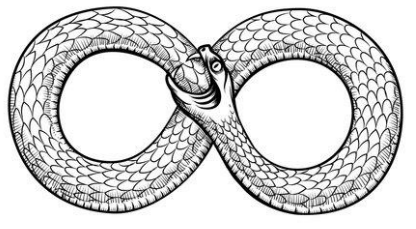
# G  del’s Incompleteness Theorems

Nicholas Ang

29 November 2023

## 1. Proving Mathematical Statements

*"Why do we want a formal system for Mathematics?"*

## 2. Axiomatic Systems

*"How do we construct such a system, and what are its desired properties?"*

## 3. Godel's Theorem

*"What limitations do we face in constructing such a system?"*

## 4. Implications

*"What effect does the Theorem have on Maths and Computer Science?"*

# 1. Mathematical Statements

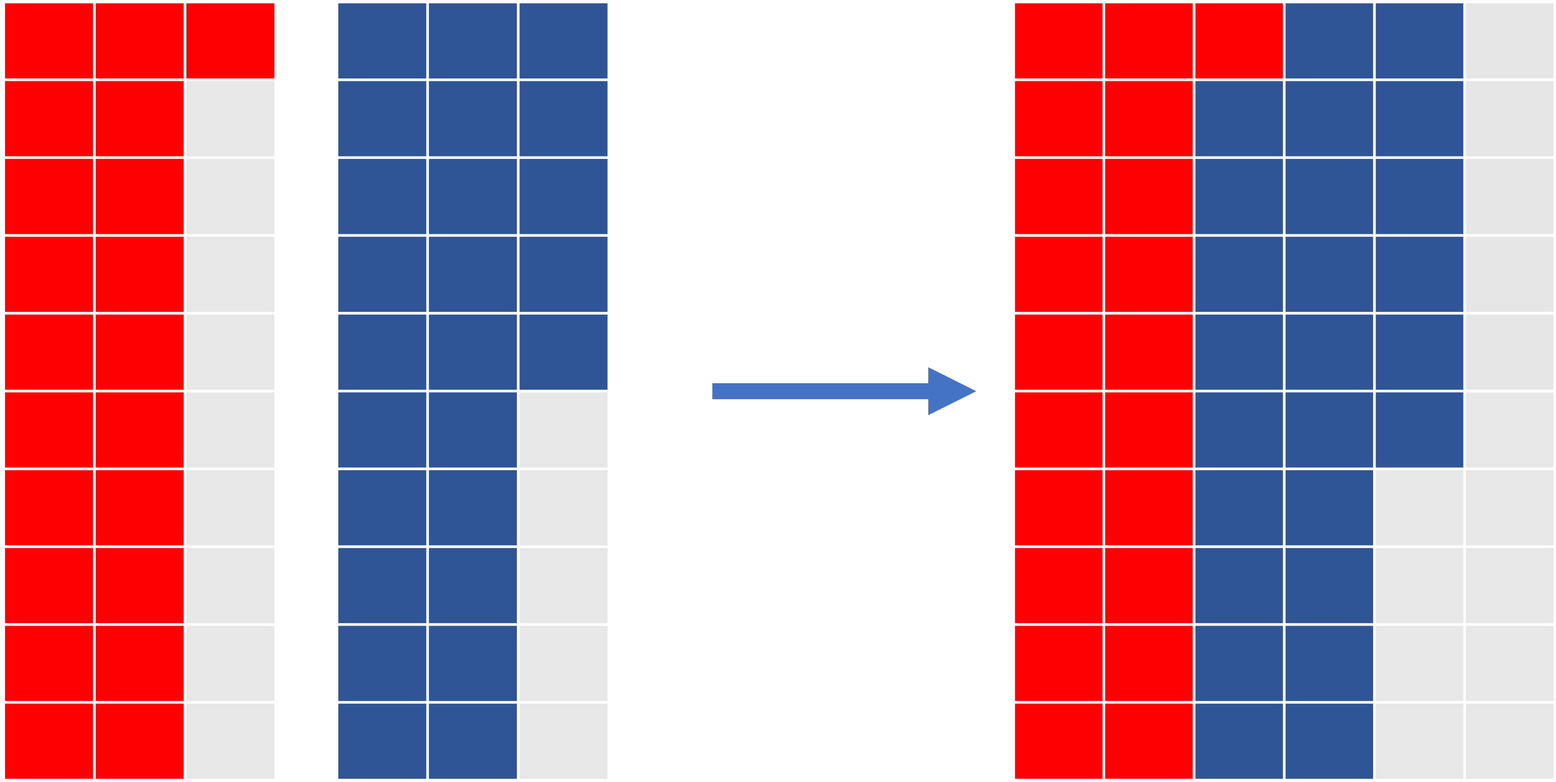
Why do we want to prove statements using a formal system?

# Mathematical Statements

- “21 + 25 = 46”
- “In a right-angled triangle, the sum of squares of the lengths of the shorter sides equals the sum of the hypotenuse squared”
- “The sum of angles in any triangle is 180 degrees”

# Mathematical Statements

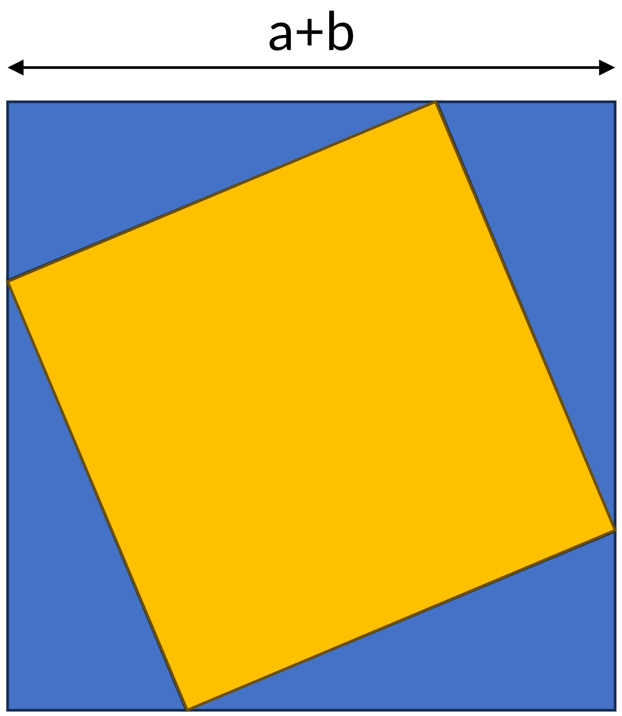
- “21 + 25 = 46”



# Mathematical Statements

- "In a right-angled triangle, the sum of squares of the lengths of the shorter sides equals the sum of the hypotenuse squared"

