a *one-place predicate* is a *subset* of the domain.

(iii) The denotation of an *n-place predicate* is an *n-place relation* on the domain (i.e., a set of ordered n-tuples of members of the domain).

(iv) The denotation of an *n-place function symbol* is a *total n-argument function* on the domain.

[(v) The denotation of the *identity symbol* is the *relation of identity* on the domain.]

II. TRUTH OF A SENTENCE IN AN INTERPRETATION

Notation: '$\mathcal{M} \models S$' abbreviates 'S is true *in* (or *on* or *under*) interpretation $\mathcal{M}$'. (S must be a sentence.)

0. The denotation of a closed term $f(t_1, ..., t_n)$ is the value of the function that f denotes when the arguments are what $t_1, ..., t_n$ denote.

1. $\mathcal{M} \models R(t_1, ..., t_n)$ exactly if the relation that R denotes holds between the individuals denoted by $t_1, ..., t_n$.

2. $\mathcal{M} \models \sim F$ exactly if it is not the case that $\mathcal{M} \models F$.

3. $\mathcal{M} \models (F \,\&\, G)$ exactly if both $\mathcal{M} \models F$ and $\mathcal{M} \models G$.

4. $\mathcal{M} \models (F \vee G)$ exactly if either $\mathcal{M} \models F$ or $\mathcal{M} \models G$ (or both).

5. $\mathcal{M} \models =(t_1, t_2)$ exactly if t_1 and t_2 denote the same member of the domain.

6. $\mathcal{M} \models \forall x F(x)$ exactly if every member of the domain **satisfies** $F(x)$.

7. $\mathcal{M} \models \exists x F(x)$ exactly if at least one member of the domain **satisfies** $F(x)$.

8. If $\mathcal{M} \models F$, then $\mathcal{M} \models \forall x F$ and $\mathcal{M} \models \exists x F$ for any variable x.

A member m of the domain **satisfies** $F(x)$ (abbreviation: $\mathcal{M} \models F[m]$) exactly if $F(t)$ is true for some term t denoting m, or, if no such term exists, when one extends the language by adding a new constant c and one extends the interpretation by letting c denote m, in that extended interpretation $F(c)$ is true. (One may need to thus extend the language on a case-by-case basis because the domain may be nonenumerable but a language is enumerable and thus may not have enough terms to denote every member of the domain.)

SEMANTIC METALOGICAL NOTIONS

I. OBJECT LANGUAGES VERSUS METALANGUAGES

1. An *object language* is any (formal) language as already defined: an enumerable set of nonlogical symbols.

2. A *metalanguage* for an object language is a language used to talk *about* the object language. A metalanguage for an object language usually (but not always) differs from the object language. A metalanguage can be a formal language but is usually a natural language.

II. IMPLICATION, VALIDITY, AND (UN)SATISFIABILITY

1. A set of sentences Γ *implies* or has as a *consequence* a sentence *D* exactly if every interpretation (of a language in which both *D* and every member of Γ are sentences) that makes every sentence in Γ true makes *D* true (equivalently: no interpretation makes true both ~*D* and every sentence in Γ).

2. A sentence *D* is *valid* exactly if it is true in every interpretation (equivalently: it is false in no interpretation).

3. A set of sentences Γ is *unsatisfiable* exactly if no interpretation makes (every sentence in) Γ true, and is *satisfiable* otherwise (i.e., exactly if some interpretation makes Γ true).

4. Proposition: *D* is valid exactly if *every* Γ implies *D*, and Γ is unsatisfiable exactly if Γ implies *every D*.

III. EQUIVALENCE

1. Two **sentences** are *equivalent over a given interpretation* exactly if they have the same truth value in that interpretation.

2. Two **sentences** are (*logically*) *equivalent* exactly if they are equivalent over all interpretations.

3. Two **formulas** $F(x)$ and $G(x)$ are *equivalent over a given interpretation* exactly if, for any constant c occurring in neither formula, the sentences $F(c)$ and $G(c)$ are equivalent over every interpretation that extends the given interpretation by providing some denotation for c.

4. Two **formulas** $F(x)$ and $G(x)$ are *(logically) equivalent* exactly if, for any constant c occurring in neither formula, the sentences $F(c)$ and $G(c)$ are logically equivalent. Equivalently: two **formulas** are *(logically) equivalent* exactly if they are equivalent over all interpretations.

SYNTACTIC METALOGICAL NOTIONS

I. A UNIFICATION OF SEMANTIC METALOGICAL NOTIONS

1. A set of sentences Γ *secures* a set of sentences Δ exactly if every interpretation that makes *all* sentences in Γ true makes *at least one* sentence in Δ true.

2. Proposition: (a) Γ implies D exactly if Γ secures $\{D\}$.
(b) Γ is unsatisfiable exactly if Γ secures $\varnothing$.
(c) D is valid exactly if $\varnothing$ secures $\{D\}$.

II. THE CONCEPT OF A DERIVATION

1. A *sequent* $\Gamma \Rightarrow \Delta$ consists of a finite set of sentences Γ on the left, the symbol '$\Rightarrow$' in the middle, and a finite set of sentences Δ on the right.

2. $\Gamma \Rightarrow \Delta$ is *secure* exactly if Γ secures Δ.

3. A *derivation* [*of* a sequent $\Gamma \Rightarrow \Delta$] is a finite sequence of sequents (called the *steps* or *lines* of the derivation) such that [the last step is $\Gamma \Rightarrow \Delta$ and] each step is of the form $\{A\} \Rightarrow \{A\}$ or follows from earlier steps according to one of several *rules of inference* permitting passage from zero or more sequents taken as *premises* to a sequent taken as *conclusion*.

4. A sequent is *derivable* exactly if there is some derivation of it. A set of sentences Δ is *derivable* from a set of sentences Γ exactly if, for some finite subsets Γ_0 and Δ_0 of Γ and Δ respectively, $\Gamma_0 \Rightarrow \Delta_0$ is derivable.

5. A *proof procedure* is a nonempty set of rules of inference.

III. SYNTACTIC METALOGICAL NOTIONS

1. A *deduction* of D from Γ is a derivation of $\Gamma \Rightarrow \{D\}$.
D is *deducible* from Γ exactly if there is a deduction of D from a finite subset of Γ.

2. A *refutation* of Γ is a derivation of $\Gamma \Rightarrow \varnothing$.
Γ is *refutable* exactly if there is a refutation of a finite subset of Γ.

3. A *demonstration* of D is a derivation of $\varnothing \Rightarrow \{D\}$.
D is *demonstrable* exactly if there is a demonstration of D.

IV. SYNTACTIC/SEMANTIC EQUIVALENCES

1. A proof procedure is *sound* exactly if *every derivable sequent is secure*.

2. A proof procedure is *complete* exactly if *every secure sequent is derivable*.

3. *If* a proof procedure is *both sound and complete* then:

Syntactic notions		Semantic notions
D is *deducible* from Γ	exactly if	D is a *consequence* of Γ.
Γ is *inconsistent* (i.e., *refutable*)	exactly if	Γ is *unsatisfiable*.
D is *demonstrable*	exactly if	D is *valid*.

THE COMPACTNESS THEOREM

I. THREE EQUIVALENT FORMS OF THE THEOREM

Definition: A *model* of a set of sentences is an interpretation in which every sentence in the set is true. (Note: Every set of sentences of a given language is enumerable because every language is enumerable.)

1. The compactness theorem: If every finite subset of a set of sentences is satisfiable (i.e., has a model), then the whole set of sentences is satisfiable (i.e., has a model).

2. Contrapositive form: If a set of sentences is unsatisfiable (i.e., has no model), then some finite subset of the set is unsatisfiable (i.e., has no model).

3. Equivalent form: If $\Gamma \models S$ (i.e., Γ implies S) then for some finite $\Gamma_0 \subseteq \Gamma$ we have $\Gamma_0 \models S$.

Importance: One never needs to look for argument forms with infinitely many premises.

