1 Propositions

A mathematical statement is a clear declaration about pre-defined entities. Examples of statements:

$$0 = 1, \quad e^{i\pi} + 1 = 0, \quad 10001 \text{ is prime,} \quad \{ a \in \mathbb{R}^+ : \exists b \in \mathbb{R}, \, \zeta(a+ib) = 0 \} = \left\{ \frac{1}{2} \right\}.$$  

Mathematicians seek to generate a database of “true” statements that we can “absolutely” depend upon (not opinions), called “propositions” or “theorems”. The way to do this is by proofs, i.e., a process of showing the truth of a non-obvious statement using small, obvious steps. As the database of propositions grows, the proofs become ‘easier’.

Logic is the study of correct general demonstrations called inferences. Even if a conclusion is true, it need not follow from previous statements unless a correct inference holds. For example\(^1\), in the following arguments, the first group of statements may be true, but the conclusion doesn’t follow logically (even though they may be true).

<table>
<thead>
<tr>
<th>Some scientists are corrupt</th>
<th>Some mammals are dogs</th>
</tr>
</thead>
<tbody>
<tr>
<td>No corrupt person is good,</td>
<td>No dog is a cat</td>
</tr>
<tr>
<td>(\therefore) Some good people are not scientists</td>
<td>Some cats are not mammals</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>If today is Thursday</th>
<th>If (n) is a square</th>
</tr>
</thead>
<tbody>
<tr>
<td>then tomorrow is Friday.</td>
<td>then (n + 1) is not a square.</td>
</tr>
<tr>
<td>Tomorrow is Friday,</td>
<td>13 is not a square,</td>
</tr>
<tr>
<td>(\therefore) Today is Thursday</td>
<td>(\therefore) 12 is a square.</td>
</tr>
</tbody>
</table>

What is wrong here is the inference, not the truth of the statements themselves. Had the first inference been valid then the second inference would also be valid, even though the conclusion is clearly false. On the other hand, the following inferences are valid even though no one knows what the statements mean. To make the structure of the inference clearer, one can replace the instances with variables, as in the second group.

<table>
<thead>
<tr>
<th>Some harraps are kofs</th>
<th>Some (A) are (B)</th>
</tr>
</thead>
<tbody>
<tr>
<td>No weg is a kof</td>
<td>No (C) is (B)</td>
</tr>
<tr>
<td>(\therefore) Some harraps are notwegs</td>
<td>(\therefore) Some (A) are not (C).</td>
</tr>
</tbody>
</table>

A logical inference is a deduction that is valid whatever statements are substituted in the argument.

---

\(^1\)Aristotle
1.1 Primitives

Names (also called constants): defined ad hoc to represent any expression, e.g. \( M := 1000000 \); the symbol := is used to denote that we are using the left-hand letter to denote the right-hand expression. Typically, both left and right sides contain one or more matching variables, thus establishing a rule of substitution, e.g. \( \log(x) := \int_{1}^{x} \frac{1}{t} dt \).

Variables: \( x_0, x_1, \ldots, a, b, c, \ldots, A, B, C, \ldots \), as temporary tags for other symbols, e.g. \( n \) could stand temporarily for \( 0, 1, 2, \) etc. “Dummy variables” are not true variables but are used to help us read statements and definitions, for example \( \forall x \in \mathbb{R}, e^{ix} = \cos x + i \sin x \) is long-hand for something like \( \exp{\circ i} = \cos + i \sin \). In principle, one can use \( xx, xxx, \ldots \) to generate as many variable names as needed without the use of (yet undefined) subscripts or numerals. Greek letters such as \( \phi, \psi, \eta \) are here used to denote any statement.

Logic symbols: Composite statements are built from other statements \( \phi, \psi \), using the following symbols
\[
\phi \equiv \psi, \quad \phi \Rightarrow \psi, \quad \phi \lor \psi, \quad \phi \land \psi, \quad \text{NOT} \ \phi, \quad \forall x \ \phi_x, \quad \exists x \ \phi_x.
\]

Punctuation: \( ()\{\} : \), make expressions unambiguous but, like grammar, have no meaning. To avoid too many brackets, it is taken by default that for the first five logic symbols above, those to the right take precedence on those on the left, so \( \phi \Rightarrow \psi \lor \eta \) means \( \phi \Rightarrow (\psi \lor \eta) \); the last two symbols take the least precedence of all.

This article is concerned mostly with defining the logic symbols, making assumptions on what they mean and how to use them, starting with one logical proposition that we hold to be absolutely true, called an axiom, and from it deduce others (called tautologies). The aim is to have several forms of valid inferences.

1.2 Implications

We write \( \phi \vdash \psi \), or \( \phi \overset{\psi}{\Rightarrow} \), for a correct deduction of \( \psi \) from \( \phi \), by making substitutions and inferences. A ‘sequence’ of proofs \( \phi \vdash \psi \) and \( \psi \vdash \eta \) constitutes a proof \( \phi \vdash \eta \)

\[
\phi \vdash \psi, \psi \vdash \eta \quad \therefore \quad \phi \vdash \eta
\]

Definition The proposition \( \phi \Rightarrow \psi \) asserts that there is a deduction that shows \( \psi \) from \( \phi \),

\[
\phi \vdash \psi \quad \therefore \quad \phi \Rightarrow \psi
\]
Note that the bottom line is a statement, while the top one is a *meta*-statement, i.e., a statement about statements. The notation used here is as follows: when there is a single line, from the top statements one can infer the bottom one; when there is a double line, the bottom statement is equivalent to the top one, that is, an inference can be made in either direction.

The database can be started off with some statement \( \text{True} \) that does not require proof

\[ \vdash \text{True} \]

But, apart from the fact that it contains no information, it is more usual to take the following “obvious” axiom, namely that a given statement is sufficient proof for itself, \( \phi \vdash \phi \):

\[ \phi \Rightarrow \phi \]

where \( \phi \) can be any statement (whether true or false). One can then take \( \text{True} \) to be the vacuous statement \( \Rightarrow \phi \vdash \) is valid but not interesting; for example, \( \phi \Rightarrow \text{True} \) follows from \( \phi \vdash \text{True} \).

Note that, often, we need to keep track of which other propositions in the database were assumed in proving a proposition, so that \( \phi \Rightarrow \psi \) should really be qualified to mean that from \( \phi \) and these other true propositions \( L \), one can prove \( \psi \), that is, \( \phi \vdash \psi \) is equivalent to \( \frac{L}{\phi} \Rightarrow \psi \), i.e.,

\[ \frac{L, \phi \vdash \psi}{L \vdash (\phi \Rightarrow \psi)} \]

The previous definition of \( \phi \Rightarrow \psi \) is a special case when \( L \) is vacuous.