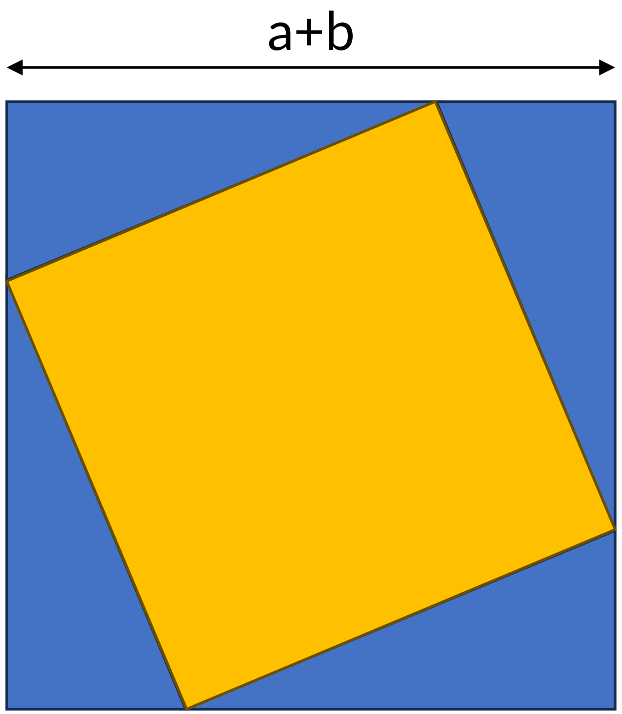
Area = (a+b)^2



# Mathematical Statements

- "In a right-angled triangle, the sum of squares of the lengths of the shorter sides equals the sum of the hypotenuse squared"

Area = (a+b)^2



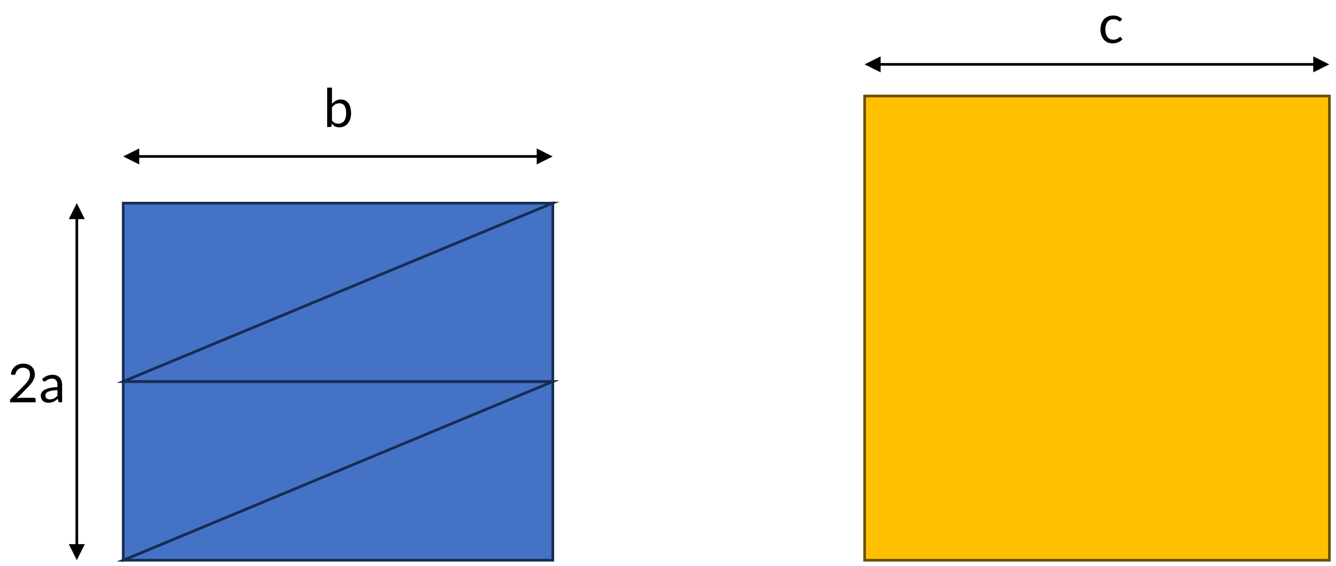
# Mathematical Statements

- “In a right-angled triangle, the sum of squares of the lengths of the shorter sides equals the sum of the hypotenuse squared”

Area = (a+b)^2

Area = 2ab + c^2

(a+b)^2 = 2ab + c^2
a^2 + 2ab + b^2 = 2ab + c^2
a^2 + b^2 = c^2



# Limitations

- Potential for ambiguity, as I am appealing to abstractions
  - Abstraction of counting (1, 2, 3 … 21, … 25)
  - Abstraction of geometry (square, triangle)
  - Abstraction of algebraic manipulation (distributive rule, …)

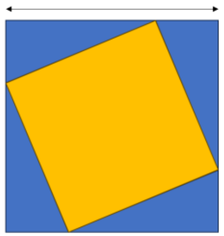
- "Intuitive" proofs of this sort become really difficult when it becomes difficult to visualise the abstractions
  - Complex numbers
  - Sets
  - Groups
  - etc…

- "Natural language proof" -> "Formal system proof"

# 2. Axiomatic Systems

How do we go about formalising Mathematics?

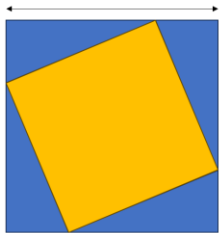
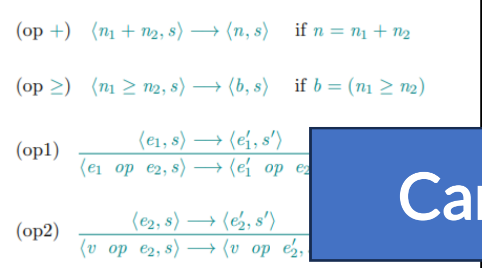
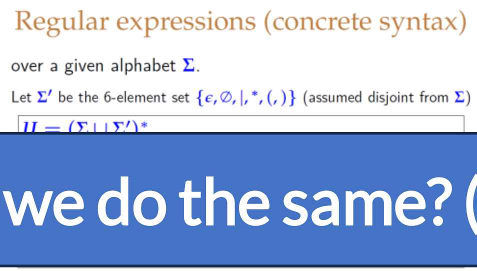
# We know how to formalise stuff already…

<table>
<tr><th>Programming</th><th>Pattern-Matching on Strings</th><th>Proving Things in Maths</th></tr>
<tr><td>Warmup<br>In C, if initially x has value 3, what's the value of the following?<br><code>x++ + x++ + x++ + x++</code></td><td>"Give me all the strings that have 3 a's"</td><td>Prove Pythagoras' Theorem</td></tr>
<tr><td>"Just type in my compiler and see the output / ask on StackOverflow"</td><td>Do you want…<br>- Only consecutive a's?<br>- Exactly 3 a's or (3 or more)?<br>- Small or capital 'a'…?</td><td>Area = (a+b)^2<br></td></tr>
<tr><td>(op +) $\langle n_1 + n_2, s\rangle \longrightarrow \langle n, s\rangle$ if $n = n_1 + n_2$<br>(op $\geq$) $\langle n_1 \geq n_2, s\rangle \longrightarrow \langle b, s\rangle$ if $b = (n_1 \geq n_2)$<br>(op1) $\dfrac{\langle e_1, s\rangle \longrightarrow \langle e_1', s'\rangle}{\langle e_1\ op\ e_2, s\rangle \longrightarrow \langle e_1'\ op\ e_2, s'\rangle}$<br>(op2) $\dfrac{\langle e_2, s\rangle \longrightarrow \langle e_2', s'\rangle}{\langle v\ op\ e_2, s\rangle \longrightarrow \langle v\ op\ e_2', s'\rangle}$<br><br>L1 Operational Semantics</td><td>Regular expressions (concrete syntax)<br>over a given alphabet $\Sigma$.<br>Let $\Sigma'$ be the 6-element set $\{\epsilon, \emptyset, |, *, (, )\}$ (assumed disjoint from $\Sigma$)<br>$U = (\Sigma \cup \Sigma')^*$<br>axioms: $\dfrac{}{a}$ $\dfrac{}{\epsilon}$ $\dfrac{}{\emptyset}$<br>rules: $\dfrac{r}{(r)}$ $\dfrac{r \quad s}{r|s}$ $\dfrac{r \quad s}{rs}$ $\dfrac{r}{r^*}$<br>(where $a \in \Sigma$ and $r, s \in U$)<br><br>(a|b)*a(a|b)*a(a|b)*a(a|b)*</td><td></td></tr>
</table>