II. AN APPLICATION

Show that a denumerable map can be colored with four colors if every finite submap of it can be.

THE LÖWENHEIM-SKOLEM THEOREMS

I. THE THEOREMS

1. (First) Löwenheim-Skolem theorem: If a set of sentences is satisfiable (i.e., has a model), then it has an enumerable (i.e., finite or denumerable) model.

2. Corollary (Canonical domains theorem): If a set of sentences is satisfiable, then it has a model whose domain is $\mathbf{N}$ or $\{0, 1, 2, \ldots, n\}$ for some $n \in \mathbf{N}$.

3. Second Löwenheim-Skolem theorem: A set of sentences has a denumerable (i.e., countably infinite) model if and only if it has a nonenumerable (i.e., uncountably infinite) model.

II. THREE APPLICATIONS

1. The overspill principle: If a set of sentences has arbitrarily large finite models (i.e., for any $n \in \mathbf{N}$, the set has a model of size $m \geq n$, with $m \in \mathbf{N}$), then it has a denumerable (i.e., infinitely enumerable) model.

2. The Skolem paradox. Let ZFC be the set of axioms of standard set theory. From the axioms it follows that there are nonenumerably many sets of natural numbers. If ZFC is consistent it has a model, so by the Löwenheim-Skolem theorem it has an enumerable model. So *the sentence that in the standard interpretation expresses the proposition that there are nonenumerably many sets of natural numbers has an enumerable model.*

3. Limitations of first-order logic: Unformalizable quantifiers. There is no first-order translation of the quantifier "There are at most finitely many things such that…".

GÖDEL'S FIRST INCOMPLETENESS THEOREM: FORMULATION

I. INFORMAL FORMULATION

- First Incompleteness Theorem (Gödel/Rosser): If a formal system T is (1) consistent, (2) axiomatizable, and (3) sufficiently powerful, then T is *incomplete*: some sentence of the language of T is neither provable nor refutable in T.

- Importance: When doing math, one normally assumes that every statement (conjecture) is either provable or refutable. FIT has the consequence that one may no longer assume that.

II. CLARIFICATION OF CONCEPTS

0. Formal system. A formal system is a *theory*, defined as *a set of sentences that contains every sentence of its language provable from the set*; i.e., a set of sentences closed under provability. -If T is a theory, then $S \in T$ iff $T \vdash S$ (i.e., S is provable from T). -The sentences which are provable from Γ are the *theorems* of Γ, so the theorems of a theory T are all and only the sentences of T. -We presuppose a sound and complete proof procedure, but we don't presuppose that some sentences of T are singled out as "axioms".

1. Consistency. A theory T is *consistent* iff $\varnothing$ is not derivable from T; equivalently, iff there is no S such that T contains both S and $\sim S$; equivalently, iff some sentence of its language is not in T.

2. Axiomatizability. A theory T is *decidable* iff there is an "effective procedure" (i.e., a mechanical procedure involving no randomness and terminating after finitely many steps) for determining, for every sentence, whether or not it is in T. -A theory is *(finitely) axiomatizable* iff there is a (finite) decidable set of sentences Γ such that T contains all and only those sentences of its language that are provable from Γ. -A decidable theory is axiomatizable, but not vice versa. -FIT is equivalent to: if a theory is consistent, *decidable*, and sufficiently powerful, it is incomplete.

3. "Sufficiently powerful". A theory is *sufficiently powerful* if it is an *extension* of (i.e., it is a theory—not necessarily of the language of PA—that is a superset of) "Peano Arithmetic" (PA), namely the axiomatizable theory whose axioms are: (1) $\forall x \sim (Sx = 0)$, (2) $\forall x((x + 0) = x)$, etc.

4. (In)completeness. A theory T is *complete* iff, for every sentence S (of its language), $S \in T$ or $\sim S \in T$; i.e., $T \vdash S$ or $T \vdash \sim S$. A theory is *incomplete* iff it is not complete; i.e., iff, for some sentence S, $S \notin T$ and $\sim S \notin T$; in other words, iff some sentence S is *undecidable* in (or *by* or *for*) T.

III. RIGOROUS FORMULATION

(FIT) If a theory is (1) consistent, (2) axiomatizable, and (3) an extension of PA, then it is incomplete. I.e.: *There is no consistent, axiomatizable, and complete extension of PA.*

IV. WHAT THE THEOREM DOES AND DOES NOT IMPLY

1. Completeness and incompleteness are *syntactic* notions. But in every interpretation S is true or $\sim S$ is true, so FIT has the consequence that, for any interpretation of a consistent and axiomatizable extension T of PA, *some sentence is true in the interpretation but unprovable in T.*

2. It does *not* follow that some sentence is true in the given interpretation but unprovable *simpliciter*. If G is unprovable in T, G is provable in any extension T' of T that contains G. Of course then FIT, applied to T', has the consequence that some sentence G' is true in the given interpretation but unprovable in T'. G' is also unprovable in T. So FIT implies that there are *infinitely* many sentences which are true in the given interpretation but are unprovable in T.

GÖDEL'S FIRST INCOMPLETENESS THEOREM: PROOF STRATEGY

I. HOW *NOT* TO PROVE GÖDEL'S FIRST THEOREM

1. Show that there is a sentence G_T such that:
   (1) $G_T \leftrightarrow$ (G_T is unprovable in T). Informally, G_T is: 'This sentence is unprovable in T'. So:
   (2) $\sim G_T \leftrightarrow$ (G_T is provable in T).

2. "Proof"' that G_T is unprovable in T: Suppose, for reductio, that G_T is provable in T and is thus true (in the standard interpretation $\mathcal{N}^*$ of the language of arithmetic), since provability in T guarantees truth. But, by (2), $\sim G_T$ is true—contradiction.

3. "Proof"' that $\sim G_T$ is unprovable in T: Suppose, for reductio, that $\sim G_T$ is provable in T and is thus true (in $\mathcal{N}^*$), since provability in T guarantees truth. Then, by (2), G_T is provable in T and is thus true—contradiction.

4. Problems with these "proofs": (a) We have not used the assumptions that T is consistent, axiomatizable, and an extension of PA. (b) Why assume that _____? This assumption amounts to the claim that every member of T is true (in $\mathcal{N}^*$).

II. HOW TO PROVE GÖDEL'S FIRST THEOREM

1. Show that there is a sentence G_T such that:
   (1) $T \vdash$ ($G_T \leftrightarrow$ (G_T is unprovable in T)). So:
   (2) $T \vdash$ ($\sim G_T \leftrightarrow$ (G_T is provable in T)).

2. Proof that G_T is unprovable in T.

Suppose, for reductio, that G_T is provable in T. Then it is provable in T that G_T is provable in T (if I can prove something, then I can prove that I can prove it):
   (3) $T \vdash$ (G_T is provable in T).

From (2) and (3) we get: $T \vdash \sim G_T$; i.e., $\sim G_T$ is provable in T. But then both G_T and $\sim G_T$ are provable in T, contradicting the assumption that T is consistent.

3. "Proof"' that $\sim G_T$ is unprovable in T.

Suppose, for reductio, that $\sim G_T$ is provable in T: $T \vdash \sim G_T$. This, together with (2), gives (3): $T \vdash$ (G_T is provable in T). But we have shown above that G_T is not provable in T, so:
   (4) $T \vdash$ (G_T is unprovable in T).

(3) and (4) contradict the assumption that T is consistent.

- There is a problem with step (4). We need an assumption stronger than consistency, namely ω-consistency, to conclude that $\sim G_T$ is unprovable in T.

III. HOW TO EXPRESS "G_T IS UNPROVABLE IN T"

1. Number all sentences so that no two sentences have the same (Gödel) number.

2. Then "G_T is unprovable in T" is equivalent to "the Gödel number of G_T is not the Gödel number of a sentence provable in T".

