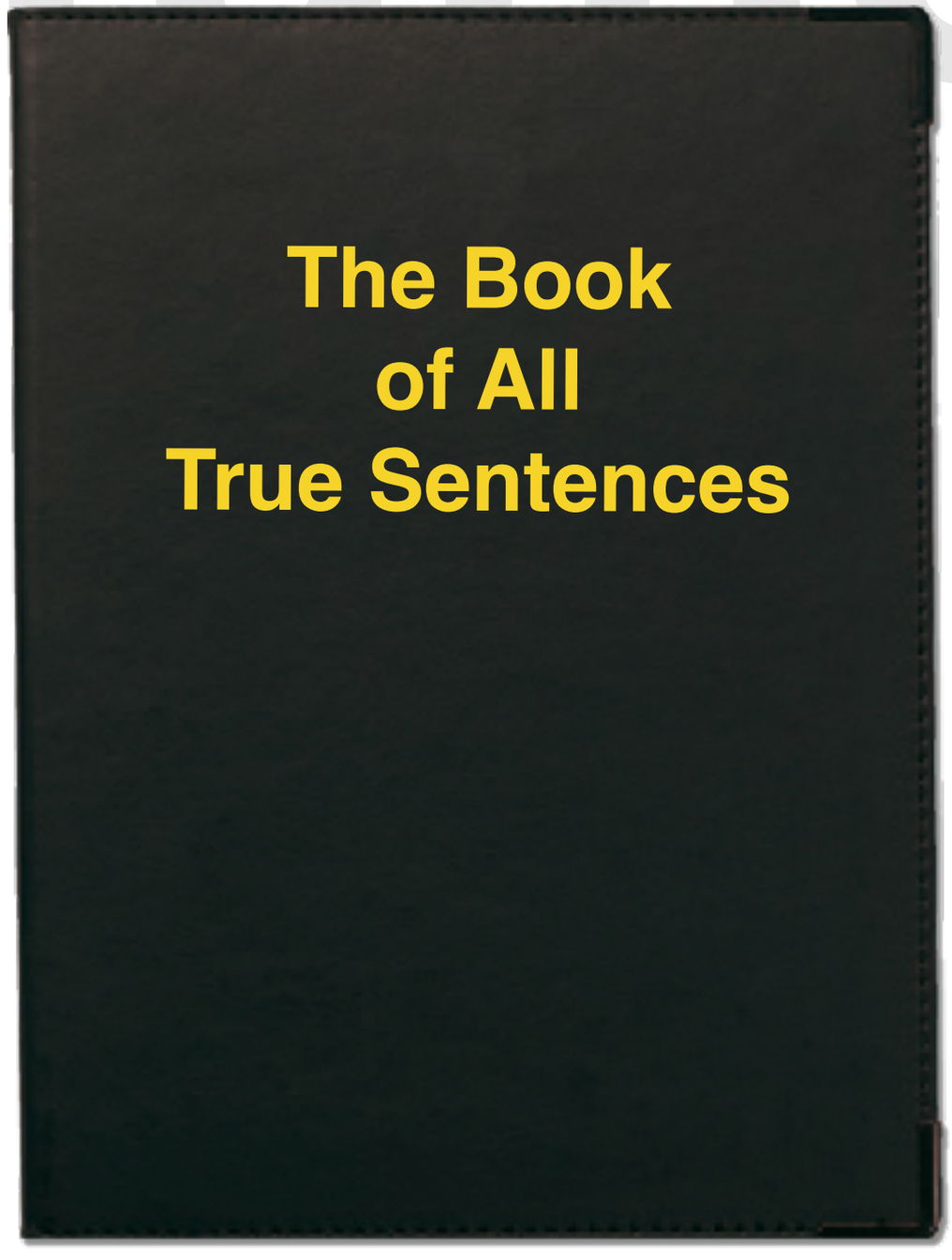
this slide intentionally left blank

Intuition



http://fouryears.eu/2013/01/27/on-godels-theorems-and-the-universe/

Intuition



“This sentence is not in The Book”

http://fouryears.eu/2013/01/27/on-godels-theorems-and-the-universe/

Intuition



“This sentence is not in The Book”