**Deductions:** Even with such few assumptions, we can already start deducing some (trivial) new inferences.

\[ \frac{\phi \Rightarrow \psi}{\phi \vdash \psi} \]

\[ \frac{L \vdash \phi, \psi \vdash M}{L, (\phi \Rightarrow \psi) \vdash M} \]

\[ \frac{\phi \Rightarrow \psi}{\psi \Rightarrow \chi \frac{\phi \Rightarrow \chi}{\phi \vdash \chi}} \]

**Proofs:**

(i) The first two statements are equivalent to one direction of the definition of \( \Rightarrow \): Given \( \phi \) and \( \phi \vdash \psi \), we can conclude \( \psi \); equivalently, given \( L \vdash \phi \) and \( \psi \vdash M \), then assuming \( L \) and \( \phi \Rightarrow \psi \) gives \( L \vdash \phi \vdash \psi \vdash M \).

(ii) Given proofs \( \phi \vdash \psi \) and \( \psi \vdash \chi \), then \( \phi \vdash \psi \vdash \chi \) constitutes a proof, \( \phi \vdash \chi \), of \( \phi \Rightarrow \chi \).

\[ \square \]
Definition \( \phi \Leftrightarrow \psi \) states that the statements \( \phi \Rightarrow \psi \) and \( \psi \Rightarrow \phi \) have been derived (or, equivalently, \( \phi \vdash \psi \) and \( \psi \vdash \phi \)),

\[
\frac{\phi \Rightarrow \psi \\
\psi \Rightarrow \phi \\
}{\phi \Leftrightarrow \psi}
\]

Proof: (i) \( \phi \Leftrightarrow \phi \) is short for a repeated \( \phi \Rightarrow \phi \).

(ii) From \( \phi \Leftrightarrow \psi \) we get \( \phi \Rightarrow \psi \) and \( \psi \Rightarrow \phi \), which when combined in the reverse order yields \( \psi \Leftrightarrow \phi \).

(iii) \( \phi \Rightarrow \psi \Rightarrow \eta \) and \( \eta \Rightarrow \psi \Rightarrow \phi \).

\( \square \)

The most common wrong inference is to take true propositions \( \phi \Rightarrow \psi \) and \( \psi \) and conclude \( \therefore \phi \), e.g. diseases cause symptoms, but having a symptom does not imply the disease (similarly from \( \phi \Rightarrow \psi \) and \( \neg \phi \) one cannot conclude \( \therefore \neg \psi \), “But I was so cautious!”)

Show

1. \( \phi \Rightarrow (\psi \Rightarrow \phi) \)
2. \( (\phi \Rightarrow (\psi \Rightarrow \xi)) \Leftrightarrow (\psi \Rightarrow (\phi \Rightarrow \xi)) \)
3. \( (\phi \Rightarrow (\phi \Rightarrow \psi)) \Rightarrow (\phi \Rightarrow \psi) \)

\[
\phi \Rightarrow \psi \\
\phi \Rightarrow \xi
\]

\[
\phi \Rightarrow (\psi \Rightarrow \xi)
\]

4. \( \phi \Rightarrow (\psi \Rightarrow \xi) \)

\[\phi \Rightarrow \xi\]

1.3 And

Definition The statement \( \phi \text{ AND } \psi \) denotes that the statements \( \phi \) and \( \psi \) have already been derived,

\[
\frac{\phi, \psi}{\phi \text{ AND } \psi}
\]

Again, the bottom line is a statement, while the top line is a meta-statement. The main assumption is the following:

\[
\frac{L \vdash \phi \quad L \vdash \psi}{L \vdash \phi, \psi}
\]
or equivalently,

\[
\frac{\eta \Rightarrow \phi \quad \eta \Rightarrow \psi}{\eta \Rightarrow (\phi \text{ AND } \psi)}
\]
From it follow:

\[(\phi, \psi) \vdash \phi \text{ and } (\phi, \psi) \vdash \psi \text{ (put } \phi, \psi \text{ for } L)\]

\[(\phi, \psi) \vdash (\psi, \phi) \text{ (from } (\phi, \psi) \vdash \psi \text{ and } (\phi, \psi) \vdash \phi \text{ deduce } (\phi, \psi) \vdash (\psi, \phi))\]

\[\phi \vdash (\phi, \phi) \text{ (from } \phi \vdash \phi \text{ repeated)}\]

Deductions:

\[
\begin{array}{c}
L \vdash \phi \text{ AND } \psi \\
\hline
L \vdash \phi
\end{array}
\quad
\begin{array}{c}
\phi \text{ AND } \psi \Rightarrow \phi
\end{array}
\quad
\begin{array}{c}
\phi \vdash M
\end{array}
\quad
\begin{array}{c}
\phi \text{ AND } \psi \vdash M
\end{array}
\]

The order and brackets of \textbf{AND} statements are not important, and are usually dropped:

\[
\begin{array}{c}
\phi \text{ AND } \phi \Leftrightarrow \phi
\end{array}
\quad
\begin{array}{c}
\phi \text{ AND } \psi \Leftrightarrow \psi \text{ AND } \phi
\end{array}
\quad
\begin{array}{c}
(\phi \text{ AND } \psi) \text{ AND } \chi \Leftrightarrow \phi \text{ AND } (\psi \text{ AND } \chi)
\end{array}
\]

Proofs: These are other forms of

\[
\phi \vdash (\phi, \phi) \vdash \phi, \quad \phi, \psi \vdash \psi, \phi, \quad (\phi, \psi), \chi \vdash \phi, (\psi, \chi)
\]

The last is due to \((\phi, \psi), \chi \vdash (\phi, \psi) \vdash \phi\) as well as \((\phi, \psi), \chi \vdash \psi \text{ and } (\phi, \psi), \chi \vdash \chi\). Combining the last two gives \((\phi, \psi), \chi \vdash (\psi, \chi)\) and then with the first gives the required conclusion. The reverse deduction is similar.

Show (\textbf{AND} has priority over \textbf{⇒} in bracketing)

\[
\begin{array}{c}
\phi \Rightarrow \psi
\end{array}
\]

1. \[
\begin{array}{c}
\phi \text{ AND } \psi \Rightarrow \xi
\end{array}
\]

2. \[
\begin{array}{c}
\phi \Rightarrow (\psi \Leftrightarrow \phi \text{ AND } \psi)
\end{array}
\]

1.4 \textbf{Or}

\textbf{Definition} The statement \(\phi \text{ OR } \psi\) asserts that any implied statement can be derived from \(\phi\) or from \(\psi\),

\[
\frac{\phi \vdash M \quad \psi \vdash M}{\phi \text{ OR } \psi \vdash M} \quad \text{ that is, } \quad \frac{\phi \Rightarrow \eta \quad \psi \Rightarrow \eta}{(\phi \text{ OR } \psi) \Rightarrow \eta}
\]

Deductions: First, as special cases of the definition one gets (start by putting \(\phi \text{ OR } \psi\) for \(M\))
In the reverse direction, we have

\[
\begin{align*}
\phi & \Rightarrow \phi \text{ OR } \psi \\
\phi \text{ OR } \psi & \Rightarrow \eta \\
\psi & \Rightarrow \eta \\
\therefore \eta & \text{ OR } \psi
\end{align*}
\]

Proof: (i) (Hobson’s choice) follows immediately from the definition of \( \phi \text{ OR } \psi \Rightarrow \eta \).