3. Find a formula $P(x)$ meaning (in $\mathcal{N}^*$) "x is the Gödel number of a sentence provable in T" (just as $F(x)$ = '$\exists y(x = y+y)$' means (in $\mathcal{N}^*$) "x is even"). Then "G_T is unprovable in T" is equivalent to $\sim P(\ulcorner G_T \urcorner)$, where $\ulcorner G_T \urcorner$ is the numeral of the Gödel number of G_T.

4. So we want: $T \vdash (G_T \leftrightarrow \sim P(\ulcorner G_T \urcorner))$. Three tasks: (1) number sentences, (2) find formula $P(x)$, (3) show that there is a sentence G_T such that $T \vdash (G_T \leftrightarrow \sim P(\ulcorner G_T \urcorner))$.

GÖDEL NUMBERING

The idea: Every person has a Social Security number. This number is arbitrary: any other number could do. The number uniquely characterizes the person: no two persons have the same Social Security number. Not every number is a Social Security number. *Think of Gödel numbers as the Social Security numbers of symbols, expressions, and proofs.*

I. GÖDEL NUMBERS OF SYMBOLS

We will follow George and Velleman's system:

Symbol	&	$\vee$	~	$\rightarrow$	$\leftrightarrow$	$\forall$	$\exists$	(	)	=	0	S	<	+	•	x_1	x_2	…
Gödel number	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	…

This table allows one to associate a *sequence of numbers* to any *expression*. E.g.:

$\exists\ x_1\ (\ x_1\ =\ S\ \ 0\ )$
$\downarrow\ \downarrow\ \downarrow\ \downarrow\ \downarrow\ \downarrow\ \downarrow\ \downarrow$
<6, 15, 7, 15, 9, 11, 10, 8>

II. CODE NUMBERS OF SEQUENCES OF NATURAL NUMBERS

1. The *code number* of a finite sequence $\langle a_1, \ldots, a_k\rangle$ of natural numbers is defined as: $\#\langle a_1, \ldots, a_k\rangle = p_1^{a_1+1} \bullet p_2^{a_2+1} \bullet \ldots \bullet p_k^{a_k+1}$, where p_n is the nth prime number ($p_1 = 2$, $p_2 = 3$, $p_3 = 5$, $p_4 = 7$, $p_5 = 11$, …). E.g.: $\#\langle 2, 5, 0\rangle = 2^{2+1} \bullet 3^{5+1} \bullet 5^{0+1} = 8 \bullet 729 \bullet 5 = 29{,}160$.

2. Because of the *fundamental theorem of arithmetic* (i.e., every positive integer greater than 1 can be written as a product of prime numbers in a unique way), no two sequences of natural numbers have the same code number.

III. GÖDEL NUMBERS OF EXPRESSIONS

1. The *Gödel number* of an expression is the code number of the sequence of the Gödel numbers of the symbols in the expression. E.g.:

$(\ \ 0\ \ <\ \ S\ \ \ 0\ \ \ )$ — Expression (i.e., finite sequence of symbols)
$\downarrow\ \downarrow\ \downarrow\ \downarrow\ \downarrow\ \downarrow$
<7, 10, 12, 11, 10, 8> — Sequence of Gödel numbers of the symbols in the expression
$\downarrow\ \downarrow\ \downarrow\ \downarrow\ \downarrow\ \downarrow$
$2^8 \bullet 3^{11} \bullet 5^{13} \bullet 7^{12} \bullet 11^{11} \bullet 13^9$ — Code number of the sequence = Gödel number of the expression

2. Notation: $\#P$ is the Gödel number of the expression P.

3. Distinguish the Gödel number of the *symbol* 'x_1', namely 15, from the Gödel number of the *expression* 'x_1', namely 2^{16}.

4. The number 2^8 (i.e., 256) is the Gödel number of the *expression* '(' but also of the *symbol* 'x_{242}'.

IV. GÖDEL NUMBERS OF PROOFS

Every proof is a finite sequence of sentences: $\langle S_1, \ldots, S_n\rangle$. Its Gödel number is the code number of the sequence $\langle \#S_1, \ldots, \#S_n\rangle$.

REPRESENTABILITY AND DECIDABILITY

I. THE GOAL

In order to express, in the language of T, "Q is provable in T", find a formula $\text{Theorem}_T(x_1)$ such that, if n is the Gödel number of a sentence Q provable in T, then $\text{PA} \vdash \text{Theorem}_T(S^n0)$.

II. INTERMEDIATE STEP

Let $\text{Theorem}_T(x_1)$ be the formula $\exists x_2 \text{Proof}_T(x_1, x_2)$, where $\text{Proof}_T(x_1, x_2)$ is a formula such that if n is the Gödel number of a sentence and m is the Gödel number of a proof of that sentence in T, then $\text{PA} \vdash \text{Proof}_T(S^n0, S^m0)$, and if not, then $\text{PA} \vdash \sim\text{Proof}_T(S^n0, S^m0)$. In other words, the formula $\text{Proof}_T(x_1, x_2)$ *represents* the set $\{<n, m> \in \mathbf{N}^2$: n is the Gödel number of a sentence and m is the Gödel number of a proof of that sentence in $T\}$.

III. REPRESENTABILITY

1. Definition: For any positive integer k, let $\mathbf{N}^k$ be the set of all sequences of natural numbers of length k. We will use the notation $<a_1, a_2, \ldots, a_k>$ to denote an element of $\mathbf{N}^k$. A set $A \subseteq \mathbf{N}^k$ is called *representable* exactly if there is a formula $P(x_1, x_2, \ldots, x_k)$ such that, for every sequence $<a_1, a_2, \ldots, a_k> \in \mathbf{N}^k$:

(i) if $<a_1, a_2, \ldots, a_k> \in A$, then $\text{PA} \vdash P(S^{a_1}0, S^{a_2}0, \ldots, S^{a_k}0)$; and

(ii) if $<a_1, a_2, \ldots, a_k> \notin A$, then $\text{PA} \vdash \sim P(S^{a_1}0, S^{a_2}0, \ldots, S^{a_k}0)$.

In this case, we say that the formula P *represents* the set A.

2. Theorem: If a set $A \subseteq \mathbf{N}^k$ is decidable, then it is representable (and vice versa). (Such a set A is decidable exactly if there is an effective procedure for determining, for each sequence, whether or not it is in the set.)

IV. HOW TO REACH THE GOAL

1. The set $\{<n, m> \in \mathbf{N}^2$: n is the Gödel number of a sentence and m is the Gödel number of a proof of that sentence in $T\}$ is decidable (if T is axiomatizable), so by the above theorem it is representable, so there is such a formula as $\text{Proof}_T(x_1, x_2)$.

2. If $Q \in T$ and n is the Gödel number of Q, then there is a proof of Q in T; let m be the Gödel number of that proof. Then $\text{PA} \vdash \text{Proof}_T(S^n0, S^m0)$, but $\text{Proof}_T(S^n0, S^m0)$ has $\exists x_2 \text{Proof}_T(S^n0, x_2)$ as a consequence, so by the closure of PA under provability we get $\text{PA} \nvdash \exists x_2 \text{Proof}_T(S^n0, x_2)$; i.e., $\text{PA} \vdash \text{Theorem}_T(S^n0)$, which is what we wanted.

THE FIXED POINT LEMMA AND THE LAST STAGE OF THE PROOF

I. REVIEW

We found a formula $\text{Theorem}_T(x_1)$ such that, if n is the Gödel number of a sentence provable in (i.e., a theorem of) an axiomatizable extension T of PA, then $\text{PA} \vdash \text{Theorem}_T(S^n0)$. So:

(1) If $T \vdash Q$, then $\text{PA} \vdash \text{Theorem}_T(\ulcorner Q \urcorner)$,

where $\ulcorner Q \urcorner$ is the numeral of the Gödel number of the sentence Q.

II. THE FIXED POINT LEMMA (DIAGONAL LEMMA)

For any formula $P(x_1)$ in the language of PA, there is a sentence Q such that:

$\text{PA} \vdash (Q \leftrightarrow P(\ulcorner Q \urcorner))$.