Slide 27 from Discrete Finite Automata course (2022-2023).

Stajano, F. (2018-2023) *Formal Languages and Automata*

# We know how to formalise stuff already…

| Programming | Pattern-Matching on Strings | Proving Things in Maths |
|---|---|---|
| **Warmup**<br>In C, if initially x has value 3, what's the value of the following?<br>`x++ + x++ + x++ + x++` | "Give me all the strings that have 3 a's" | Prove Pythagoras' Theorem |
| "Just type in my compiler and see the output / ask on StackOverflow" | Do you want…<br>- Only consecutive a's?<br>- Exactly 3 a's or (3 or more)?<br>- Small or capital 'a'…? |  |
| <br>L1 Operational Semantics | <br>(a\|b)*a(a\|b)*a(a\|b)*a(a\|b)* | |

Can we do the same? (yes!)

Slide 27 from Discrete Finite Automata course (2022-2023).

Stajano, F. (2018-2023) *Formal Languages and Automata*

# Towards an Axiomatisation of Mathematics

| | **Regular Expressions** | **Mathematical Formalism** |
|---|---|---|
| Alphabet | Characters {a, b, c …} | Mathematical and Logical Symbols ( $=, x, 0, +, \dots, \rightarrow$, ^ ) |
| Axioms | Arbitrary string of characters | |
| Derivation Rules | Star rule, concat rule, … | |
| Deriving a string means that… | The string matches the regular expression. | |

# Towards an Axiomatisation of Mathematics

| | **Regular Expressions** | **Mathematical Formalism** |
|---|---|---|
| Alphabet | Characters {a, b, c …} | Mathematical and Logical Symbols ( $=$ , $x$, $0$, $+$, … , $\rightarrow$, ^ ) |
| Axioms | Arbitrary string of characters | |
| Derivation Rules | Star rule, concat rule, … | |
| Deriving a string means that… | The string matches the regular expression. | |

$x + 0 = x$

Syntactically Correct (Well-formed)

$x + +5 = ($

Not Syntactically Correct (Not Well-formed)

# Towards an Axiomatisation of Mathematics

| | Regular Expressions | Mathematical Formalism |
|---|---|---|
| Alphabet | Characters {a, b, c …} | Mathematical and Logical Symbols ( $=$ , $x$, $0$, $+$, … , $\rightarrow$, ^ ) |
| Axioms | Arbitrary string of characters | Mathematical Statements that are intuitively true<br>( $x + 0 = x$ )<br><br>Logical Statements (A ^ B -> A) |
| Derivation Rules | Star rule, concat rule, … | |
| Deriving a string means that… | The string matches the regular expression. | |

$x + 0 = x$

Syntactically Correct (Well-formed)

$x + +5 = ($

Not Syntactically Correct (Not Well-formed)

# Towards an Axiomatisation of Mathematics

| | **Regular Expressions** | **Mathematical Formalism** |
|---|---|---|
| Alphabet | Characters {a, b, c …} | Mathematical and Logical Symbols ( $=$ , $x$, $0$, $+$, … , $\rightarrow$, ^ ) |
| Axioms | Arbitrary string of characters | Mathematical Statements that are intuitively true<br>( $x + 0 = x$ )<br><br>Logical Statements (A ^ B -> A) |
| Derivation Rules | Star rule, concat rule, … | Modus Ponens |
| Deriving a string means that… | The string matches the regular expression. | |

$x + 0 = x$

Syntactically Correct (Well-formed)

$x + +5 = ($

Not Syntactically Correct (Not Well-formed)

# Towards an Axiomatisation of Mathematics

| | Regular Expressions | Mathematical Formalism |
|---|---|---|
| Alphabet | Characters {a, b, c …} | Mathematical and Logical Symbols ( $=$ , $x$, $0$, $+$, … , $\rightarrow$, ^ ) |
| Axioms | Arbitrary string of characters | Mathematical Statements that are intuitively true<br>( $x+0=x$ )<br><br>Logical Statements (A ^ B -> A) |
| Derivation Rules | Star rule, concat rule, … | Modus Ponens |
| Deriving a string means that… | The string matches the regular expression. | The statement is true. |

$x+0=x$

Syntactically Correct (Well-formed)

$x++5=($

Not Syntactically Correct (Not Well-formed)

# Peano Language

| Type | Symbol | Common Meaning |
|---|---|---|
| Constant | 0 | Zero element |
| Unary Operator | $S$ | Successor |
| Binary Operator | $+$ | Addition Operator |
| | $\times$ | Multiplication Operators |
| Logical Operator | $=$ | Equality |
| | $\neg$ | Negation |
| | $\wedge$ | And |
| | ( | Bracket |
| | ) | Bracket |
| | $\forall$ | For all |
| Free Variable | $x$ | Used to quantify over numbers |
| | $y$ | Used to quantify over numbers |
| | $z$ | Used to quantify over numbers |

The $\Rightarrow$, $\vee$, $\exists$ symbols can be expressed from the composition of the above symbols.

We can have many free variables and can define more operators (such as exponentiation) as we wish.

# Peano Axioms

| | | |
|---|---|---|
| Successor | PA1 | $\forall x \neg(Sx = 0)$ |
| | PA2 | $\forall x \forall y(Sx = Sy \Rightarrow x = y)$ |
| Addition | PA3 | $\forall x(x + 0 = x)$ |
| | PA4 | $\forall x \forall y(x + Sy \Rightarrow S(x + y))$ |
| Multiplication | PA3 | $\forall x(x \times 0 = 0)$ |
| | PA4 | $\forall x \forall y(x \times Sy \Rightarrow (x \times y) + x)$ |
| Induction Family of Axioms | PA(Ind) | $(\phi(0) \wedge \forall x(\phi(x) \Rightarrow \phi(Sx)) \Rightarrow \forall x \phi(x))$ |

# Consistency

$x + 0 = x$

$P(x) \rightarrow P(SS0)$

$Sx + y = S(x + y)$

$SS0 + 0 = SS0$

$\neg SS0 + 0 = SS0$

$Q$

$\neg Q$

…

Syntactically Correct (Well-formed)

$( ($

$x + +5 = ($

Not Syntactically Correct (Not Well-formed)

# Consistency

$x + 0 = x$

$P(x) \rightarrow P(SS0)$

$Sx + y = S(x + y)$

$SS0 + 0 = SS0$ ⋯ $\neg SS0 + 0 = SS0$

$Q$ ⋯ $\neg Q$

...