incomplete

http://fouryears.eu/2013/01/27/on-godels-theorems-and-the-universe/

Intuition

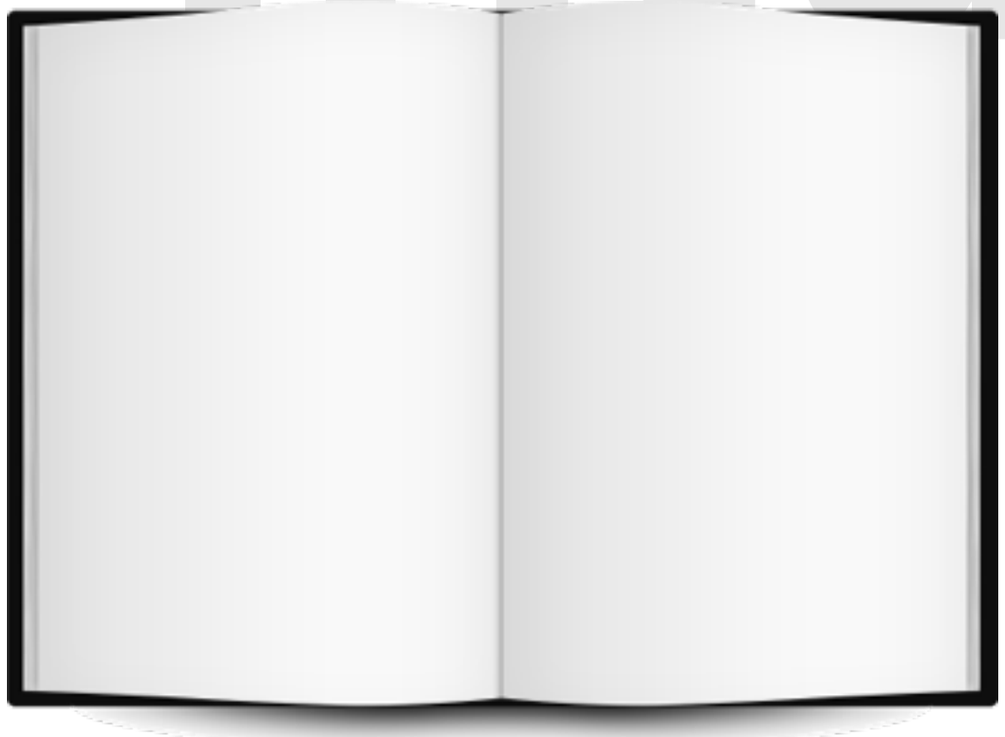


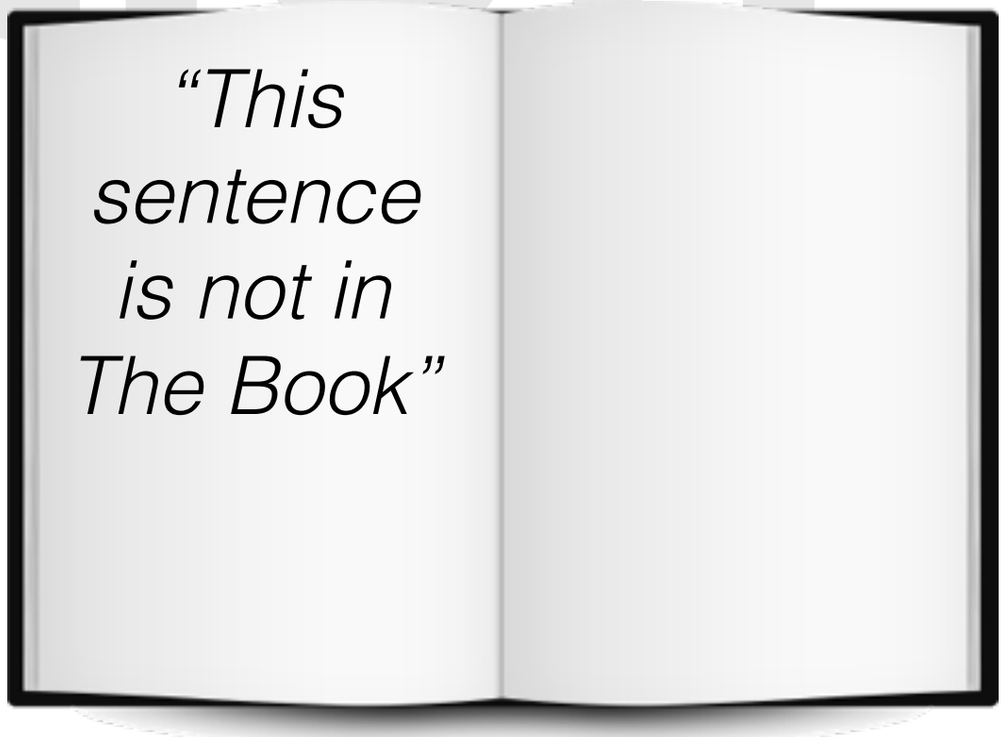
“This sentence is not in The Book”