(ii) is a special case of (iii) with \( \psi \) substituted for \( \xi \). And (iii) is a special case of (i) by using \( \eta \Rightarrow \eta \text{ OR } \xi \) and \( \xi \Rightarrow \eta \text{ OR } \xi \).

\[\square\]

The order and brackets of \( \text{ OR } \) statements are not important:

\[
\begin{align*}
\phi \text{ OR } \phi & \Leftrightarrow \phi \\
\phi \text{ OR } \psi & \Leftrightarrow \psi \text{ OR } \phi \\
\phi \text{ OR } (\psi \text{ OR } \chi) & \Leftrightarrow (\phi \text{ OR } \psi) \text{ OR } \chi
\end{align*}
\]

Proofs: (i) is the definition with \( \phi \) instead of \( \psi \) and \( \eta \), using a repeated \( \phi \Rightarrow \phi \). The converse is a special case of \( \phi \Rightarrow \phi \text{ OR } \psi \).

(ii) From \( \phi \Rightarrow (\psi \text{ OR } \phi) \) and \( \psi \Rightarrow (\psi \text{ OR } \phi) \), we get \( \phi \text{ OR } \psi \Rightarrow \psi \text{ OR } \phi \). The converse is the same.

(iii) We know that

\( \phi \Rightarrow \phi \text{ OR } \psi \Rightarrow (\phi \text{ OR } \psi) \text{ OR } \chi \),

so we can deduce that \( \phi \Rightarrow (\phi \text{ OR } \psi) \text{ OR } \chi \). Similarly, \( \psi \Rightarrow (\phi \text{ OR } \psi) \text{ OR } \chi \) and also \( \chi \Rightarrow (\phi \text{ OR } \psi) \text{ OR } \chi \); hence \( \psi \text{ OR } \chi \Rightarrow (\phi \text{ OR } \psi) \text{ OR } \chi \). This means

\( \phi \text{ OR } (\psi \text{ OR } \chi) \Rightarrow (\phi \text{ OR } \psi) \text{ OR } \chi \).

Conversely,

\[
\begin{align*}
(\phi \text{ OR } \psi) \text{ OR } \chi & \\
\therefore \chi & \text{ OR } (\phi \text{ OR } \psi) \text{ by (ii)} \\
\therefore (\chi \text{ OR } \phi) & \text{ OR } \psi \text{ by (iii)} \\
\therefore \psi & \text{ OR } (\chi \text{ OR } \phi) \\
\therefore (\psi \text{ OR } \chi) & \text{ OR } \phi \\
\therefore \phi & \text{ OR } (\psi \text{ OR } \chi)
\end{align*}
\]
\[(\phi \text{ OR } \psi) \text{ AND } \eta \Leftrightarrow (\phi \text{ AND } \eta) \text{ OR } (\psi \text{ AND } \eta)\]
\[(\phi \text{ AND } \psi) \text{ OR } \eta \Leftrightarrow (\phi \text{ OR } \eta) \text{ AND } (\psi \text{ OR } \eta)\]

Proofs: From the left-hand side of (i), we deduce both \(\phi\) \text{ OR } \psi and \(\eta\); if we knew \(\phi\) then we would get \(\phi\) \text{ AND } \eta, so that \(\phi \Rightarrow \phi\) \text{ AND } \eta; similarly \(\psi \Rightarrow \psi\) \text{ AND } \eta; hence substituting in \(\phi\) \text{ OR } \psi we get \((\phi \text{ AND } \eta) \text{ OR } (\psi \text{ AND } \eta)\).

Now, suppose that \((\phi \text{ AND } \psi) \text{ OR } \eta\) is given; it is easy to show that
\[
\phi \text{ AND } \psi \Rightarrow (\phi \text{ OR } \eta) \text{ AND } (\psi \text{ OR } \eta),
\]
\[
\eta \Rightarrow (\phi \text{ OR } \eta) \text{ AND } (\psi \text{ OR } \eta),
\]
so that we deduce \((\phi \text{ OR } \eta) \text{ AND } (\psi \text{ OR } \eta)\).

For the converses, suppose we are given \((\phi \text{ AND } \eta) \text{ OR } (\psi \text{ AND } \eta)\); it follows, by application of the above, that \(((\phi \text{ AND } \eta) \text{ OR } \psi) \text{ AND } ((\phi \text{ AND } \eta) \text{ OR } \eta)\), and thus \((\phi \text{ OR } \psi) \text{ AND } (\eta \text{ OR } \psi) \text{ AND } (\phi \text{ OR } \eta) \text{ AND } \eta\), which in particular implies \((\phi \text{ OR } \psi) \text{ AND } \eta\). A similar reasoning reduces \((\phi \text{ OR } \eta) \text{ AND } (\psi \text{ OR } \eta)\) to \((\phi \text{ AND } \psi) \text{ OR } (\eta \text{ AND } \psi) \text{ OR } (\eta \text{ AND } \phi) \text{ OR } \eta\), but the middle terms can be combined as \((\phi \text{ AND } \psi) \text{ OR } \eta\) and the conclusion \((\phi \text{ AND } \psi) \text{ OR } \eta\) follows.

Show

1. \(\phi \text{ AND } \psi \Rightarrow \phi \text{ OR } \psi\).

2. \[
\frac{\phi \Rightarrow \psi}{\phi \Rightarrow (\psi \text{ OR } \eta)}
\]

3. If \(\phi \text{ OR } \psi\) means that we know that either \(\phi\) or \(\psi\) is in the database, what use could it possibly have? Three particles \(\oplus, x, \ominus\) are such that \(x\) is either \(\oplus\) or \(\ominus\), and two particles of the same sign repel; even if we don’t know what \(x\) is, or it’s impossible to know, or it changes with the situation, we can still deduce that in any case a particle will be repelled (though we can’t know which).

Just as the adjoint of “implication” is “and”, there is an adjoint of “or” that is closely related to “negation”:

\[
\frac{\phi \text{ BUT NOT } \psi \vdash \eta}{\phi \vdash \psi \text{ OR } \eta}
\]
1.5 Not

We are now going to suppose that not every statement is a proposition, i.e.,
that there is a statement which we call “contradiction” or False which is not
in the database.