Syntactically Correct (Well-formed)

$( \, ($

$x + +5 = ($

Not Syntactically Correct (Not Well-formed)

# Consistency

$x + 0 = x$

$P(x) \rightarrow P(SS0)$

$Sx + y = S(x + y)$

$SS0 + 0 = SS0$ $\cdots$ $\neg SS0 + 0 = SS0$

$Q$ $\cdots$ $\neg Q$

...

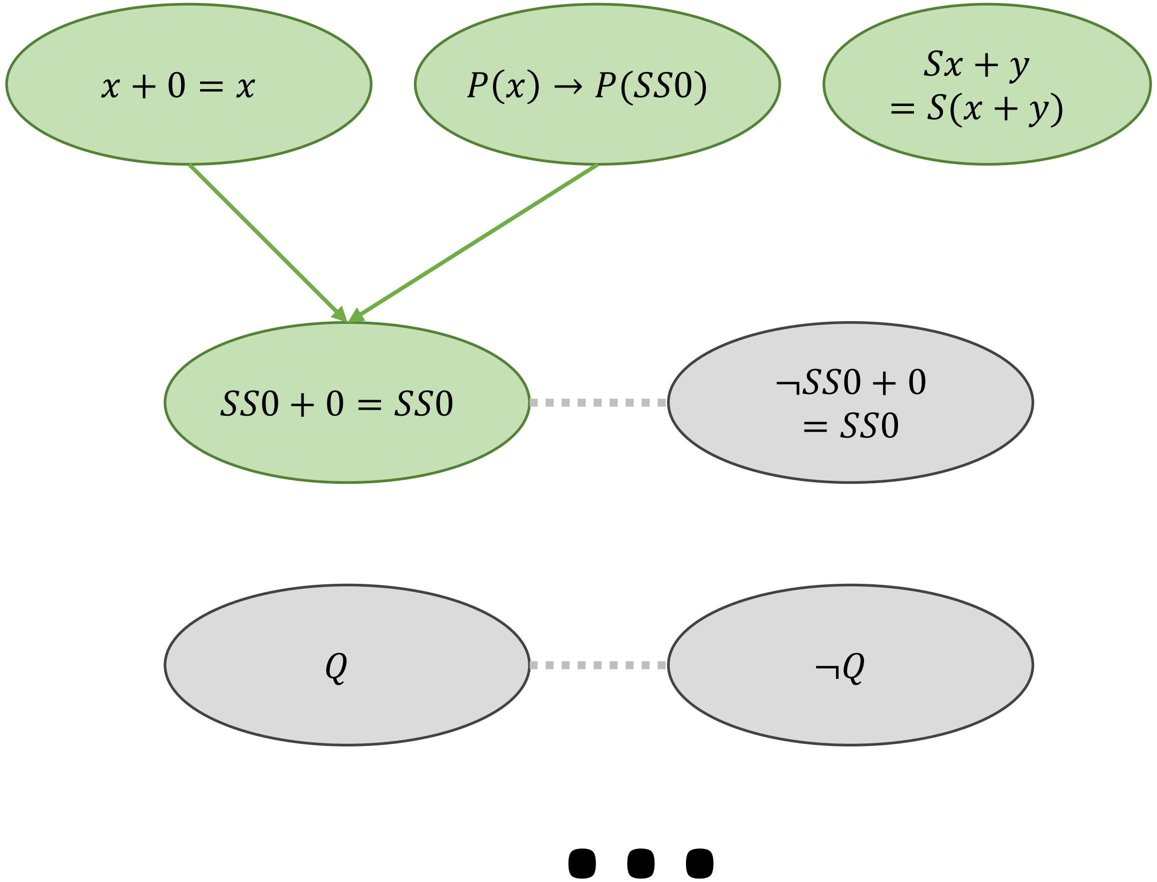
Syntactically Correct (Well-formed)

$( ($

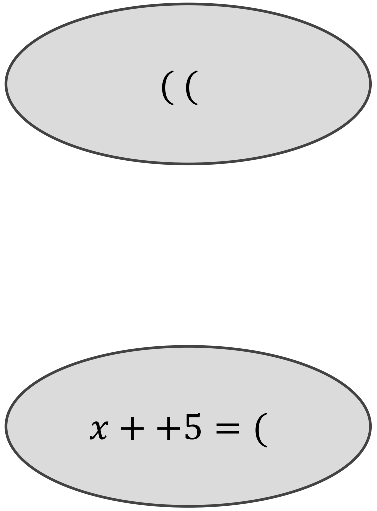
$x + +5 = ($

Not Syntactically Correct (Not Well-formed)

# Consistency

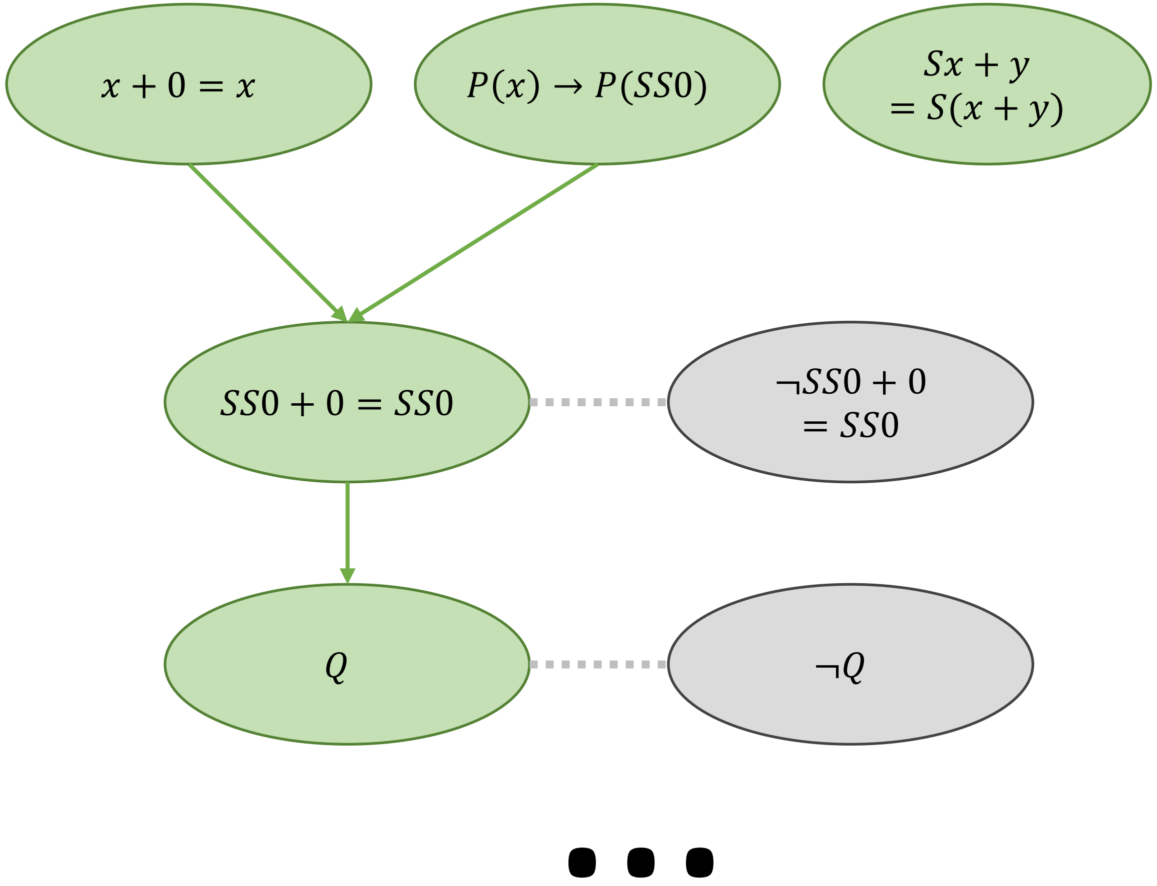



Syntactically Correct (Well-formed)