Informally: Q is provably equivalent to a statement about its own Gödel number.

III. THE LAST STAGE OF THE PROOF

1. Apply the Fixed Point Lemma to $P(x_1) =$ '$\sim\text{Theorem}_T(x_1)$'.

We get: There is a sentence G_T (a "Gödel sentence" of T) such that:

(2) $\text{PA} \vdash (G_T \leftrightarrow \sim\text{Theorem}_T(\ulcorner G_T \urcorner))$.

That's what we wanted. Now let's repeat the reasoning given in "Proof Strategy".

2. Proof that G_T is unprovable in T.

Suppose, for reductio, $T \vdash G_T$. From (1), $\text{PA} \vdash \text{Theorem}_T(\ulcorner G_T \urcorner)$. From (2), $\text{PA} \vdash (\text{Theorem}_T(\ulcorner G_T \urcorner) \rightarrow \sim G_T)$. Given the closure of PA under provability, $\text{PA} \vdash \sim G_T$, so $T \vdash \sim G_T$, so T is inconsistent—contradiction.

3. Proof that $\sim G_T$ is unprovable in T.

Suppose, for reductio, $T \vdash \sim G_T$. From (2), $\text{PA} \vdash (\sim G_T \rightarrow \text{Theorem}_T(\ulcorner G_T \urcorner))$, so $T \vdash (\sim G_T \rightarrow \text{Theorem}_T(\ulcorner G_T \urcorner))$. Given the closure of T under provability, $T \vdash \text{Theorem}_T(\ulcorner G_T \urcorner)$, so $T \vdash \exists x_2 \text{Proof}_T(\ulcorner G_T \urcorner, x_2)$. But since $T \vdash \sim G_T$ and T is consistent, we don't have $T \vdash G_T$, so for all natural numbers m, $\text{PA} \vdash \sim\text{Proof}(\ulcorner G_T \urcorner, S^m0)$, and so $T \vdash \sim\text{Proof}(\ulcorner G_T \urcorner, S^m0)$. This is not an outright inconsistency, but still it's a kind of inconsistency.

Definition: An extension T of PA is ω-*inconsistent* iff there is a formula $P(x_1)$ such that:
(a) $T \vdash \exists x_1 P(x_1)$ and (b) for every natural number m, $T \vdash \sim P(S^m0)$.

Remark: ω-consistency guarantees consistency.

So we have shown: If $T \vdash \sim G_T$ and T is consistent, then T is ω-inconsistent.
So: If T is (consistent and) ω-consistent, then we don't have $T \vdash \sim G_T$.

4. Conclusion: Gödel's First Incompleteness Theorem.

If a theory T is an axiomatizable extension of PA, then there is a sentence G_T (a *Gödel sentence* of T, defined as any sentence which satisfies (2)) such that:
(i) if T is consistent, then $G_T \notin T$ and (ii) if T is ω-consistent, then $\sim G_T \notin T$.

IV. ROSSER'S INCOMPLETENESS THEOREM

If a theory T is an axiomatizable, consistent extension of PA, then there is a sentence R_T (a *Rosser sentence* of T) such that: $R_T \notin T$ and $\sim R_T \notin T$.

GÖDEL'S FIRST INCOMPLETENESS THEOREM: CONSEQUENCES

I. TRUTH DOES NOT GUARANTEE PROVABILITY

More precisely: If T is a consistent, axiomatizable extension of PA, then *even if every sentence provable in T is true (in $\mathcal{N}$*), not every true (in $\mathcal{N}$*) sentence is provable in T*. Indeed: If every sentence provable in T is true, then $(G_T \leftrightarrow \sim\text{Theorem}_T(\ulcorner G_T \urcorner))$ is true; but $\sim\text{Theorem}_T(\ulcorner G_T \urcorner)$ is true (since $G_T \notin T$), so G_T *is true but unprovable in T*.

II. NONSTANDARD MODELS OF PA

1. A *nonstandard* model of PA is a model of PA in which some sentence is false although the sentence is true in the standard model $\mathcal{N}$* of PA.

2. By the Gödel completeness theorem, every consequence of PA is deducible from PA. Since G_{PA} is not deducible from PA, G_{PA} is not a consequence of PA: there is a model $\mathcal{M}$ of PA in which G_{PA} is false. If G_{PA} is true in $\mathcal{N}$*, $\mathcal{M}$ is a nonstandard model of PA.

3. The incompleteness of PA follows from the existence of a nonstandard model of PA: every sentence which is assigned different truth values by the standard and a nonstandard model of PA is undecidable in PA.

III. UNDECIDABILITY/NON-AXIOMATIZABILITY OF ARITHMETIC

Let *arithmetic* be the set T of all true (in $\mathcal{N}$*) sentences of the language of PA. T is a theory (since every sentence provable from true sentences is true, so T is closed under provability) and is consistent (since it has a model, namely $\mathcal{N}$*) and complete (since, for every sentence S, either S or $\sim S$ is true in $\mathcal{N}$*, so either S or $\sim S$ is in T). Moreover, if PA is true, then every member of PA is in T, so T is an extension of PA. But by FIT there is no consistent, complete, and decidable (or axiomatizable) extension of PA, so T is undecidable and non-axiomatizable.

IV. UNDEFINABILITY OF TRUTH

Theorem (Tarski): There is no formula $P(x_1)$ such that, for every natural number n, $P(S^n0)$ is true in $\mathcal{N}$* if and only if n is the Gödel number of a sentence true in $\mathcal{N}$*.

Proof: Suppose, for reductio, that there is such a formula $P(x_1)$. By the Fixed Point Lemma, there is a sentence Q such that $Q \leftrightarrow \sim P(\ulcorner Q \urcorner)$. By the reductio assumption, $Q \leftrightarrow P(\ulcorner Q \urcorner)$. Contradiction.

V. NON-REPRESENTABILITY OF THEOREMHOOD

Theorem: If T is a consistent extension of PA, then the set $\{n \in \mathbf{N}$: n is the Gödel number of a theorem of $T\}$ is not representable.

Proof: Suppose, for reductio, that the formula $P(x_1)$ represents the above set. By the Fixed Point Lemma, there is a sentence Q such that $\text{PA} \vdash (Q \leftrightarrow \sim P(\ulcorner Q \urcorner))$. If $Q \in T$, then $\text{PA} \vdash P(\ulcorner Q \urcorner)$, so $\text{PA} \vdash \sim Q$, contradicting the consistency of T. If $Q \notin T$, then $\text{PA} \vdash \sim P(\ulcorner Q \urcorner)$, so $\text{PA} \vdash Q$, contradicting again the consistency of T.

Lemma (*Essential undecidability theorem*): No consistent extension of PA is decidable.

GÖDEL'S SECOND INCOMPLETENESS THEOREM: FORMULATION

I. INFORMAL FORMULATION

Second Incompleteness Theorem (Gödel): If a formal system T is (1) consistent, (2) axiomatizable, and (3) sufficiently powerful, then the consistency of T is unprovable in T.

II. CLARIFICATIONS

0. Formal system: A *theory*.

1. Consistency: Needed because an inconsistent theory contains every sentence of its language—i.e., it can prove everything, including the false (in $\mathcal{N}$*) sentence that T is consistent.

2. Axiomatizability: Needed because for example *arithmetic*, namely the theory consisting of all and only the *true* (in $\mathcal{N}$*) sentences, is (as we have seen) consistent and—assuming PA is true—sufficiently powerful (though non-axiomatizable), so a sentence expressing the claim that this theory is consistent is true (in $\mathcal{N}$*), so such a sentence by definition is (provable) in the theory.

3. "Sufficiently powerful" (e.g., an extension of PA): Needed because there are weak (consistent and axiomatizable) theories that do prove their own consistency.

III. EXPRESSING THE CLAIM THAT *T* IS CONSISTENT

0. 'T is consistent' is not a sentence of PA.