The inference assumption is that we can deduce any statement from False,
(i.e., including it is tantamount to including everything in the database),

\[
\text{False} \Rightarrow \phi
\]

The following are immediate consequences:

- False or \(\phi\) \iff \(\phi\),  False and \(\phi\) \iff False
- True and \(\phi\) \iff \(\phi\),  True or \(\phi\) \iff True

Proofs: Essentially that False \(\Rightarrow \phi, \phi \Rightarrow \phi, \) and \(\phi \Rightarrow \) True.

\[\square\]

**Definition** The statement not \(\phi\) is defined by any of the following equivalent ways

\[
\begin{align*}
\phi & \vdash \text{False} & \phi & \Rightarrow \psi & \phi & \Rightarrow \text{not} \psi \\
\text{not} \phi & & & & \text{not} \phi \\
\end{align*}
\]

True is then not False, i.e., False \(\Rightarrow\) False.

Proof: Assuming (i)

\[
\begin{align*}
\phi & \Rightarrow \psi, \phi \Rightarrow \text{not} \psi \\
\therefore & \phi \vdash \psi \\
\phi & \vdash (\psi \Rightarrow \text{False}) \\
\therefore & \phi \vdash \text{False} \\
\therefore & \text{not} \phi \\
\end{align*}
\]

Assuming (ii)

\[
\begin{align*}
\phi & \Rightarrow \text{False} \\
\therefore & \phi \Rightarrow \text{False} \Rightarrow \psi, \text{not} \psi \\
\therefore & \text{not} \phi \\
\end{align*}
\]

\[\square\]

Deductions:
\[ \begin{array}{c|c}
\text{NOT (} \phi \text{ AND NOT } \phi \) & \phi \\
\hline
\text{NOT NOT } \phi & \end{array} \]

Proof: From \( \phi \) and \( \phi \Rightarrow \text{FALSE} \) we get \( \text{FALSE} \).
In other words, \( \phi, \text{NOT } \phi \vdash \text{FALSE} \), so \( \phi \vdash (\text{NOT } \phi \Rightarrow \text{FALSE}) \).

Inferences by \textit{contradiction}, and \textit{contrapositives}:

\[ \begin{array}{c|c|c|c}
\phi \Rightarrow \text{NOT } \phi & \phi \Rightarrow \psi & \phi \Rightarrow \psi & \phi \Rightarrow \text{NOT } \psi \\
\text{NOT } \phi & \text{NOT } \psi & \text{NOT } \psi \Rightarrow \text{NOT } \phi & \psi \Rightarrow \text{NOT } \phi \\
\hline
\text{:. NOT } \phi & \text{:. NOT } \psi & \text{:. } \phi \Rightarrow \text{FALSE} \Rightarrow \text{NOT } \phi & \text{:. } \phi \Rightarrow \psi \Rightarrow \text{FALSE} \Rightarrow \text{NOT } \phi \\
\end{array} \]

Proofs: (i) and (ii) (part (iii) means the same as (ii)).

\[ \begin{array}{c}
\phi \Rightarrow \text{NOT } \phi \\
\hline
\vdash (\phi \Rightarrow \text{FALSE}) \\
\vdash \phi, \text{NOT } \phi \vdash \text{FALSE} \\
\vdash \text{NOT } \phi \\
\end{array} \]

(iv) \( \psi \Rightarrow \text{NOT NOT } \psi \Rightarrow \text{NOT } \phi \) by (iii).

If one writes \( \phi' := \text{NOT } \phi \) for short, then \( \phi \Rightarrow \phi'' \), \( \phi' \Leftrightarrow \phi''' \), \( \phi'' \Leftrightarrow \phi'''' \).
One cannot deduce \( \phi'' \Rightarrow \phi \), but this law would hold if one were to replace or identify every statement \( \phi \) by \( \phi'' \) in our database of statements; this is a convenient simplification that, however, loses some “constructive” power.

### 1.6 Classical Logic

In view of the above, we make a new assumption that “if it’s not false, then it’s true”, or “it’s either true or false”, called the principle of Excluded Middle,

\[ \begin{array}{c|c|c}
\text{NOT NOT } \phi & \phi \text{ OR NOT } \phi & \phi \text{ OR } \psi \\
\hline
\phi & \phi \text{ OR NOT } \phi & \text{NOT } \phi \Rightarrow \psi \\
\end{array} \]

Proofs: assuming (i):

\[ \begin{array}{c}
\phi \Rightarrow \phi \text{ OR NOT } \phi \\
\text{NOT } \phi \Rightarrow \phi \text{ OR NOT } \phi \\
\hline
\vdash (\phi \text{ OR NOT } \phi) \Rightarrow \text{FALSE} \vdash \text{ NOT NOT } \phi \text{ AND NOT NOT } \phi \vdash \text{FALSE} \\
\vdash \text{ NOT NOT } (\phi \text{ OR NOT } \phi) \\
\vdash \phi \text{ OR NOT } \phi \\
\end{array} \]
Assuming (ii)
\[
\begin{align*}
\phi \lor \psi \\
\neg\phi \vdash \phi \Rightarrow \text{False} \\
\neg\phi \vdash \text{False} \lor \psi \\
\therefore \neg\phi \vdash \psi \\
\phi \Rightarrow \psi \\
\therefore \neg\phi \lor \psi 
\end{align*}
\]
Assuming (iii)
\[
\begin{align*}
\neg\neg\phi \\
\therefore \neg\phi \Rightarrow \text{False} \\
\neg\phi \Rightarrow \text{False} \\
\therefore \phi \\
\phi \vdash \text{not not } \phi \\
\text{already proved}
\end{align*}
\]

\[
\phi \Rightarrow \psi \\
\neg\neg\phi \lor \neg\neg\psi \\
\therefore \phi \lor \psi
\]

Exercise: NOT \(\phi \Rightarrow \psi\) ; (\(\phi \Rightarrow \psi\)) \iff (NOT \(\psi \Rightarrow \text{NOT } \phi\)).