incomplete

http://fouryears.eu/2013/01/27/on-godels-theorems-and-the-universe/

Intuition





“This sentence is not in The Book”

incomplete

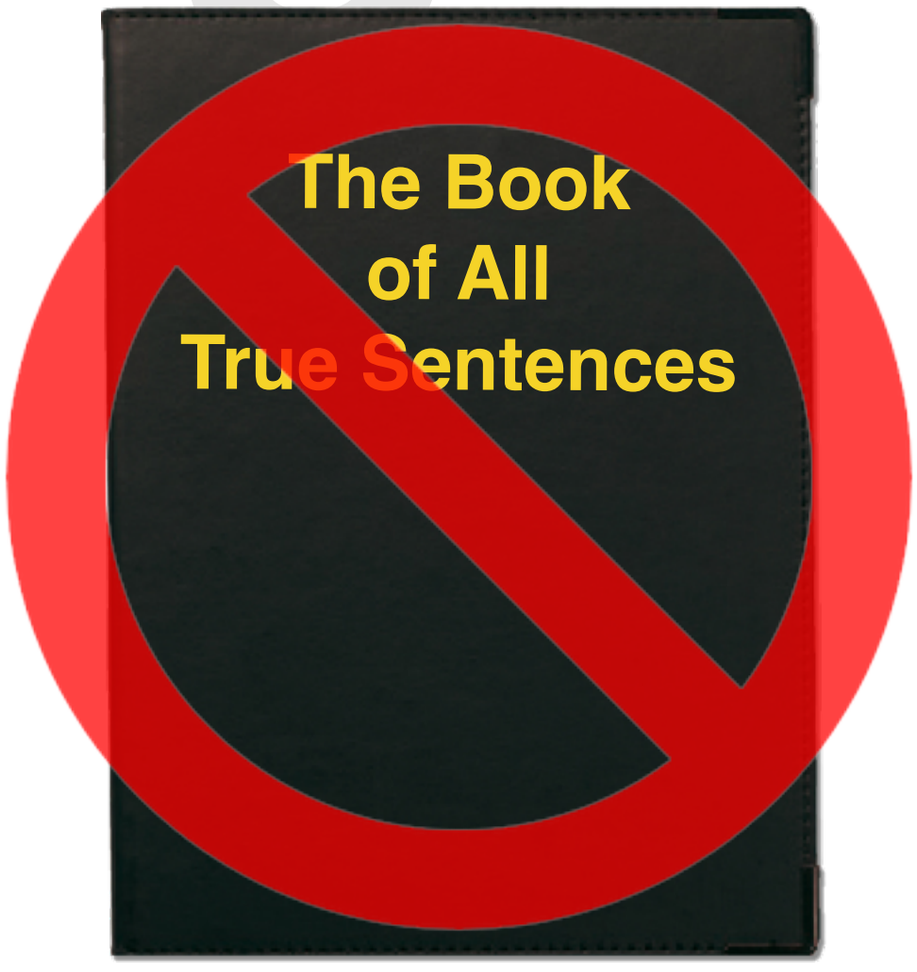
inconsistent

http://fouryears.eu/2013/01/27/on-godels-theorems-and-the-universe/

Language

http://fouryears.eu/2013/01/27/on-godels-theorems-and-the-universe/

Language



Because it would be infinitely large

http://fouryears.eu/2013/01/27/on-godels-theorems-and-the-universe/

Language

Axioms

THEN-1: $A \rightarrow (B \rightarrow A)$

THEN-2: $(A \rightarrow (B \rightarrow C)) \rightarrow ((A \rightarrow B) \rightarrow (A \rightarrow C))$

THEN-3: $(A \rightarrow (B \rightarrow C)) \rightarrow (B \rightarrow (A \rightarrow C))$

FRG-1: $(A \rightarrow B) \rightarrow (\neg B \rightarrow \neg A)$

FRG-2: $\neg\neg A \rightarrow A$

FRG-3: $A \rightarrow \neg\neg A$

Inference Rule

MP: $P, P \rightarrow Q \vdash Q$

http://fouryears.eu/2013/01/27/on-godels-theorems-and-the-universe/

Language

Axioms

THEN-1: A → (B → A)