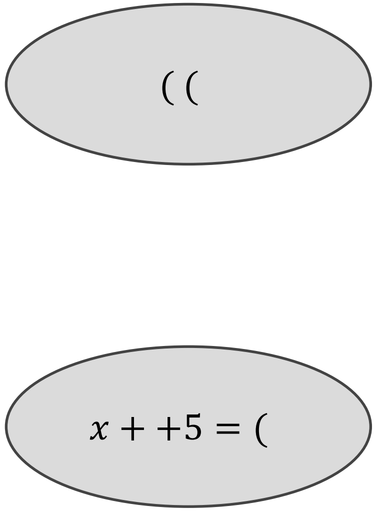



Not Syntactically Correct (Not Well-formed)

# Consistency

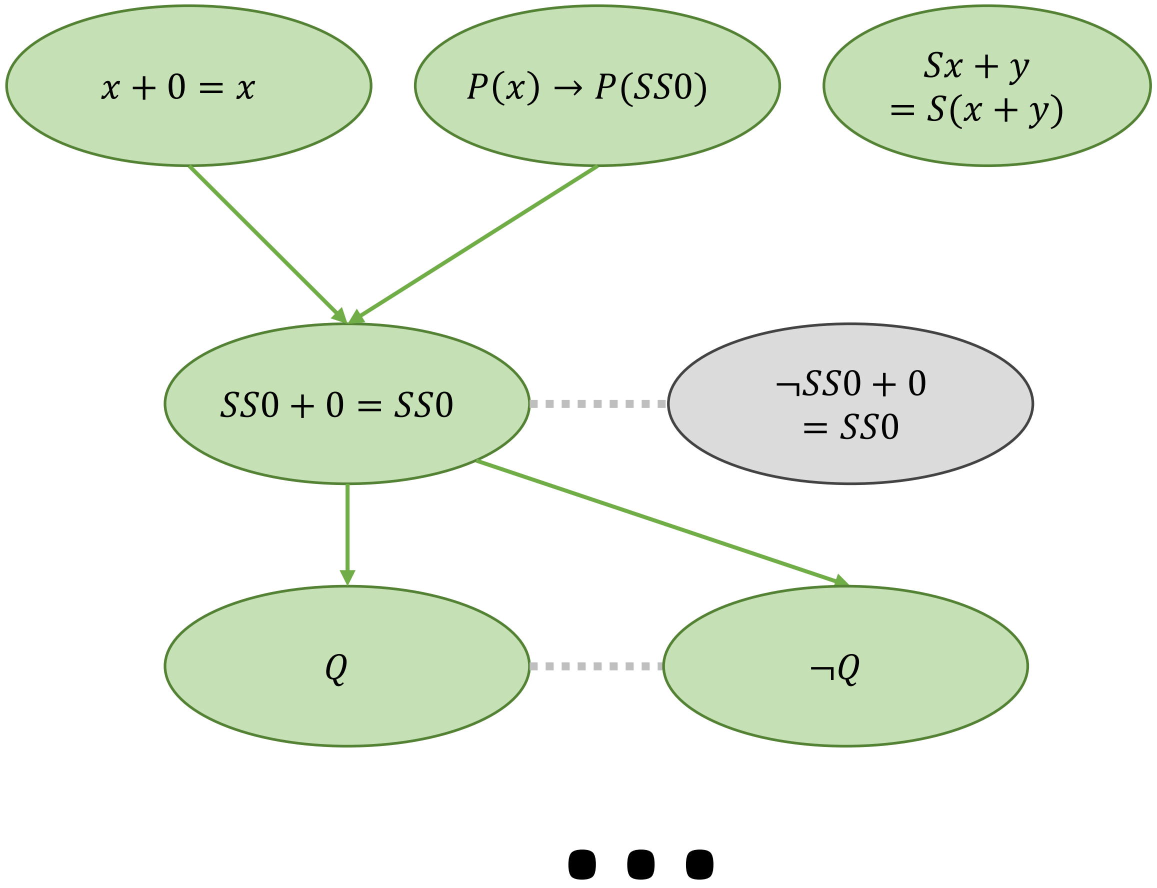



Syntactically Correct (Well-formed)




Not Syntactically Correct (Not Well-formed)