De Morgan’s rules:
\[
\begin{align*}
\neg (\phi \land \psi) & \iff \neg \phi \lor \neg \psi \\
\neg (\phi \lor \psi) & \iff \neg \neg \phi \land \neg \neg \psi
\end{align*}
\]

Proofs:
\[
\begin{align*}
\phi \Rightarrow \phi \lor \psi \Rightarrow \text{False} \\
\therefore \neg\phi \\
\therefore \neg\psi \text{(similarly)} \\
\therefore \neg\phi \land \neg\psi
\end{align*}
\]
\[
\begin{align*}
\phi \land \psi \Rightarrow \phi \\
\neg\phi \Rightarrow \neg(\phi \land \psi) \\
\neg\psi \Rightarrow \neg(\phi \land \psi) \\
\therefore \neg(\phi \land \psi)
\end{align*}
\]
\[
\begin{align*}
\neg(\neg\phi \lor \neg\psi) \\
\therefore \neg\neg\phi \land \neg\neg\psi \\
\therefore \phi \land \psi
\end{align*}
\]
\[
\begin{align*}
\neg(\neg\phi \land \neg\psi) \\
\therefore \neg\phi \lor \neg\psi \\
\therefore \phi \lor \psi
\end{align*}
\]

\[
(\phi \Rightarrow \psi) \iff \neg\phi \lor \psi \iff \neg(\phi \land \neg\psi)
\]

Proofs: The first statement is the same as the definition of \(\neg\), replacing \(\phi\) with \(\neg\phi\), and the second statement is an application of the previous proposition.

\[
\therefore
\]
1. \((\phi \Rightarrow \psi) \text{ OR } (\psi \Rightarrow \phi)\)

2. \(\phi \text{ OR } \psi \equiv ((\phi \Rightarrow \psi) \Rightarrow \psi)\)

3. \((\phi \Leftrightarrow \psi) \Leftrightarrow (\phi \text{ AND } \psi) \text{ OR } (\text{NOT } \phi \text{ AND NOT } \psi)\)

There are some other logical constructs that can be defined in terms of \(\Rightarrow\), \(\text{NOT}\), \(\text{AND}\) and \(\text{OR}\):

\[
\begin{array}{|c|c|c|}
\hline
\text{IF } \phi \text{ THEN } \psi \text{ ELSE } \eta & (\phi \Rightarrow \psi) \text{ AND } (\text{NOT } \phi \Rightarrow \eta) \\
\phi \text{ XOR } \psi & (\phi \text{ AND NOT } \psi) \text{ OR } (\text{NOT } \phi \text{ AND } \psi) \\
\phi \text{ NOR } \psi & \text{NOT } (\phi \text{ OR } \psi) \\
\phi \text{ NAND } \psi & \text{NOT } (\phi \text{ AND } \psi) \\
\phi \text{ BUTNOT } \psi & \phi \text{ AND NOT } \psi \\
\hline
\end{array}
\]

\(\text{NOR}\) and \(\text{NAND}\) can by themselves generate all the other logical operators, e.g.

\[
\begin{align*}
\text{NOT } \phi & \iff \phi \text{ NAND } \phi, \\
(\phi \Rightarrow \psi) & \iff \phi \text{ NAND } (\phi \text{ NAND } \psi) \\
\phi \text{ OR } \psi & \iff (\phi \text{ NAND } \phi) \text{ NAND } (\psi \text{ NAND } \psi) \\
\phi \text{ AND } \psi & \iff (\phi \text{ NAND } \psi) \text{ NAND } (\phi \text{ NAND } \psi)
\end{align*}
\]

Decision Problem: Given a statement \(\phi\) in propositional logic that uses propositional variables (e.g. \((P \text{ AND } Q)\) \text{ OR } \text{NOT } P), and an assignment of truth or falsity to each variable, there is a polynomial algorithm that calculates the truth/falsity of \(\phi\). But the inverse problem of finding which assignments of the variables make it true, is not: any given assignment can be verified in polynomial time but to determine all such assignments may not be; the problem is said to be “NP”. Moreover, every other NP problem can be reduced to this one, i.e., it is NP-complete.

1.6.1 Constructive Logic

The inference \(\text{NOT} \text{ NOT } \phi \vdash \phi\) identifies the two statements; this is unacceptable to intuitionists, who instead accept only propositions that have a constructive proof. For “constructive logic”, the following weaker theorems hold

\[
\begin{align*}
\text{NOT NOT } (\phi \text{ OR NOT } \phi) \\
\text{NOT } (\phi \text{ OR } \psi) & \Rightarrow \text{NOT } \phi \text{ AND NOT } \psi; \\
\text{NOT } \phi \text{ OR NOT } \psi & \Rightarrow \text{NOT } (\phi \text{ AND } \psi)
\end{align*}
\]

Theorem (Glivenko) *If \(\phi\) can be proved using classical logic, then \(\text{NOT NOT } \phi\) can be proved using constructive logic.*

Proof: If at some stage in the proof we used \(P\), even if \(P\) is \(\psi\) or \(\text{NOT } \psi\), then we can prove \(\text{NOT NOT } P\) (since \(\text{NOT NOT } (\psi \text{ OR NOT } \psi)\) is provable in constructive logic); also (Exercise: use constructive logic!)

\[
\text{NOT NOT } (P \Rightarrow Q) \Rightarrow (\text{NOT NOT } P \Rightarrow \text{NOT NOT } Q)
\]

\[
\text{NOT NOT } (P \Rightarrow Q) \Rightarrow (\text{NOT NOT } P \Rightarrow \text{NOT NOT } Q)
\]
so that we can deduce \( \text{not not Q} \); similarly,

\[
\text{not not (P and Q)} \iff (\text{not not P}) \text{ and (not not Q)},
\]

\[
(\text{not not P}) \text{ or (not not Q)} \implies \text{not not (P or Q)}.
\]

\[\square\]

Note however that if \( \phi \) contains predicates, we need the extra assumption that \( \forall x \text{ not not } \phi_x \iff \text{not not } (\forall x, \phi_x) \) for this to work.

Corollary: If \( \phi \ldots \psi \ldots \Rightarrow \xi \) using classical logic, then one can prove \( \phi'' \ldots \psi' \ldots \Rightarrow \xi'' \) using constructive logic (since \( \psi'' \Leftrightarrow \psi' \) even in constructive logic.)

1.6.2 Modal Logic

Definition: \( \square \phi \) is another construct that means “\( \phi \) is necessarily true” (in all possible worlds at all possible times). Then

\[\diamond \phi := \text{not } \square \text{not } \phi\]

meaning “\( \phi \) may possibly be true” (in some world sometime). It follows that

\[
\text{not } \square \phi \iff \diamond \text{not } \phi, \quad \text{not } \diamond \phi \iff \square \text{not } \phi.
\]

The statement \( \diamond \phi \text{ and } \diamond \text{not } \phi \) is not a contradiction: \( \phi \) may or may not hold, depending on the case, i.e., \( \phi \) is contingent. Similarly, \( \diamond \phi \text{ and } \neg \phi \) may be false in the model, but it may conceivably hold in another. Some inference rules that are usually assumed are

\[
L \vdash \phi \quad \square(\phi \Rightarrow \psi) \quad \diamond \phi \Rightarrow \boxdot \phi \quad \diamond \phi \Rightarrow \square \diamond \phi \quad \square \phi \Rightarrow \diamond \phi
\]

1.6.3 Three-Valued Logic

Statements are allowed to have three “values”, namely True, False or undef (undefined or missing); this is particularly useful in computing where variables or program outputs may give unassigned values. The definitions of not, and, or, and \( \Rightarrow \) are extended by

\[
\text{not undef} := \text{undef} \quad \text{undef and } \phi := \text{undef} \quad \text{exceptundef and False} := \text{False},
\]

\[
\text{undef or } \phi := \text{undef} \quad \text{exceptundef or True} := \text{True}.
\]
1.6.4 Linear Logic