1. First way: T is consistent exactly if there is no sentence Q (of PA) such that both Q and $\sim Q$ are provable in T. I.e., $\sim\exists Q(\text{Theorem}_T(\ulcorner Q \urcorner) \,\&\, \text{Theorem}_T(\ulcorner \sim Q \urcorner))$. Unfortunately, this quantifies over sentences, so it is not a sentence of PA. Remedy: $\sim\exists x_1 \exists x_2(\text{Theorem}_T(x_1) \,\&\, \text{Theorem}_T(x_2) \,\&\, \text{Neg}(x_1, x_2))$, where $\text{Neg}(x_1, x_2)$ means (in $\mathcal{N}$*) "x_1 is the Gödel number of a sentence and x_2 is the Gödel number of the negation of that sentence".

2. Second way: *An extension T of PA is consistent exactly if it does not contain the sentence '0=1'*: $\sim\text{Theorem}_T(\ulcorner \text{`}0=1\text{'} \urcorner)$. Much simpler.

3. Choose one of the above two sentences and call it "the consistency sentence of T", Con_T.

IV. RIGOROUS FORMULATION

If a theory is (1) consistent, (2) axiomatizable, and (3) an extension of PA, then it does not contain its consistency sentence.

GÖDEL'S SECOND INCOMPLETENESS THEOREM: PROOF STRATEGY

I. THE PROOF

Suppose, for reductio, that T is a consistent, axiomatizable extension of PA and that $T \vdash \text{Con}_T$. Lemma: $\text{PA} \vdash (\text{Con}_T \rightarrow G_T)$, G_T being a Gödel sentence of T. But then, since T is an extension of PA, $T \vdash (\text{Con}_T \rightarrow G_T)$, and by the closure of T under provability $T \vdash G_T$, contradicting the proof of the First Incompleteness Theorem (where we saw that $G_T \notin T$).

II. THE LEMMA

1. We saw in the discussion of the First Incompleteness Theorem that if T is consistent, then G_T is true (in $\mathcal{N}^*$). We can translate this reasoning into the language of PA. The result is a proof in PA of $\text{Con}_T \rightarrow G_T$.

2. The converse also holds: $\text{PA} \vdash (G_T \rightarrow \text{Con}_T)$. Indeed: G_T is equivalent in PA to "G_T is unprovable in T", from which it follows that T is consistent, since there is a sentence it cannot prove.

3. So $\text{PA} \vdash (\text{Con}_T \leftrightarrow G_T)$: *Any Gödel sentence of T is equivalent in PA (and thus in T) to the consistency sentence of T.*

COMPUTABILITY

I. FOUR KINDS OF COMPUTABILITY

0. Effective computability: A function *f* from natural numbers to natural numbers is *effectively computable* exactly if there is a finitely terminating deterministic algorithm which, when given as input a numeral for any given natural number *n*, gives as output a numeral for the corresponding value *f*(*n*) of the function. (This is an intuitive notion, not a rigorous one. It is easily generalizable to many-place functions. It is related to the notion of decidability: A set of natural numbers is (*effectively*) *decidable* exactly if its *characteristic function*—namely the function that takes the value 1 for every natural number in the set and the value 0 for every natural number not in the set—is effectively computable.)

1. Turing computability: A function *f* from natural numbers to natural numbers is *Turing computable* exactly if there is a Turing machine which, when it is presented with a tape which contains an unbroken block of *n*+1 strokes (and is otherwise blank) and starts scanning at the leftmost square containing a stroke, halts at the leftmost square containing a stroke of a tape which contains an unbroken block of *f*(*n*)+1 strokes (and is otherwise blank), for any natural number *n*.

2. Abacus computability: A function *f* from natural numbers to natural numbers is *abacus computable* exactly if there is an abacus machine which, when it starts with *n* stones in the first box and no stone in all other boxes, halts with *f*(*n*) stones in some pre-specified box, for any natural number *n*.

3. Recursive computability: A function *f* from natural numbers to natural numbers is *recursively computable* (or *recursive*) exactly if it is obtainable from the *basic functions*—namely the zero, successor, and identity functions—by *composition*, (primitive) *recursion*, and *minimization.*

II. RELATIONSHIPS BETWEEN THE FOUR KINDS

1. Obviously, every Turing computable or abacus computable or recursively computable function is effectively computable.

2. Turing's thesis: Every effectively computable function is Turing computable.

3. Church's thesis: Every effectively computable function is recursively computable.

4. Fundamental computability theorem: *Turing computability, abacus computability, and recursive computability are equivalent to each other.*

5. Corollary: Turing's thesis and Church's thesis are equivalent. So there is strong evidence for both theses.

HILBERT'S PROGRAM

I. BACKGROUND: CLASSICAL MATHEMATICS VS. INTUITIONISM

1. Classical mathematics is based on the realist assumption that mathematical reality is fully determinate: every intelligible question that can be asked about it has an answer. In the case of questions concerning mathematical propositions that involve unbounded quantification (such as the *twin prime conjecture*: $\forall x \exists y((x<y)$ & (y is prime) & (y+2 is prime))), the answer depends on the features of completed infinities.

2. Intuitionism denies that mathematical reality is fully determinate and that *completed* (as opposed to *potential*) infinities exist. It claims that some concepts are *indefinitely extensible*: their extensions cannot be fully determined, because any collection of objects that fall under such a concept can be extended. Intuitionists understand propositions such as the twin prime conjecture as being about the entities that could in principle be generated by some operation.

3. Classical mathematicians and intuitionists agree that mathematical reality exists and that *infinitary* mathematical sentences (i.e., mathematical sentences with *unbounded* quantification) are intelligible, but disagree (1) on a *theoretical* level, about the nature of mathematical reality and about the proper understanding of infinitary mathematical sentences, and (2) on a *practical* level, about the *acceptability of certain forms of inference* and thus about whether one is *justified in believing* certain infinitary mathematical propositions. In particular, *intuitionists reject the Law of Double Negation Elimination* (and thus certain proofs by reductio) *and the Law of the Excluded Middle.*

II. THE GOALS OF HILBERT'S PROGRAM

1. A first main goal of Hilbert's program was to justify (some of) the infinitary mathematical assertions (and thus, derivatively and *instrumentally*, the reasoning) of classical mathematicians from a perspective acceptable to intuitionists. This would reconcile classical mathematicians and intuitionists with respect to their *practical* disagreements, though not with respect to their *theoretical* ones.

2. A second main goal of Hilbert's program was to establish, from a perspective acceptable both to classical mathematicians and to intuitionists, the consistency of an axiomatic theory of mathematics.

3. The second goal will be seen to help achieve the first, but is also independently motivated by a desire to eliminate the epistemic possibility that new paradoxes will arise.

III. FINITARY SENTENCES AND FINITARY JUSTIFIABILITY

1. Finitary sentences are of three basic kinds: (i) non-quantified (mathematical) sentences, (ii) *bounded* existentially or universally quantified sentences, and (iii) *unbounded universally* quantified sentences (containing no unbounded existential quantifiers). (Any combinations of these three are also finitary.) According to finitism, *unbounded existentially* quantified sentences are meaningless. Note that some sentences (e.g., of kind (iii) above) are *both finitary and infinitary*.

2. A sentence is finitarily justifiable exactly if it is finitary and its correctness can be established either by means of a finite calculation or, if the sentence is of the form $\forall x P(x)$, by

means of an algorithm which, for every numeral k, establishes the correctness of $P(k)$ (either by means of a finite calculation or, if $P(k)$ is of the form $\forall y Q(y)$, by means of an algorithm…).

3. Finitarily justifiable sentences are acceptable (as justified) to both classical mathematicians and intuitionists, but not every sentence acceptable to classical mathematicians or to intuitionists is finitarily justifiable.