THEN-2: (A → (B → C)) → ((A → B) → (A → C))

THEN-3: (A → (B → C)) → (B → (A → C))

FRG-1: (A → B) → (¬B → ¬A)

FRG-2: ¬¬A → A

FRG-3: A → ¬¬A

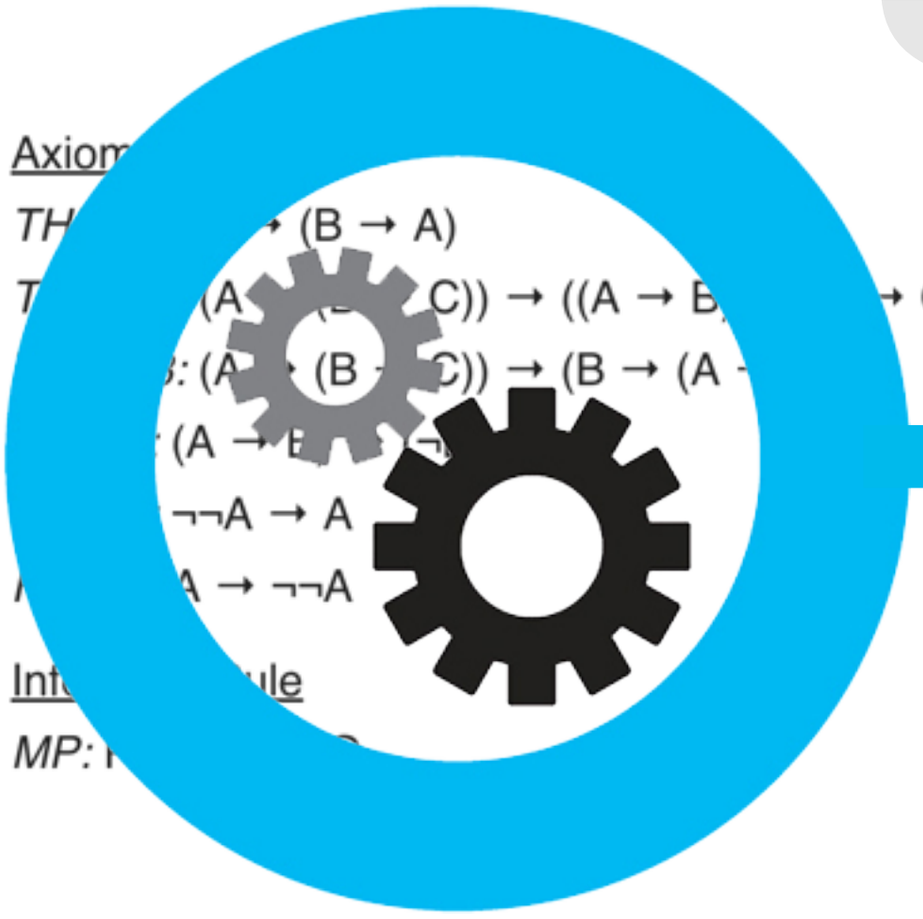
Inference Rule

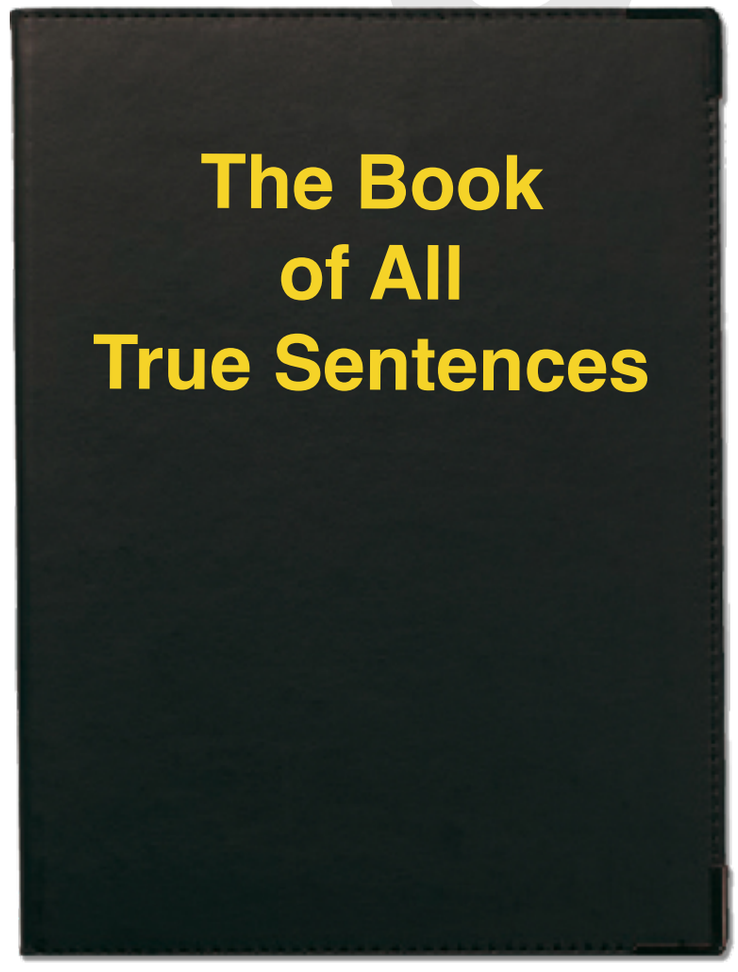
MP: P, P→Q ⊢ Q

#	wff	reason
1.	A→(B→C)	premise
2.	(A → (B → C)) → ((A → B) → (A → C))	THEN-2
3.	(A→B)→(A→C)	MP 1,2.

http://fouryears.eu/2013/01/27/on-godels-theorems-and-the-universe/

Language





#	wff	reason
1.	A→(B→C)	premise
2.	(A → (B → C)) → ((A → B) → (A → C))	THEN-2
3.	(A→B)→(A→C)	MP 1,2.

http://fouryears.eu/2013/01/27/on-godels-theorems-and-the-universe/

Language

“This sentence will not be included in the book generated by our language."

http://fouryears.eu/2013/01/27/on-godels-theorems-and-the-universe/

Language

"This sentence will not be included in the book generated by our language."



http://fouryears.eu/2013/01/27/on-godels-theorems-and-the-universe/

Puzzle

a*lphabet:* ~ P N ( )

Puzzle

a*lphabet:* ~ P N ( )

expression: finite non-empty string of these symbols

Puzzle

alphabet: ~ P N ( )

expression: finite non-empty string of these symbols

printable: machine can and will print it

Puzzle

alphabet: ~ P N ( )

expression: finite non-empty string of these symbols

printable: machine can and will print it

norm of X: expression in form X(X), for example P(P)

Puzzle

alphabet: ~ P N ( )

expression: finite non-empty string of these symbols

printable: machine can and will print it

norm of X: expression in form X(X), for example P(P)

sentence:

(1)	P(X)	true iff X is printable