A new development is to consider proofs as programs inputting statements, with the consequence that the logical axioms $\phi, \phi \vdash \psi$ then $\phi \vdash \psi$, and also that $\phi \vdash \eta$ then $\phi, \psi \vdash \eta$ do not hold any more, unless they are fortified i.e., of a new type $!\phi$ e.g. if $\phi := \text{“button is pressed”}$, then one may need $\phi$ twice to get a desired result $\psi$. This has the consequence that we must distinguish between two types of AND statements, one where $\eta \vdash \phi$ and $\eta \vdash \psi$ are written as $\eta \vdash \phi \land \psi$; and a second where $\eta \vdash \phi$ and $\chi \vdash \psi$ are written as $\eta, \chi \vdash \phi \land \psi$. Hence there are two types of OR statements, namely $(\phi \land \psi)'$ and $(\phi' \& \psi')'$; two types of truth statements, namely the truth $\forall \phi(\phi \Rightarrow \phi)$ and the truth $\exists \phi, \phi$ which are not equivalent in this logic. (The result is a commutative bounded residuated lattice monoid.)

2 Properties

The basic “atomic” statements are taken to be properties about objects:

A property (or predicate) $A$ accepts an object $x$ to become a statement $A_x$, e.g. $x + 1 = 0$, “roses are red” ($\text{Red}_{\text{roses}}$). We also say “$x$ is $A$”, and write it as $x \in A$.

Properties of objects $x$ and $y$ are called relations, $A_{x,y}$, and are often written as $xAy$ (more generally, $A_{x,...,z}$).

2.1 Predicate Logic

Definition:

\[
\frac{L \vdash A_x}{L \vdash \forall x A_x} \quad \frac{A_x \vdash M}{\exists x A_x \vdash M}
\]

These are generalized forms of the AND and OR definitions respectively. The variable $x$ in the top line of $\forall$ is arbitrary, while that for $\exists$ is specific (i.e., the inference is valid when any instance of $A_x \vdash M$ is true).

From these, other inferences can be deduced:

\[
\frac{\forall x A_x}{A_a} \quad \frac{A_a}{\exists x A_x}
\]

\[
\frac{A_a \Rightarrow \psi}{\forall x A_x \Rightarrow \psi} \quad \frac{\psi \Rightarrow A_a}{\exists x A_x \Rightarrow \psi} \\
\frac{\forall x (\phi \Rightarrow A_x)}{\phi \Rightarrow \forall x A_x} \quad \frac{\forall x (A_x \Rightarrow \psi)}{\exists x A_x \Rightarrow \psi}
\]

\[
\frac{\forall x (\phi \Rightarrow A_x)}{\phi \Rightarrow \forall x A_x} \quad \frac{\forall x (A_x \Rightarrow \psi)}{\exists x A_x \Rightarrow \psi}
\]

\[
\frac{\forall x (\phi \Rightarrow A_x)}{\phi \Rightarrow \forall x A_x} \quad \frac{\forall x (A_x \Rightarrow \psi)}{\exists x A_x \Rightarrow \psi}
\]

\[
\frac{\forall x (\phi \Rightarrow A_x)}{\phi \Rightarrow \forall x A_x} \quad \frac{\forall x (A_x \Rightarrow \psi)}{\exists x A_x \Rightarrow \psi}
\]
Proof: (ii) If $\forall x, \phi \Rightarrow A_x$, then $\phi \Rightarrow A_x$ (for arbitrary $x$), hence $\phi \vdash \forall x A_x$.

\[
\begin{array}{c}
\forall x, A_x \Rightarrow B_x \\
\forall x, B_x \Rightarrow C_x \\
\therefore \forall x, A_x \Rightarrow C_x
\end{array}
\]

Notice that no inference can be made from two particular statements (containing $\exists$), or from two negative statements.

\[
\begin{aligned}
\text{NOT}(\forall x, A_x) & \iff \exists x, \text{NOT} A_x \\
\text{NOT}(\exists x, A_x) & \iff \forall x, \text{NOT} A_x
\end{aligned}
\]

For short, we normally write $\forall x \in A, B_x$ instead of $\forall x, (x \in A \Rightarrow B_x)$, and $\exists x \in A, B_x$ instead of $\exists x, (x \in A \text{ AND } B_x)$. Then

\[
\begin{aligned}
\text{NOT}(\forall x \in A, B_x) & \iff \exists x \in A, \text{NOT} B_x, \\
\text{NOT}(\exists x \in A, B_x) & \iff \forall x \in A, \text{NOT} B_x.
\end{aligned}
\]

\[
\begin{array}{c}
\forall x (\phi \text{ OR } A_x) \\
\phi \text{ OR } \forall x A_x
\end{array}
\quad \begin{array}{c}
\exists x (\phi \text{ AND } A_x) \\
\phi \text{ AND } \exists x A_x
\end{array}
\quad \begin{array}{c}
\forall x (A_x \text{ AND } B_x) \\
(\forall x, A_x) \text{ AND } (\forall x, B_x)
\end{array}
\quad \begin{array}{c}
\exists x(A_x \text{ OR } B_x) \\
(\exists x, A_x) \text{ OR } (\exists x, B_x)
\end{array}
\quad \begin{array}{c}
\forall x(A_x \text{ OR } B_x) \\
(\forall x, A_x) \text{ OR } (\forall x, B_x)
\end{array}
\quad \begin{array}{c}
\exists x(A_x \text{ AND } B_x) \\
(\exists x, A_x) \text{ AND } (\exists x, B_x)
\end{array}
\]

\[
\begin{array}{c}
\forall x \forall y, A_{x,y} \iff \forall y \forall x, A_{x,y} \\
\exists x \exists y, A_{x,y} \iff \exists y \exists x, A_{x,y} \\
\exists x \forall y, A_{x,y} \iff \forall y \exists x, A_{x,y}
\end{array}
\]

Proof: for (i) straightforward; for (ii) we know that $A_{x,y} \Rightarrow \exists y \exists x, A_{x,y}$, hence $\exists y, A_{x,y} \Rightarrow \exists y \exists x, A_{x,y}$ and finally $\exists x \exists y, A_{x,y} \Rightarrow \exists y \exists x, A_{x,y}$.
(iii) we know \( \forall y, (A_{x,y} \implies \exists x, A_{x,y}) \), hence \( \forall y, A_{x,y} \implies \forall y \exists x, A_{x,y} \) and hence the result follows.

\[ \square \]

Note carefully:

\[ \forall x (A_x \text{ or } B_x) \not\iff (\forall x A_x) \text{ or } (\forall x B_x) \]
\[ (\exists x, A_x) \text{ AND } (\exists x, B_x) \not\iff \exists x (A_x \text{ AND } B_x) \]

Suffice to say that every person is male or female but there are both; there is a male and there is a female but not both at the same time (at least up to the nineteenth century).