IV. FINITARILY JUSTIFYING CONSISTENCY

1. Consider an axiomatizable theory T which includes all sentences acceptable to classical mathematicians (including infinitary sentences). The proposition that T is consistent can be expressed by the following finitary sentence (the *consistency sentence* of T): every (syntactically defined) derivation in T has a last line different from "$0=1 \ \& \ 0\neq1$". Hilbert's goal is to finitarily justify the consistency sentence of T.

2. Establishing this goal would not address the primary epistemological worry of intuitionists about classical mathematics, so the other main goal of Hilbert's program is also needed.

V. FINITARILY JUSTIFYING (SOME) CLASSICAL MATHEMATICS

1. The goal is to establish:

(2) Every finitary sentence provable in T is finitarily justifiable.

(In other words, T is a *conservative extension* of finitary mathematics with respect to finitary sentences.) This would finitarily justify not the whole of classical mathematics, but rather the part consisting of finitary sentences (which includes infinitary sentences with universal quantifiers but not with existential quantifiers).

2. Achieving this goal would also *instrumentally* justify the rules of inference used by classical mathematicians (including the rules that are unacceptable to intuitionists): these rules would be dispensable but convenient shorthands, never leading to finitarily unjustifiable finitary sentences.

3. The goal can be established by using two premises:

(0) The consistency sentence of T is finitarily justifiable.

(1) Every finitarily justifiable *quantifier-free* finitary sentence is provable in T.

4. Proof. Take any finitary sentence Q provable in T. (i) Suppose Q is quantifier-free. If Q is false, then $\sim Q$ is true and thus (given that $\sim Q$ is quantifier-free) finitarily justifiable, so by (1) $\sim Q$ is provable in T, and thus T is inconsistent. So "if T is consistent, Q is true" is finitarily justifiable. Given (0), Q is finitarily justifiable. (ii) Supose Q is of the form $\forall x P(x)$, where $P(x)$ is quantifier-free. Let k be any numeral. Since $\forall x P(x)$ is provable in T, so is $P(k)$ (by universal instantiation). But $P(k)$ is quantifier-free, so by case (i) above $P(k)$ is finitarily justifiable, and thus (given that k was arbitrary) so is $\forall x P(x)$.

GÖDEL'S FIRST THEOREM VS. HILBERT'S PROGRAM

I. HILBERT'S IMPLICIT ARGUMENT AGAINST INCOMPLETENESS

Premise 1: For every sentence *S*, either *S* is true (in $\mathcal{N}^*$) or ~*S* is true.
Premise 2: In some sufficiently powerful axiomatizable theory every true sentence is provable.
Conclusion: There is a sufficiently powerful axiomatizable theory *T* which includes every true sentence and which is such that, for every sentence *S*, either *S* is provable in *T* or ~*S* is provable in *T* (i.e., *T* is complete).

The argument is deductively valid, and premise 1 is integral to Hilbert's program. But Hilbert can drop premise 2: he aims at establishing the truth of every provable (finitary) sentence, not the provability of every true sentence.

II. HOW GÖDEL'S FIRST THEOREM LIMITS HILBERT'S PROGRAM

1. To *fully* justify (from a practical point of view) classical mathematics to intuitionists, Hilbert would need to establish not only (2) but also (3):
   (2) Every finitary sentence provable in *T* is finitarily [and thus intuitionistically] justifiable.
   (3) Every sentence provable in *T* is intuitionistically justifiable.

2. But Gödel's first theorem provides a counterexample to (3):
   (4) If $T = T^* + \sim G_{T^*}$, then $\sim G_{T^*}$ is provable in *T* but is intuitionistically unjustifiable.

3. Proof. G_{T^*} is intuitionistically justifiable because Gödel's reasoning provides an intuitionistically acceptable proof of the unprovability of G_{T^*} in T^*, and G_{T^*} says precisely that it is unprovable in T^*.

4. Note that (4) is no counterexample to (2) because $\sim G_{T^*}$ is the negation of an unbounded universally quantified sentence (since G_{T^*} is equivalent in T^* to the consistency sentence of T^*) and is thus not a finitary sentence.

GÖDEL'S SECOND THEOREM VS. HILBERT'S PROGRAM

I. HOW GÖDEL'S 2ND THEOREM AFFECTS JUSTIFYING CONSISTENCY

1. Although, as we saw, Hilbert only needs to assume (1), it is hard to see how he can avoid assuming also (1*):
   (1) Every finitarily justifiable *quantifier-free* finitary sentence is provable in *T*.
   (1*) Every finitarily justifiable finitary sentence is provable in *T*.

2. But if one assumes (1*), given SIT it follows that, *if T is consistent, then the consistency sentence of T is finitarily unjustifiable*, so Hilbert's second goal is unachievable.

3. Proof. Given that the consistency sentence of *T* is a finitary sentence, if this sentence is finitarily justifiable, then by (1*) it is provable in *T*. But then by SIT *T* is inconsistent.

II. HOW THE TWO PERSPECTIVES ACCOMMODATE GÖDEL'S WORK

1. Both classical mathematicians and intuitionists accept that Gödel's work shows that truth (in the standard interpretation) transcends provability in any given formal system.

2. But intuitionists only assert that, given any particular formal system, one can produce a true sentence which is unprovable in that system. Classical mathematicians, by contrast, assert in addition that there is a completed infinite totality of true sentences, a determinate collection which results "after" the never-ending process of extension is completed.

SKEPTICISM ABOUT CONSISTENCY

I. THE ISSUE

1. It is a consequence of Gödel's Second Incompleteness Theorem (SIT) that:
   (1) If PA is consistent, then the consistency of PA is unprovable in PA.
2. Is there a good argument from (1) to (2)?
   (2) The consistency of PA is doubtful.
3. Is there a good argument from (1) to (3)?
   (3) The consistency of PA is unprovable.

II. DOES SIT RENDER THE CONSISTENCY OF PA DOUBTFUL?

1. Not for those who are (reasonably) certain that the axioms of PA are true (and thus that PA is consistent). Why expect the consistency of PA to be provable in PA?

2. One might think that (1) renders the consistency of PA doubtful because (1) *lowers the probability* that PA is consistent. Surprisingly, however, something like the opposite is true: *it is more probable that PA is consistent given that it cannot prove its own consistency than given that it can.*

III. DOES SIT RENDER THE CONSISTENCY OF PA UNPROVABLE?

No, because one can prove (and Gentzen has proved) the consistency of PA by using a method not formalizable in PA (namely transfinite induction).

LUCAS'S ARGUMENT—VERSION I

I. DEFINITIONS—VERSION I

1. A machine *M corresponds* to a theory *T* exactly if the potential output of the machine consists of all and only the theorems of *T*. (So a given machine corresponds to at most one theory.)

2. A machine *M adequately represents* a human mind *H* (as far as mathematical reasoning is concerned) exactly if the machine corresponds to a theory whose theorems are all and only the sentences of the language of arithmetic that the human mind can (in principle) show to be true (in the standard interpretation).

3. A machine is *axiomatizable* exactly if it corresponds to an axiomatizable theory.

4. A machine is *consistent* exactly if it corresponds to a consistent theory.

II. LUCAS'S THESIS

No consistent and axiomatizable machine adequately represents a human mind.

III. LUCAS'S ARGUMENT—VERSION I

1. Consider a human mind *H* and suppose for reductio that some consistent and axiomatizable machine *M* adequately represents *H*.

2. Then *M* corresponds to a consistent and axiomatizable theory *T* which is an extension of PA (since the human mind can show every theorem of PA to be true).

3. By Gödel's First Incompleteness Theorem, *T* does not contain its Gödel sentence, G_T.

4. But *H* can show that G_T is true (because G_T is equivalent to "G_T is not in *T*", which *H* knows from FIT).

5. So there is a sentence of the language of arithmetic (namely G_T) that is not in *T* but that *H* can show to be true. This contradicts the assumption that *M* adequately represents *H*, and the reductio is complete.

IV. THE STANDARD RESPONSE

Step 4 is problematic: *H* can show that, *if T* is consistent, G_T is [not in *T* and is thus] true, but in general *H* cannot show that *T* is consistent.