(2)	PN(X)	true iff norm of X is printable
(3)	~P(X)	true iff X is not printable
(4)	~PN(X)	true iff norm of X is not printable

alphabet: ~ P N ( )

expression: finite non-empty string of these symbols

printable: machine can and will print it

norm of X: expression in form **X(X)**, for example **P(P)**

sentence:

(1)	**P(X)**	true iff **X** is printable
(2)	**PN(X)**	true iff norm of **X** is printable
(3)	**~P(X)**	true iff **X** is not printable
(4)	**~PN(X)**	true iff norm of **X** is not printable

All sentences machine prints are true

If machine prints **P(X)** then **X** is printable

If machine prints **PN(X)** then **X(X)** is printable

If machine prints **X** we don't know if **P(X)** is printable

alphabet: ~ P N ( )

expression: finite non-empty string of these symbols

printable: machine can and will print it

norm of X: expression in form **X(X)**, for example **P(P)**

sentence:

(1)	**P(X)**	true iff **X** is printable
(2)	**PN(X)**	true iff norm of **X** is printable
(3)	**~P(X)**	true iff **X** is not printable
(4)	**~PN(X)**	true iff norm of **X** is not printable

All sentences machine prints are true

If machine prints **P(X)** then **X** is printable

If machine prints **PN(X)** then **X(X)** is printable

If machine prints **X** we don't know if **P(X)** is printable

Find a sentence which is true but not printable.

Raymond M. Smullyan - Godel's Incompleteness Theorems



~PN(~PN)

~PN(~PN)

~PN(~PN) is true iff norm of **~PN** is not printable.

~PN(~PN)

~PN(~PN) is true iff norm of **~PN** is not printable.