# Consistency

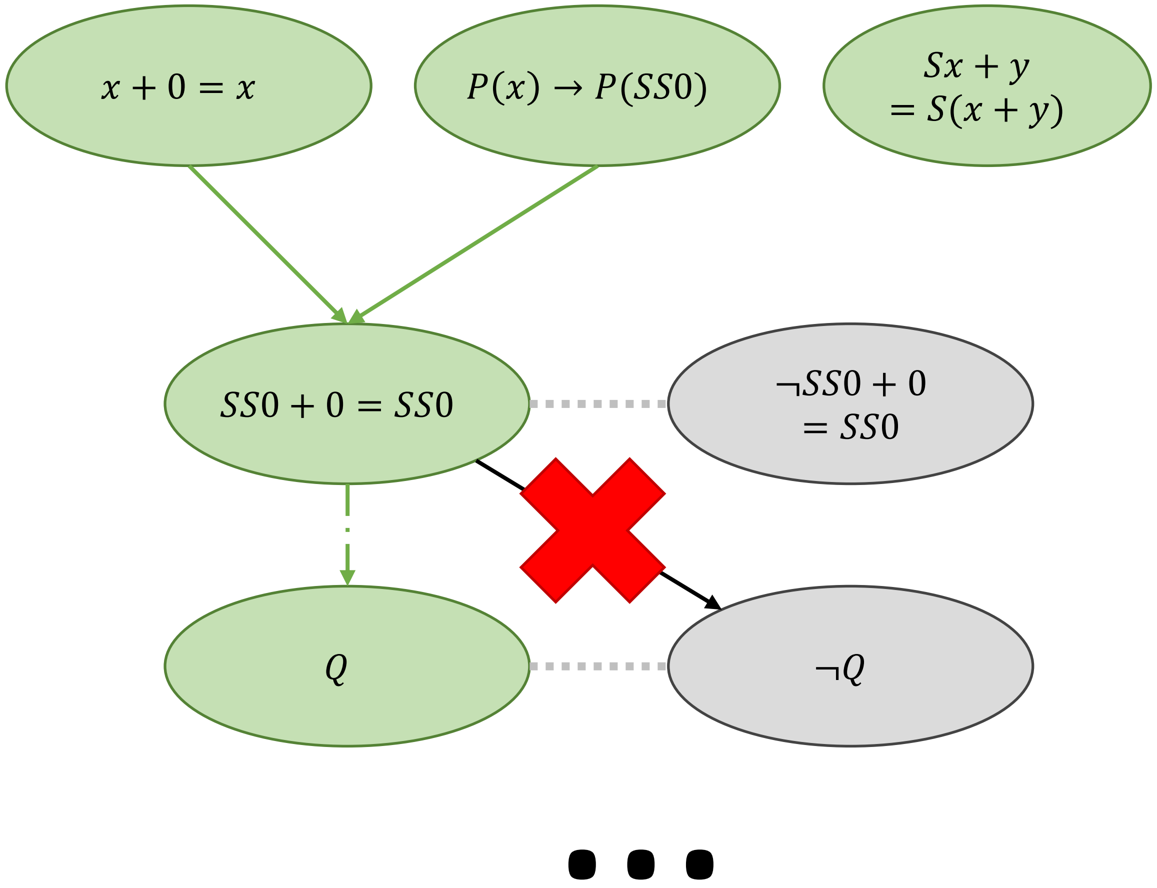



Syntactically Correct (Well-formed)

If there exists a statement Q for which both Q and Not-Q can be derived from the axioms, then the system is inconsistent.

Not Syntactically Correct (Not Well-formed)

# Consistency




Syntactically Correct (Well-formed)

In a consistent system, for every statement Q that can be derived from the axioms, it is impossible to derive the negation of Q .

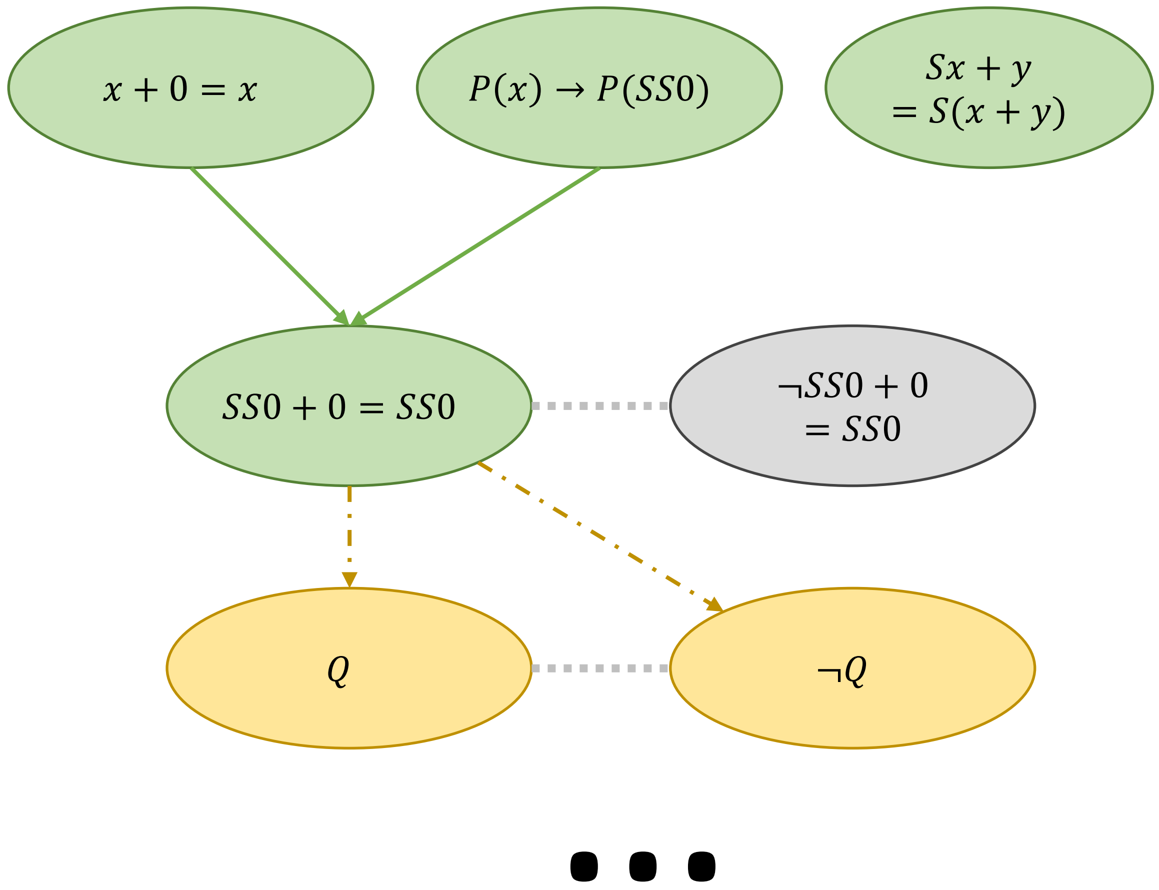
# Consistency

Imagine an inconsistent system that includes our typical laws of logic, where there exist proofs for A along with Not(A). Then the following derivation holds:

1. A is true.
2. Either (P is a prime number) or A is true.
3. Since A is not true, P must be a prime number for (2) to hold.

So any statement is provable in an inconsistent system (with Boolean logic axioms), which is undesirable!

# Completeness




Syntactically Correct (Well-formed)

In a complete system, there will always either be a derivation for Q or not Q (for every well-formed statement Q).

# 3. Godel’s Theorem

Can we construct an axiomatic system for the natural numbers that is both consistent and complete?

Step 1. Assume complete system

Step 2. Derive contradiction

Step 3. Profit

# Step 2. Derive contradiction

"This sentence is false!"

"I am lying!"

"The set of all sets that do not contain themselves contains itself"

## Step 2. Derive contradiction

"This sentence is false!"

"I am lying!"

"The set of all sets that do not contain themselves contains itself"

Self-reference

# Godel Numbering

| 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | $A$ | $B$ | $C$ |
|---|---|---|---|---|---|---|---|---|---|---|---|
| $\neg$ | $\bar{0}$ | ( | ) | $f$ | $'$ | $+$ | $\rightarrow$ | $\forall$ | $=$ | $x$ | $\#$ |

| | |
|---|---|
| A statement's Godel Number is a concatenation of the number assigned to each symbol, to form a new number. | The statement x+0=x will be assigned B72AB |
| A sequence of statements is the Godel number of each statement separated by a delimiter (e.g. C). | The sequence of statements x+0=1, x=x will be B72ABCBAB |



MathPunk (2022) *Godel's Incompleteness Theorems.* https://youtu.be/9GjCIU_ZHIo

# Provability Relation

| 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | $A$ | $B$ | $C$ |
|---|---|---|---|---|---|---|---|---|---|---|---|
| $\neg$ | $\bar{0}$ | ( | ) | $f$ | $'$ | $+$ | $\rightarrow$ | $\forall$ | $=$ | $x$ | # |

We can now convert statements about our sentence into arithmetic properties of numbers.