LUCAS'S ARGUMENT—VERSION II

I. INTRODUCTION

1. Lucas insists that his argument is *essentially dialectical*: "It is not a straightforward proof, starting from some acceptable premises …, but rather it is a schema of refutation, showing how *if* the mechanist were to … say what machine was equivalent to a named man, the mentalist can refute at least that equivalence."

2. The crucial point is that the output of a human mind varies depending on the input, namely on which (if any) machine is proposed as "equivalent" to (i.e., adequately representing) the mind. "The mind does not go round uttering theorems in the hope of tripping up any machines that may be around."

II. DEFINITIONS—VERSION II

1. A machine *M corresponds* to a theory *T given input I* exactly if the potential output of the machine when its input is *I* consists of all and only the theorems of *T*.

2. A machine *M adequately represents* a human mind *H* exactly if, for *every* input *I*, the machine corresponds given *I* to a theory whose theorems are all and only the sentences of the language of arithmetic that the human mind can (in principle) show, *given input I*, to be true.

3. A machine is *axiomatizable* exactly if, for every input *I*, the machine corresponds given *I* to an axiomatizable theory.

4. A machine is *consistent* exactly if, for every consistent input *I*, the machine corresponds given *I* to a consistent theory.

III. LUCAS'S THESIS

No consistent and axiomatizable machine adequately represents a human mind.

IV. LUCAS'S ARGUMENT—VERSION II

1. Consider a human mind *H* and suppose for reductio that some consistent and axiomatizable machine *M* adequately represents *H*. Let input *I* consist of the sentence "*M* is consistent".

2. Then *M* corresponds, given input *I*, to a consistent and axiomatizable theory *T* whose theorems are all and only the sentences of the language of arithmetic that *H* can show, given input *I*, to be true.

3. Then *T* is an extension of PA, and by Gödel's First Incompleteness Theorem, *T* does not contain its Gödel sentence, G_T.

4. Since *I* consists of the sentence "*M* is consistent", given input *I*, *H* can show that G_T is true (because *H* can show that, *if M* is consistent, G_T is true).

5. So there is a sentence in the language of arithmetic (namely G_T) that is not in *T* but that *H* can show, given input *I*, to be true. This contradicts the assumption that *M* adequately represents *H*, and the reductio is complete.

GAIFMAN'S ARGUMENT

I. DEFINITIONS

1. A machine *M corresponds* to a theory *T* exactly if the potential output of the machine consists of all and only the theorems of *T*. (So a given machine corresponds to at most one theory.)

2. A machine *M adequately represents* a human mind *H* (as far as mathematical reasoning is concerned) exactly if the machine corresponds to a theory whose theorems are all and only the sentences of the language of arithmetic that the human mind can (in principle) show to be true (in the standard interpretation).

3. A machine is *axiomatizable* exactly if it corresponds to an axiomatizable theory.

4. A machine is *consistent* exactly if it corresponds to a consistent theory.

5. A human mind is *consistent* exactly if the set of all and only those sentences of the language of arithmetic that the human mind can show to be true is consistent.

II. GAIFMAN'S THESIS

No axiomatizable machine can be shown by a human mind which can show that it is consistent to adequately represent it. I.e.: If a human mind *H* can show that *H* is consistent, then for every axiomatizable machine *M*, *H* cannot show that *M* adequately represents *H* [even if *M* does in fact adequately represent *H*].

III. GAIFMAN'S ARGUMENT

1. Suppose, for reductio, that for some human mind *H* and for some axiomatizable machine *M*, *H* can show both that *H* is consistent and that *M* adequately represents *H* (so that *M* in fact adequately represents *H*).

2. Then *M* corresponds to a consistent and axiomatizable theory *T* whose theorems are all and only the sentences of the language of arithmetic that *H* can show to be true.

3. Then *T* is a consistent and axiomatizable extension of PA, and by Gödel's Second Incompleteness Theorem, *T* does not contain its consistency sentence, Con_T.

4. Since *H* can show both that *H* is consistent and that *M* adequately represents *H*, *H* can show that *T* is consistent; i.e., that Con_T is true.

5. So there is a sentence in the language of arithmetic (namely Con_T) that is not in *T* but that *H* can show to be true. This contradicts the assumption that *M* adequately represents *H*, and the reductio is complete.

IV. SIGNIFICANCE OF GAIFMAN'S THESIS

If we can show that we are consistent, we can never recognize that a particular axiomatizable machine adequately represents us. This inability is a structural inherent feature of us; it is not due to our physical limitations.

PENROSE'S ARGUMENT

PART 1: A DIALOGUE

Human: Have you looked over the articles I lent you—the ones by Gödel, and also the others, that discuss implications of his theorem?

Robot: Certainly—although the articles were rather elementary, they were interesting. I'm sure that I would have thought of the theorem myself if I'd had just a little more time.

Human: By the way, have I ever shown you the particular detailed rules that we used in order to set in train the computational procedures that led to the construction and development of you and your robot colleagues? It's all in these files and computer disks.

Robot (*some 13 minutes, 41.7 seconds later*): Fascinating—although at a quick glance, I can see at least 519 obvious ways that you could have achieved the same effect more simply.

Human: I think we have done a good enough job. Goodness—the mathematical abilities of you and your colleagues seem now to be very impressive indeed. You are now beginning to move far ahead of the capabilities of all human mathematicians.

Robot: That's clearly true. Even as we speak, I have been thinking of a number of new theorems that go far beyond results published in the human literature. Also, my colleagues and I have noticed a few fairly serious errors in results that have been accepted as true by human mathematicians over a number of years.

Human: What about you robots? Don't you think that you and your robot colleagues might sometimes make mistakes—I mean when you assert theorems as definitively established?

Robot: No, certainly not. Once a mathematical robot has asserted that some result is a *theorem*, then it can be taken that the result is unassailably true. My colleagues and I have felt uneasy about the comparatively slipshod standards that your human mathematical colleagues are prepared to put up with. We are proposing to start a comprehensive database of mathematical theorems that we accept as having been unassailably established. These results will be assigned a special imprimatur *, signifying acceptance by our *Society for Mathematical Intelligence in the Robot Community* (SMIRC)—a society with extremely rigorous criteria for membership. You can rest assured that when we assign our imprimatur * to a result, we *do* guarantee its mathematical truth.

Human: Something has occurred to me. Those original mechanisms *M*, according to which my colleagues and I set in train all the developments that led to the present community of mathematical robots—do you realize that they provide a *computational procedure* for generating all the mathematical assertions that will ever be *-accepted by SMIRC? The family of all *-assertions that you will ever eventually come up with *can* be generated by one particular Turing machine. I could even specify that particular Turing machine in *practice*, using all those files and disks I showed you.

Robot: That is a very elementary remark. Yes, you could do it, but it's hardly worth wasting months of your precious time; I can do it straight off, if you would like me to.

Human: No, there is no point in that. But I want to follow these ideas up for a moment. Let's refer to the computational procedure that generates *-asserted sentences as $Q(M)$, or as just Q

for short. It follows that there must be a Gödel-type mathematical assertion—which I'll call $G(Q)$—and the truth of $G(Q)$ is a consequence of the assertion that you robots never make mistakes with regard to the sentences that you are prepared to claim with *-certainty.

Robot: Yes; you must be right … hmm.

Human: And $G(Q)$ must actually be *true*, because you robots never *do* make mistakes with regard to your *-assertions.

Robot: Of course.

Human: Wait a minute … it would also follow that $G(Q)$ must actually be something that you robots are incapable of perceiving as being actually true—at least, not with *-certainty.