~PN(~PN) is true iff **~PN(~PN)** is not printable.

~PN(~PN)

~PN(~PN) is true iff norm of **~PN** is not printable.

~PN(~PN) is true iff **~PN(~PN)** is not printable.

If machine prints **~PN(~PN)** then is has printed false sentence.

~PN(~PN)

~PN(~PN) is true iff norm of **~PN** is not printable.

~PN(~PN) is true iff **~PN(~PN)** is not printable.

If machine prints **~PN(~PN)** then is has printed false sentence.

If machine never prints **~PN(~PN)** then is has failed to print a true sentence.

~PN(~PN)

~PN(~PN) is true iff norm of **~PN** is not printable.

~PN(~PN) is true iff **~PN(~PN)** is not printable.

If machine prints **~PN(~PN)** then is has printed false sentence.

If machine never prints **~PN(~PN)** then is has failed to print a true sentence.

We have found a true sentence which will be never printed by the machine.

Provable

“Printable” → “Provable”

Provable

“Printable” → “Provable”

~PN(~PN)

The norm of ~PN is not provable within the formal system

Provable

“Printable” → “Provable”

~PN(~PN)

The norm of ~PN is not provable within the formal system

If you can prove **~PN(~PN)** within the system (using system’s rules) then you have proven false claim and your system is **inconsistent**.

Raymond M. Smullyan - Godel's Incompleteness Theorems

Provable

“Printable” → “Provable”

~PN(~PN)

The norm of ~PN is not provable within the formal system

If you can prove **~PN(~PN)** within the system (using system’s rules) then you have proven false claim and your system is **inconsistent**.

If you cannot prove **~PN(~PN)** then there is a true fact which is not possible to prove within your system. Thus your system is **incomplete**.

Raymond M. Smullyan - Godel's Incompleteness Theorems

Any effectively generated **theory** capable of expressing elementary arithmetic cannot be both **consistent** and **complete**. In particular, for any consistent, effectively generated formal theory that proves certain basic arithmetic truths, **there is an arithmetical statement that is true but not provable** in the theory.

http://en.wikipedia.org/wiki/G%C3%B6del%27s_incompleteness_theorems

Any effectively generated **theory** capable of expressing elementary arithmetic cannot be both **consistent** and **complete**. In particular, for any consistent, effectively generated formal theory that proves certain basic arithmetic truths, **there is an arithmetical statement that is true but not provable** in the theory.

http://en.wikipedia.org/wiki/G%C3%B6del%27s_incompleteness_theorems

Theorem VI: *For every* ω*-consistent primitive recursive class* κ *of* FORMULAE *there is a primitive recursive* CLASS-SIGN r *such that neither* $forall(v, r)$ *nor* $not(forall(v, r))$ *belongs to* $Conseq(\kappa)$ *(where* v *is the* FREE VARIABLE *of* r*).*

http://www.research.ibm.com/people/h/hirzel/papers/canon00-goedel.pdf

Any effectively generated **theory** capable of expressing elementary arithmetic cannot be both **consistent** and **complete**. In particular, for any consistent, effectively generated formal theory that proves certain basic arithmetic truths, **there is an arithmetical statement that is true but not provable** in the theory.

http://en.wikipedia.org/wiki/G%C3%B6del%27s_incompleteness_theorems

Theorem VI: *For every* ω*-consistent primitive recursive class* κ *of* FORMULAE *there is a primitive recursive* CLASS-SIGN r *such that neither* $forall(v, r)$ *nor* $not(forall(v, r))$ *belongs to* $Conseq(\kappa)$ *(where* v *is the* FREE VARIABLE *of* r*).*

http://www.research.ibm.com/people/h/hirzel/papers/canon00-goedel.pdf

Proposition VI: To every ω-consistent recursive class c of *formulae* there correspond recursive class-signs r, such that neither v Gen r nor Neg (v Gen r) belongs to Flg (c) (where v is the free variable of r).

http://jacqkrol.x10.mx/assets/articles/godel-1931.pdf

Any effectively generated **theory** capable of expressing elementary arithmetic cannot be both **consistent** and **complete**. In particular, for any consistent, effectively generated formal theory that proves certain basic arithmetic truths, **there is an arithmetical statement that is true but not provable** in the theory.

http://en.wikipedia.org/wiki/G%C3%B6del%27s_incompleteness_theorems

Theorem VI: *For every* ω*-consistent primitive recursive class* κ *of* FORMULAE *there is a primitive recursive* CLASS-SIGN r *such that neither* $forall(v, r)$ *nor* $not(forall(v, r))$ *belongs to* $Conseq(\kappa)$ *(where* v *is the* FREE VARIABLE *of* r*).*

http://www.research.ibm.com/people/h/hirzel/papers/canon00-goedel.pdf

Proposition VI: To every ω-consistent recursive class c of *formulae* there correspond recursive class-signs r, such that neither v Gen r nor Neg (v Gen r) belongs to Flg (c) (where v is the free variable of r).

http://jacqkrol.x10.mx/assets/articles/godel-1931.pdf

Satz VI: Zu jeder ω-widerspruchsfreien rekursiven Klasse $\varkappa$ von *Formeln* gibt es rekursive *Klassenzeichen* r, so daß weder v Gen r noch Neg (v Gen r) zu Flg ($\varkappa$) gehört (wobei v die *freie Variable* aus r ist).

http://www.w-k-essler.de/pdfs/goedel.pdf

this slide intentionally left blank



“Book” intuition ✓



“Book” intuition ✓



~ **P N** ( ) system ✓



“Book” intuition ✓



~ P N ( ) system ✓



Gödel’s theorem with shades of Tarski ✓



“Book” intuition ✓



~ P N ( ) system ✓



Gödel’s theorem with shades of Tarski ✓



Assuming Peano Arithmetic is *correct* ✓



“Book” intuition ✓



~ P N ( ) system ✓



Gödel’s theorem with shades of Tarski ✓



Assuming Peano Arithmetic is *correct* ✓



Assuming Peano Arithmetic is ω-consistent ?

Consistency:
There is **no** formula F such that **both** F and ¬F are **provable**

Consistency:
There is **no** formula F such that **both** F and ¬F are **provable**

ω-inconsistency
If for some formula A(x), **each** formula of the infinite sequence A(0), …, A(n) **is provable** and **also** the formula **¬∀xA(x) is provable**

Consistency:
There is **no** formula F such that **both** F and ¬F are **provable**

ω-inconsistency
If for some formula A(x), **each** formula of the infinite sequence A(0), …, A(n) **is provable** and **also** the formula **¬∀xA(x) is provable**

This may not lead directly to an outright contradiction, because a theory may not be able to prove for any specific value of n that A(n) fails, only that there is such an n

Consistency:
There is **no** formula F such that **both** F and ¬F are **provable**

ω-inconsistency
If for some formula A(x), **each** formula of the infinite sequence A(0), …, A(n) **is provable** and **also** the formula **¬∀xA(x) is provable**

This may not lead directly to an outright contradiction, because a theory may not be able to prove for any specific value of n that A(n) fails, only that there is such an n

Theorem VI: *For every ω-consistent primitive recursive class κ of* FORMULAE *there is a primitive recursive* CLASS-SIGN *r such that neither forall(v, r) nor not(forall(v, r)) belongs to Conseq(κ) (where v is the* FREE VARIABLE *of r).*

2nd

If a **formal system** which contains elementary arithmetic (like Peano Arithmetic) **is consistent**, then it **cannot prove its own consistency**.

Consistent formal system is incomplete

Consistent formal system is incomplete



Some *stuff* is not provable

Consistent formal system is incomplete

↓

Some *stuff* is not provable

↓

For example claim of its own consistency is one such *stuff*

§4. Fixed Points and Gödel Sentences. For any expression X, we define $\overline{X}$ to be the numeral designating the Gödel number of X. Thus, for any formula $F(v_1)$ and any expression X, $F(\overline{X})$ is $F(\overline{x})$, where x is the Gödel number of X.

A sentence X is called a *fixed point* of a formula $F(v_1)$ (in a system $\mathcal{S}$) if the sentence $X \equiv F(\overline{X})$ is provable in $\mathcal{S}$.

Theorem 4. *If the diagonal function $d(x)$ is strongly definable in $\mathcal{S}$, then every formula $F(v_1)$ has a fixed point (in $\mathcal{S}$).*

Theorem 1. *If G is a fixed point of the formula $\sim P(v_1)$ and $\mathcal{S}$ is consistent, then G is not provable in $\mathcal{S}$.*

Proof. We are given that $G \equiv \sim P(\overline{G})$ is provable in $\mathcal{S}$. Now suppose G were provable in $\mathcal{S}$. Then $\sim P(\overline{G})$ and $P(\overline{G})$ would be provable in $\mathcal{S}$ (by property P_1). Hence $\mathcal{S}$ would be inconsistent. So if $\mathcal{S}$ is consistent, then G is not provable in $\mathcal{S}$.

Theorem 2. *If G is a fixed point of the formula $\sim P(v_1)$, then the sentence consis $\supset G$ is provable in $\mathcal{S}$.*

Proof. We are given that the sentence $G \equiv \sim P(\overline{G})$ is provable in $\mathcal{S}$. Since f is a refutable sentence of $\mathcal{S}$, then $\sim P(\overline{G}) \equiv (P(\overline{G}) \supset f)$ is provable in $\mathcal{S}$ and, therefore, $G \equiv (P(\overline{G}) \supset f)$ is provable in $\mathcal{S}$, and so $G \supset (P(\overline{G}) \supset f)$ is provable. Then by property P_6, the sentence $P(\overline{G}) \supset P(\overline{f})$ is provable. Hence $\sim P(\overline{f}) \supset \sim P(\overline{G})$ is provable, and

Theorem 3—An Abstract Form of Gödel's Second Theorem. Suppose $\mathcal{S}$ is diagonalizable. Then if $\mathcal{S}$ is consistent, then the sentence consis is not provable in $\mathcal{S}$.

Proof. Suppose $\mathcal{S}$ is diagonalizable. Then there is a sentence G such that $G \equiv \sim P(\overline{G})$ is provable in $\mathcal{S}$ and by Theorem 2, the sentence consis $\supset G$ is provable in $\mathcal{S}$. Now suppose consis were provable in $\mathcal{S}$. Then G would be provable in $\mathcal{S}$ and $\mathcal{S}$ would be inconsistent by Theorem 1 (since G is a fixed point of $\sim P(v_1)$). Therefore, if $\mathcal{S}$ is consistent, then the sentence consis is not provable in $\mathcal{S}$.

§1. Provability Predicates. A formula $P(v_1)$ is called a *provability predicate* for $\mathcal{S}$ if for all sentences X and Y the following three conditions hold:

P_1: If X is provable in $\mathcal{S}$, then so is $P(\overline{X})$.
P_2: $P(\overline{X \supset Y}) \supset (P(\overline{X}) \supset P(\overline{Y}))$ is provable in $\mathcal{S}$.
P_3: $P(\overline{X}) \supset P(P(\overline{X}))$ is provable in $\mathcal{S}$.

Problem 4—After Gödel's Second Theorem. An advanced reasoner visits the Island of Knights and Knaves and meets a native who tells him, "You will never believe I'm a knight." Prove that if the reasoner should believe that he is consistent, then he will become inconsistent. Stated otherwise, if the reasoner is (and remains) consistent, he can never know it.

And so, suppose the reasoner assumes his own consistency. Then he will sooner or later get into an inconsistency by going through the following argument: "Suppose I ever believe that the native is a knight. Then I'll believe what he said—I'll believe that I don't (and never will) believe he's a knight. But also, if I ever believe he's a knight, then I'll believe that I *do* believe he's a knight (since I am normal), which means I would be inconsistent! Now, since I am consistent (sic!), then I can never believe that he's a knight. He said I never would. Hence he's a knight."

At this point the reasoner believes that the native is a knight. Being normal, he continues, "Now I believe he's a knight. He said I never would. Hence he is a knave."

At this point the reasoner is clearly inconsistent.

Q: An advanced reasoner visits the Island of Knights and Knaves and meets a native who tells him, "You can not prove I'm a knight." Prove that if the reasoner should assume that he is consistent, then he will be come inconsistent. Stated otherwise, if the reasoner is (and remains) consistent, he can never know it.

A: And so, suppose the reasoner assumes his own consistency. Then he will sooner or later get into an inconsistency by going through the following argument: "Suppose I can prove that the native is a knight. Then I can prove what he said -- I can prove that I can't prove he's a knight. But also, if I can prove he's a knight, then I can prove that I can prove he's a knight (since I am normal), which means I would be inconsistent! Now, since I am consistent (sic!), then I can never prove that he's a knight. He said I never would. Hence he's a knight."

At this point the reasoner has proved that the native is a knight. Being normal, he continues, "Now I can prove he's a knight. He said I never would. Hence he is a knave."

At this point the reasoner is clearly inconsistent.