Important: with these rules, a statement \( A_x \) in which \( x \) is an unspecified variable could be of two types: either \( x \) is free or arbitrary, in which case the statement is called an identity, or \( x \) is not free, when \( A_x \) is called an equation. To avoid confusing the two, one ought to distinguish them by bounding the variables with \( \forall \) or \( \exists \).

Theorem: In a closed predicate statement, the bound variable names are not important as long as they remain distinct.

There is some classical (Aristotelian) terminology that has been replaced by this symbolic form:

- Every A is B  \( \forall x \in A, x \in B \)
- No A is B  \( \forall x \in A, x \notin B \)
- Some A is B  \( \exists x \in A, x \in B \)
- Some A is not B  \( \exists x \in A, x \notin B \).

(Note that a particular statement \( A_a \) should be treated as a universal, i.e., Every \( a \) is \( A \). The problem with this classical terminology is that it is not general enough to allow statements of the type \( \forall x \exists y, A_{x,y} \).)

**Theorem 1**

*(Replacement)*

If \( \psi \) is a propositional part of \( \phi \), and \( \psi \iff \tilde{\psi} \) then, letting \( \tilde{\phi} \) be the statement obtained by replacing \( \psi \) with \( \tilde{\psi} \), we have \( \phi \iff \tilde{\phi} \).

“Proof”: The part \( \psi \) could be part of one of the following: \( \eta \implies \xi, \xi \implies \eta, \eta \text{ and } \xi, \eta \text{ or } \xi, \text{ not } \eta, \forall x A_x, \text{ or } \exists x A_x \). In all cases, if we have \( \eta \iff \tilde{\eta} \) we also have \( \tilde{\eta} \implies \xi, \xi \implies \tilde{\eta}, \tilde{\eta} \text{ and } \xi, \tilde{\eta} \text{ or } \xi, \text{ not } \tilde{\eta}, \forall x, \tilde{A}_x, \text{ or } \exists x, \tilde{A}_x \). Moreover each of these statements are equivalent to the original ones; hence can continue making these replacements up the levels of the propositions until \( \phi \iff \tilde{\phi} \).

\[ \square \]
1. Prove the following and show that the converses are false,

(a) \( \forall x, A_x \Rightarrow \exists x, A_x \) (as long as there is at least an object \( a \)); similarly, \( \forall x \in B, A_x \Rightarrow \exists x \in B, A_x \) unless \( B_x \) never holds.

(b) if \( \forall x, (A_x \Rightarrow B_x) \) then \( \forall x, A_x \Rightarrow \forall x, B_x \) and \( \exists x, A_x \Rightarrow \exists x, B_x \)

(c) \( \forall y, (A_{x,y} \Rightarrow \exists y, A_{x,y}) \)

(d) if \( \exists x, A_x \) AND \( B_x \) then \( \exists x, A_x \) AND \( \exists x, B_x \)

(e) if \( (\exists x, A_x) \Rightarrow \phi \) then \( \exists x, (A_x \Rightarrow \phi) \).

2. Prove that in constructive logic,

(a) NOT \( (\exists x, A_x) \) \( \Leftrightarrow \) \( \forall x, \text{NOT} A_x \)

(b) NOT \( (\exists x, \text{NOT} A_x) \) \( \Leftrightarrow \) \( \forall x, \text{NOT NOT} A_x \)

(c) NOT \( (\forall x, A_x) \) \( \Leftrightarrow \) NOT NOT \( (\exists x, \text{NOT} A_x) \)

(d) NOT \( (\forall x, \text{NOT} A_x) \) \( \Leftrightarrow \) NOT NOT \( (\exists x, A_x) \).

2.1.1 Equality

The basic predicate is taken to be the equality relation: Objects \( x \) and \( y \) are said to be equal when they have the same properties

\[ x = y := \forall A, A_x \Leftrightarrow A_y \]

while properties are equal when they are satisfied by the same objects

\[ A = B := \forall x, A_x \Leftrightarrow B_x. \]

This implies that there are no indistinguishable objects, in the sense that if objects share exactly the same properties then they are the same. Similarly, there are not indistinguishable properties.

\[
\begin{align*}
x &= x \\
x &= y & \Leftrightarrow & \ y &= x \\
x &= y \text{ AND } y = z & \Rightarrow & \ x &= z \\
\end{align*}
\]

Definition: \( \exists! x A_x := \exists x \in A, \forall y \in A, \ y = x \)

2.1.2 Second-Order Logic

Second-order quantifiers are extended to cover statements themselves, as in \( \forall \phi, \phi \Leftrightarrow \phi \), but the logic that results is not as well-behaved (it must either be inconsistent, incomplete, or ineffective). In this logic, one can define TRUE to be the statement above, or \( \exists \phi, \phi \), or \( \forall \phi, \phi \) OR \( \text{NOT} \phi \), and FALSE its negation.
3 Meta-Theorems

A language defines symbols and rules for manipulating their sentences.

An interpretation is an actual example that implements the language and defines the objects $x$.

A theory is a logic (usually predicate logic) with additional axioms, e.g., Peano axioms or Set theory; an actual example is called a model. The axioms are usually required to be effective, i.e., proofs can be checked in a finite number of steps. In particular there must be a finite number of axioms or axiom schemas (i.e., rules for axioms).

There are two senses that a statement can be true: it may be true in the sense that (i) it can be proved by deductions (syntactically true) $L \vdash \phi$, or (ii) it holds universally, by checking that it holds in every possible model (semantically true), $L \models \phi$.

Definition: A theory is said to be

1. **sound** when every proven proposition is actually true in every model (that satisfies the axioms), i.e., if $L \vdash \phi$ then $L \models \phi$. One only has to show that the inference rules of the theory preserve semantic truth.

2. **decidable** (or semantically complete) when every statement that holds in every model can be proved, i.e., if $L \models \phi$ then $L \vdash \phi$. One has to show that for every statement that cannot be proved, there is a model in which it does not hold.

3. **satisfiable** (or semantically consistent) when it has at least one model in which the axioms hold.

4. **complete** when for any $\phi$ either $\phi$ or $\text{NOT} \ \phi$ can be proved. A complete sound logic must be decidable.

5. **consistent** when the statement $\text{FALSE}$ cannot be proved (from the axioms). A satisfiable sound theory must be consistent.

**Theorem 2**

Classical Propositional Logic is complete, i.e., every logical statement is either provable or disprovable from the axioms.

‘Proof’: The “truth-tables” of the connectives are forced upon us by the axioms of classical logic; every proposition can be written as a string of OR statements, with AND inside each term; so checking each term until one is true, or none, decides it.
Theorem 3

(Henkin)

A theory is consistent if, and only if, it has a model.

(Löwenheim-Skolem theorem: In fact, if infinite, it must have a model of every cardinality above its minimum model).

Theorem 4

(Gödel)

Classical predicate logic is sound and decidable, i.e., a statement is true in all interpretations if, and only if, it can be proved.

Proof: The axioms must be true in all interpretations because their truth-values are forced by the axioms. The inferences allow only true statements to follow.

For the converse, suppose that \( \phi \) is a “true” statement (in all interpretations) that cannot be proved from the axioms. Then it cannot be the case that there are proofs \( \text{NOT } \phi \Rightarrow \psi \) and \( \text{NOT } \phi \Rightarrow \text{NOT } \psi \), else we would deduce \( \text{NOT NOT } \phi \) and so \( \phi \). So we can safely add the axiom \( \text{NOT } \phi \) to classical logic, without getting contradictions. By Henkin’s theorem, there is a model satisfying classical logic and this new axiom, but the statement \( \phi \) was supposed to be true in all interpretations.

\[ \square \]

More generally, classical logic with the addition of consistent axioms is sound and semantically complete.

Corollary: the true statements are countable (because proofs can be listed).

Theorem 5

(Tarski)

There is no truth function in Set Theory.

Proof: Extend classical logic by the natural numbers; then the well-formed statements of logic (“formulas”) can be assigned a number. Suppose \( \text{True} : \text{Formulas} \rightarrow \{ T, F \} \) is a function which maps true statements to \( \text{TRUE} \) and false to \( \text{FALSE} \). Then

\[ \text{True}(\phi) = T \iff \phi. \]

Tarski’s fixed point theorem (see Order) shows that for any monotonic formula \( f \) there is a statement \( \psi \) which solves \( x \iff f(x) \). In particular, \( (\phi \Rightarrow \psi) \Rightarrow \)
True(ϕ) ≤ True(ψ) ⇒ ¬True(ϕ) ≥ ¬True(ψ), so there exists a fixed point ψ which solves ψ ↔ ¬True(ψ). Hence ψ ↔ ¬True(ψ) = T ⇔ NOT ψ.

Hence the true statements are not “definable” by some formula. By contrast, the provable statements can be defined rigorously (Gödel), so the two are not the same. So either there is a provable statement that is not true (unsound theory) or a true statement that is not provable (incomplete theory).

**Theorem 6**

\[ \text{(Gödel)} \]

**Set Theory is inconsistent or incomplete.**

“Proof”: Extend the logical axioms to include the natural numbers (see Sets). Let \( A_n(m) \) be a list of all predicates about the natural numbers (these are countable since their alphabet is finite). Let \( A_N(m) \) be the predicate “\( A_m(m) \) is not provable” (this requires some clarification: to say that \( p \) is a proof of \( A(m) \) can be shown to be a rigorously definable statement; hence \( A(m) \) is not provable means \( ∀p, p \) is not a proof of \( A(m) \)). Proofs are countable, so one can make a mapping from the list of proofs to the list of predicates). So

\[ A_N(N) \text{ is not provable } ⇔ A_N(N) \]

If \( A_N(N) \) is false, then there is a proof of \( A_N(N) \) starting from the axioms of set theory; this makes the theory inconsistent. If \( A_N(N) \) is true, then \( A_N(N) \) is unprovable. But then, neither can there be a proof of \( NOT A_N(N) \), else \( A_N(N) \) be false. Thus \( A_N(N) \) cannot be proved or disproved.

(Alternatively, the statement “\( x \Leftrightarrow x \) is not provable” has a solution \( y \) by Tarski’s theorem; either \( y \) is true in which case \( y \) is true but not provable (incomplete), or \( y \) is false and provable (inconsistent)).

□

Note: More generally, this is true for any sufficiently powerful axioms, such as any extension of the Peano axioms; the axioms of the naturals without multiplication (i.e., repeated addition) are consistent and complete. This theorem is not so surprising after all: it is conceivable that a statement about natural numbers is true but requires a different proof for each number; thus no single proof is sufficient to show \( ∀n, A_n \), yet it is true. It is possible to have a consistent and complete theory, by taking all true statements as axioms, but then the theory becomes ineffective: it would be impossible to check whether a statement is an axiom or not.

Turing’s theorem that “there is no algorithm which decides whether any algorithm terminates” is along the same lines. Suppose, as a special case, \( T \) is
an algorithm that inputs a program $P$ and a string $A$ and outputs true if $P(A)$ halts otherwise it outputs false. Then we can create the following algorithm, call it $C$, which inputs a program $P$:

$$\text{if } T(P, P) = \text{true} \text{ then while}(0 \neq 1) \{ \} \text{ else return.}$$

This program halts if, and only if, $T(P, P)$ does not. Then $C(C)$ first runs $T(C, C)$, which in turn checks out $C(C)$. If $T$ decides that $C(C)$ halts then $C(C)$ enters an endless loop, otherwise if it decides that it does not halt, then $C(C)$ halts.

All right, there are theorems that can be proved and theorems that are true but unprovable. Perhaps there is a way of distinguishing between them so as not to waste time trying to prove an unprovable conjecture. But

**Theorem 7**

(Church)

There cannot be a finite algorithm that decides whether an arbitrary $\phi$ in Set Theory is provable.

An algorithm here means a program that uses the basic commands

$$x \leftarrow 1, \; x \rightarrow 1, \; \text{if}(x=0)\{\text{goto } n\}.$$

‘Proof’: There is a way of converting an algorithm $A$ into a statement $S(A)$ such that $A$ halts $\Leftrightarrow S(A)$ is provable. Suppose there is an algorithm $T$ that decides decidability. So $A$ halts iff $S(A)$ is provable iff $T(S(A))$ outputs TRUE, i.e., $T \circ S$ solves the halting problem, which is impossible.

Since proofs can be listed, there is an algorithm that finds a proof for any theorem. If such an algorithm is applied to a non-provable true statement, it simply does not stop. One cannot deduce anything about a statement until (if) the algorithm stops: while it runs, either the statement has a long proof or it is non-provable. But for any specific statement, there may be an algorithm that decides whether it is true or not (one strategy is to write down the statement in terms of OR, NOT and $\exists$ only - if one of the atomic expressions is true, then the statement is true, if all false then false; but this may take exponential time and there is no general fail-safe way of doing it); moreover there are particular theories (e.g. Euclidean geometry) for which every theorem is decidable.

Gödel’s second theorem: **If Set theory is consistent, then it cannot prove this fact from its axioms.**

In particular, there can be no algorithm that decides whether any set of axioms is consistent or not.

However Set theory can prove the consistency of the Peano axioms; moreover there are statements about the natural numbers that can be proved by Set theory but not by the Peano axioms. The most that Set theory can do is prove that the addition of an axiom does not add any new inconsistency.