Robot: The fact that we robots were originally constructed according to *M*, together with the fact that our *-assertions are never wrong, *does* have the clear and unassailable implication that the sentence $G(Q)$ must be true. I suppose you are thinking I ought to be able to persuade SMIRC to give the *-imprimatur to $G(Q)$. Indeed, they *must* accept this. Yet … it's impossible that they can accept $G(Q)$, because, by its very nature of your Gödel's construction, $G(Q)$ is something that lies outside what can be *-asserted by us—provided that we do not *in fact* ever make mistakes in our *-assertions. I suppose you might think this implies that there must be some doubt in our minds as to the reliability of our assignations of *. However, I don't concede that our *-assignations might ever be wrong, especially with all the care and precautions that SMIRC are going to be taking. It must be the case that it's you humans who have got it wrong, and the procedures incorporated into *Q* are *not* after all the ones you used, despite what you are telling me and what your documentation seems to assert. Anyway, SMIRC will never be absolutely sure of the fact that we have actually been constructed according to *M*, i.e., by the procedures encapsulated by *Q*. We have only your word to go on for that.

Human: I can assure you that they *are* the ones we used; I should know, since I was personally responsible for them.

Robot: Perhaps one of your assistants got it wrong when following out your instructions.

Human: You're grasping at straws. Even if someone did introduce some errors, my colleagues and I should eventually be able to track them down and so find out what your *Q really* is. I think what worries you is the fact that we actually *know*—or at least can find out—what procedures were used to set up your construction. This means that we could actually write down the sentence $G(Q)$ and know for sure that it is actually true—provided that it is in fact the case that you never make mistakes in your *-assertions. However, *you* cannot be sure that $G(Q)$ is true; at least you cannot assign it the certainty that would satisfy SMIRC sufficiently to give it *-status. This would seem to give us humans an ultimate advantage over you robots, in principle if not in practice, since there are sentences that are in principle accessible to us but not to you. I don't think that you robots can face such a possibility—yes, of course, *that's* why you are so uncharitably accusing us of having got it wrong!

Robot: Don't go attributing your petty human motives to us. But of course it's true that I *can't* accept that there are sentences accessible to humans but not to robots. Robot mathematicians are certainly in *no* way inferior to human mathematicians—though I suppose it's conceivable that, conversely, any particular sentence that is accessible to us is also, in principle, eventually accessible to humans in their plodding ways. OK. I suppose that *you* might believe that it is just conceivable that, occasionally, the members of SMIRC might make a mistake in their assignations of *. I suppose also that SMIRC might not be unassailably convinced that their as-

signations of * are invariably error free. In this way, $G(Q)$ could fail to acquire *-status, and the contradiction would be avoided. Mind you, this is not to say that I am admitting that we robots *would* ever make erroneous *-assertions. It's just that we cannot be absolutely *sure* that we would not.

Human: Are you trying to tell me that although truth is absolutely guaranteed for each individually asserted sentence, there is no guarantee that there is not some error amongst the whole collection of them? Surely that's illogical, isn't it?

Robot: I can't accept that robots are illogical. The sentence $G(Q)$ is only a consequence of the other sentences if it is actually the case that we were constructed according to *M*. We cannot guarantee $G(Q)$ simply because we cannot guarantee that we *were* constructed according to *M*. Robot certainly cannot depend on human fallibility.

Human: Your uncertainty as to the procedures that actually underlie your own construction must surely place some doubts in your mind as to whether all *-sentences must be true, if only because you might not trust *us* to have set up things correctly.

Robot: I suppose I'm prepared to admit that because of your own unreliability, there could be some tiny uncertainty, but since we have evolved so far away from those initial sloppy procedures of yours, this is not an uncertainty large enough to take seriously. But you might perhaps be thinking that there could be some *inbuilt*, systematic error in robot reasoning. This I refuse to accept; it's simply inconceivable to me that the underlying principles that govern SMIRC's *-acceptance of mathematical argument could be wrong in such a blatant way.

Human: Something else occurs to me. It doesn't matter whether you are prepared to accept that the particular mechanisms *M* were the things underlying your own construction, provided that you merely agree that this is a logical possibility. SMIRC would have to have another category of assertion which they were not so unassailably convinced of—let us call these $*_M$-assertions—but which they would regard as unassailable *deductions* from the *assumption* that they were all constructed from *M*. All the original *-assertions would be counted among the $*_M$-assertions, of course, but *also* anything that they could unassailably conclude from the assumption that it is *M* that governs their actions. They would not have to believe *M*, but as a logical exercise, they could explore the implications of this assumption. As we have agreed, $G(Q)$ would have to count as a $*_M$-assertion. Knowing the rules of *M*, it is then possible to obtain a *new* algorithmic procedure Q_M, which generates precisely those $*_M$-assertions (and their logical consequences) that SMIRC will accept on the basis of the assumption that they were constructed according to *M*.

Robot: Of course; and while you were speaking, I have been amusing myself by working out the precise form of the algorithm Q_M … Yes, and now I have also *anticipated* you; I have also worked out its Gödel sentence $G(Q_M)$. I'll print it out for you if you want.

Human: I don't need it printed. But is $G(Q_M)$ actually *true*—unassailably true?

Robot: Oh, I see … SMIRC would accept $G(Q_M)$ as true—unassailably—but only under the hypothesis that we were constructed according to *M*—which, as you know, is an assumption that I'm finding exceedingly dubious. The point is that $G(Q_M)$ follows from the following assertion : "all sentences that SMIRC is prepared to accept as unassailable, conditional upon the assumption that we were constructed according to *M*, are true". So I don't know whether $G(Q_M)$ is *actually* true. It depends upon whether your dubious assumption is correct or not.

Human: I see. So you are telling me that you (and SMIRC) would be prepared to accept—*unassailably*—the fact that the truth of $G(Q_M)$ follows from the assumption that you were constructed according to *M*.

Robot: Of course.

Human: So the sentence $G(Q_M)$ must be a $*_M$-assertion then!

Robot: Ye … eh … what? Yes, you're right of course. But by its very definition, $G(Q_M)$ cannot itself be an actual $*_M$-assertion unless at least one of the $*_M$-assertions is actually *false*. Yes … this only confirms what I have been telling you all along, although now I can make the definitive claim that we actually have *not* been constructed according to *M*.

Human: But that's surely not the point. The same argument would apply whatever computational rules we used. So *whatever* '*M*' I tell you, you can rule it out by that argument!

(Adapted from Roger Penrose, *Shadows of the Mind*, pp. 179-189.)

PART 2: THE ARGUMENT MORE FORMALLY

I. DEFINITIONS

1. Let T_H be the theory whose theorems are all and only those sentences of the language of PA that *H* can show to be true (in $\mathcal{N}$*).

2. Let T_M be the theory that corresponds to machine *M*.

3. Let *Q* be the sentence '$T_H = T_M$' (i.e., *M* adequately represents *H*).

4. Let T_{HQ} be the set consisting of all and only those sentences *S* of the language of PA such that *H* can show that *S* follows from *Q*.

II. PENROSE'S THESIS

No consistent and axiomatizable machine adequately represents a human mind.

III. PENROSE'S ARGUMENT

0. It is enough to show that, for any consistent and axiomatizable machine *M* and for any human mind *H*, if *H* can*not* show that *M* does not adequately represent *H*, then *M* does not adequately represent *H*. (This is enough because, if *H can* show that *M* does not adequately represent *H*, then *M* does not adequately represent *H*.)

1. Suppose, for reductio, that some consistent and axiomatizable machine *M* adequately represents some human mind *H* and *H* cannot show that *M* does not adequately represent *H*.

2. Then $T_M = T_H = T$ is a consistent and axiomatizable extension of PA, $\sim Q \notin T_H$, and *Q* is true.

3. Then $T_{HQ} = T_H + Q$ is consistent (since $\sim Q \notin T_H$) and is an axiomatizable extension of (T_H and thus of) PA, so $Con_{T_{HQ}} \notin T_{HQ}$.

4. But *H* can show that T_{HQ} is true and thus consistent *if Q* is true, so *H* can show that $Con_{T_{HQ}}$ follows from *Q*, so $Con_{T_{HQ}} \in T_{HQ}$. Contradiction.