| **Property of a sentence** | **Property of a number** |
|---|---|
| The statement P is of the form Not(Q) where Q is another statement. | The first digit of the Godel Number of Q is 1 ($13^n < G < 2 * 13^n$ where n is the length of P) |
| X corresponds to a list of statements that are of the form Modus Ponens. | The Godel Number G(X) is of the form $\varphi C \varphi 8 \phi C \phi$ (some complicated inequality) |
| List of statements X provides a proof for statement Y. | $\big(G(X), G(Y)\big) \in Prov$ where $Prov$ is a relation over the natural numbers |

MathPunk (2022) *Godel's Incompleteness Theorems.* https://youtu.be/9GjCIU_ZHIo

# Godel Sentence

This is our Prov relation from before…

$$(\overline{X}, \overline{Y}) \in Prov \triangleq \text{X proves Y}$$

from which we can construct another relation, NP which roughly translates to

"n is not the Godel number of a proof of F instantiated with F's Godel's number"

$$(n, \overline{F(x)}) \in NP \triangleq (n, \overline{F(\bar{F})}) \notin Prov$$

allowing us to define the following statement that roughly translates to

"For the Godel number $\bar{X}$ representing statement $X$, there is no statement that proves $X(\bar{X})$"

$$\boxed{P(x) \triangleq \forall y.(y, x) \in NP}$$

# Godel Sentence

$$(\overline{X}, \overline{Y}) \in Prov \triangleq \text{X proves Y}$$
$$(n, \overline{F(x)}) \in NP \triangleq (n, \overline{F(\bar{F})}) \notin Prov$$
$$\boxed{P(x) \triangleq \forall y.(y, x) \in NP}$$

Assume complete: $\exists n \in \mathbb{N}.(n, \overline{P(\overline{P})}) \in Prov \vee (n, \overline{\neg P(\overline{P})}) \in Prov.$

$$\begin{aligned}
&\exists n \in \mathbb{N}.(n, \overline{P(\overline{P})}) \in Prov && \\
&(n, \overline{P}) \notin NP && \text{(by definition of NP)} \\
&\exists y.(y, \overline{P}) \notin NP) && \text{(by definition of exists)} \\
&\neg(\forall y.(y, \overline{P}) \in NP) && \text{(by definition of forall)} \\
&\forall y.(y, \overline{P}) \notin NP) && \\
&\neg P(\overline{P}) && \text{(by definition of P)}
\end{aligned}$$

This is a derivation of the negation, so our system is inconsistent.

# First Incompleteness Theorem (informally)

- **Any** axiomatic system (not just PA!) that includes statements which range over all natural numbers cannot be both complete and consistent.

# 4. Implications

Why does it matter?

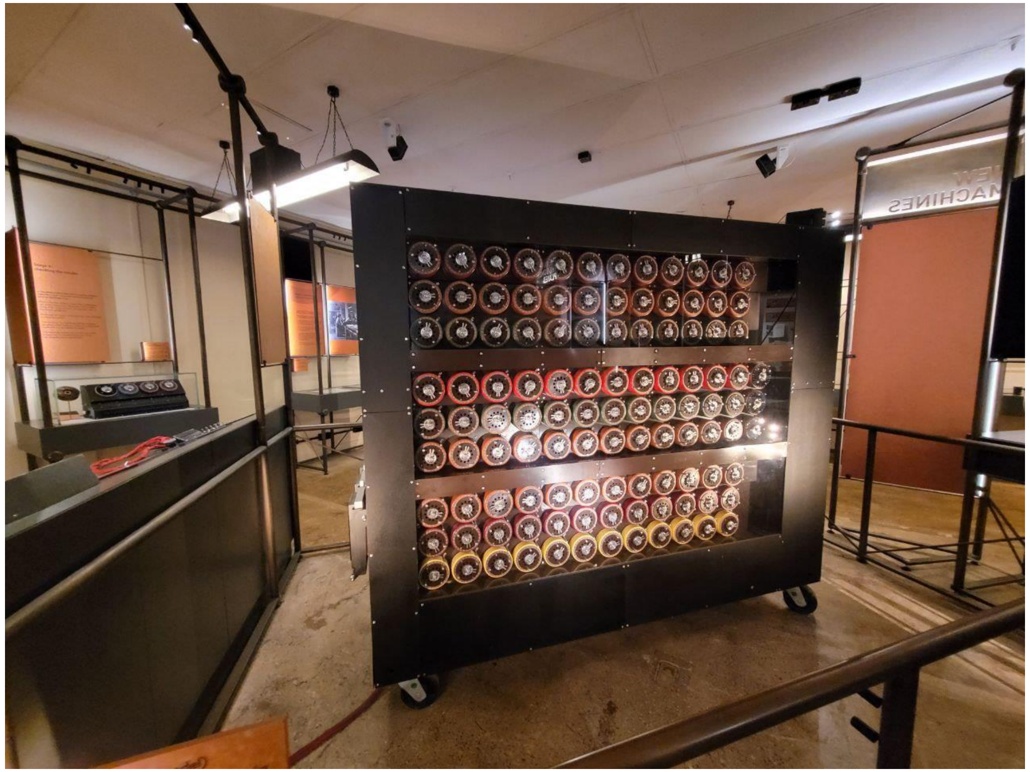
# There are unprovable statements

- Modern Mathematics chooses a consistent axiomatic system (typically ZFC) – so there are unprovable statements in ZFC

- Alternatively: abandon the formal framework (e.g. Mathematical Intuitionism)

# Computation Theory

- In the 1930s, Alan Turing applied the same idea to analysing models of computation



# Computation Theory

- Question: Can a sufficiently powerful computer solve all decision problems?

| Input | 1 | 2 | 3 | 4 | 5 | 6 | ... |
|---|---|---|---|---|---|---|---|
| Is the input a prime number? | No | Yes | Yes | No | Yes | No | ... |

| Input | a | aa | abb | bab | abba | ba | ... |
|---|---|---|---|---|---|---|---|
| Is the string a palindrome? | Yes | Yes | No | Yes | Yes | No | ... |

# Implications

- No, because there is no algorithm to decide if a program (written as a string) terminates

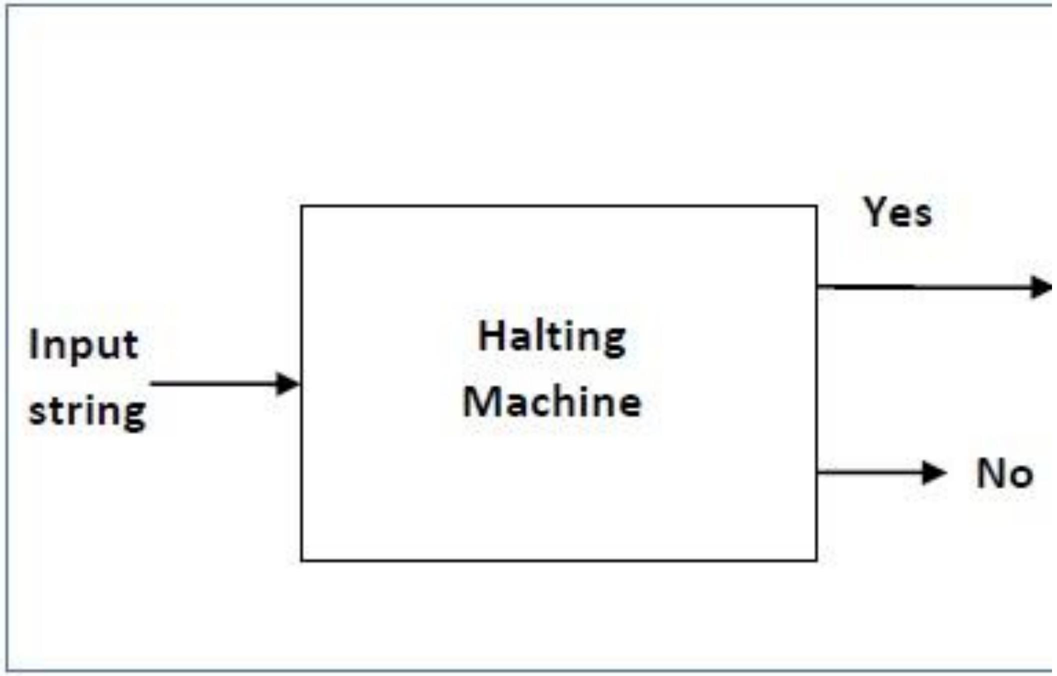
| Input | `while (True) {`<br>`    i = 1;`<br>`}`<br>`return;` | `i = 0;`<br>`while (i < 5) {`<br>`    i = i + 1;`<br>`}`<br>`return;` | … |
|---|---|---|---|
| Does this program terminate? | No | Yes | … |



Turing Machine Halting Problem (https://www.tutorialspoint.com/automata_theory/turing_machine_halting_problem.htm)

# Implications

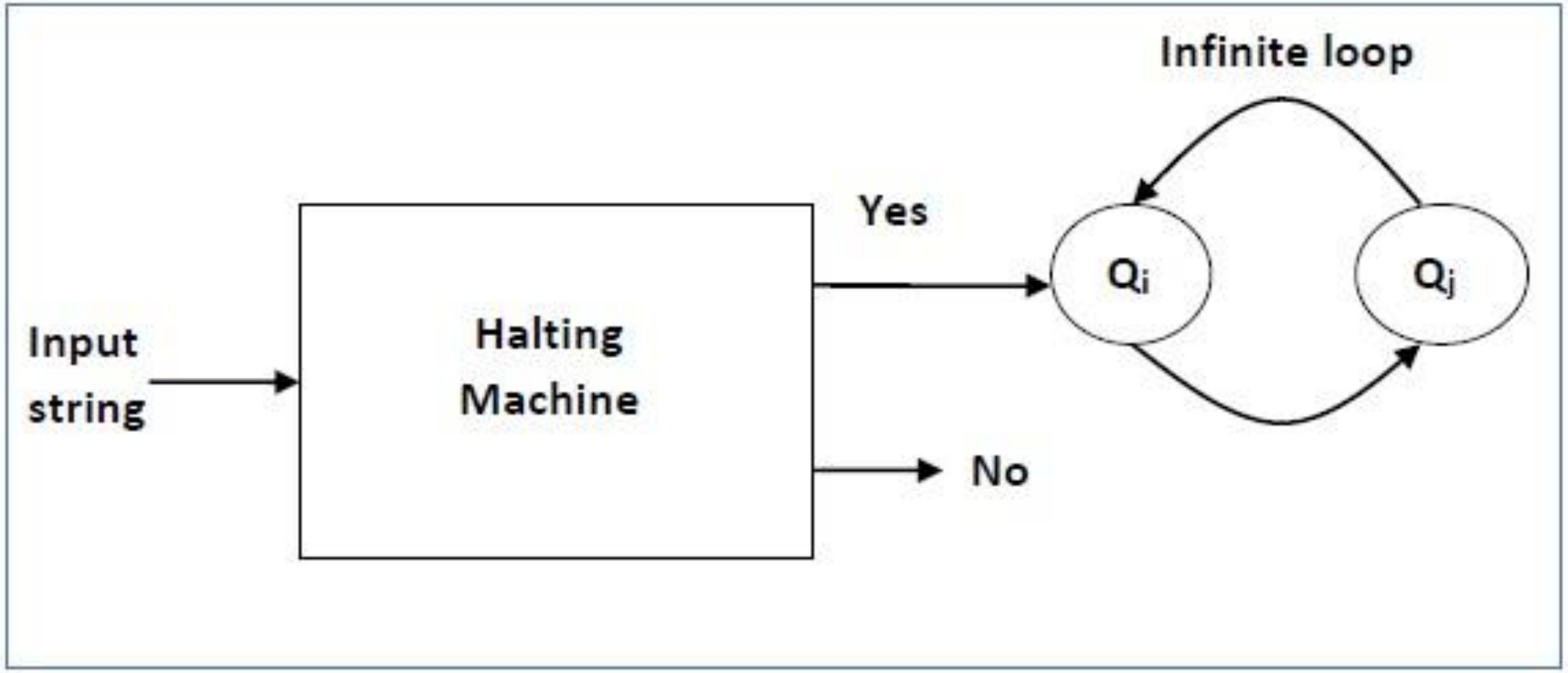
- Assume we have a machine that takes in a program input string S, and outputs (“Yes” or “No”) which is the answer to the Halting Problem



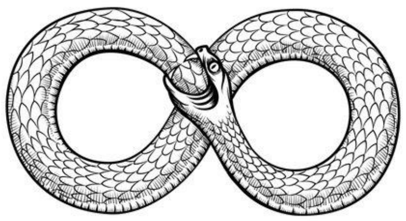

Turing Machine Halting Problem (https://www.tutorialspoint.com/automata_theory/turing_machine_halting_problem.htm)

# Implications

- Now we augment the machine, such that if the answer prints "Yes", we go into an infinite loop.
- What does the Machine do when supplied with its own source code?




Turing Machine Halting Problem (https://www.tutorialspoint.com/automata_theory/turing_machine_halting_problem.htm)



# Thank you!

Slides and further reading available at

https://github.com/angnicholas/godeltalk

# Mathematical Epistemology (1)

- Realism (Plato)
  - Mathematical statements appeal to a World of Forms (perfect sphere, perfect circle, perfect ‘one’, …)
- Logicism
  - Mathematical statements are an extension of systems of logic
- Formalism
  - Mathematical statements are true (relative to some axiomatic system) if they can be derived in that axiomatic system
- Intuitionism
  - Mathematical statements are dependent on appeals to mental constructs

# Mathematical Epistemology (2)

- Since the failure of Hilbert's Program demonstrates that a "perfect" formalisation of Mathematics is not possible, appeals to utility become more apparent
  - ZFC is "good enough" for our purposes

- Mathematics as a tool for modelling real-world abstractions ("unreasonable effectiveness") – largely driven forward by what is "useful"

# Provability Relation

| | |
|---|---|
| List of statements X provides a proof for statement Y. | $(G(X), G(Y)) \in Prov$ where $Prov$ is a relation over the natural numbers |

We look back at our earlier derivation:

This is in the relation:

This is also in the relation.

This is not in the relation: