1 Preliminaries

We begin with some basic definitions and facts.

A string is a finite sequence of symbols, e.g. abba. The concatenation of two strings $s_1$ and $s_2$ is just $s_1s_2$. Length, suffix, prefix of a string.

Operations on sets: union, intersection, Cartesian product, etc. Two sets are said of the same cardinality if there is a bijection from one onto the other. A set is countable if it has the same cardinality as $\mathbb{N}$. For example, the set of all rational numbers is countable and the set $\{0,1\}^*$ of all finite length binary strings is countable (Why?). The set of all real numbers is not countable (Why?).

In this course we will use various inductive proofs.

Consider a set $U$. For simplicity, let us consider just two operations $f : U \times U \rightarrow U$ and $g : U \rightarrow U$, in other words, $f$ is a binary operation on $U$ and $g$ is a unary one. Recall that a set $X$ is closed under $f$ if for any $x, x' \in X$ then $f(x, x') \in X$. Let $B \subseteq U$. Define $\overline{C}$ — the closure of $B$ under $f$ and $g$ — to be the smallest set containing $B$ which is also closed under $f$ and $g$, i.e.,

$$\overline{C} = \bigcap\{X \subseteq U | B \subseteq X \text{ and } X \text{ is closed under } f \text{ and } g\}.$$

Next we show that we can approximate the closure of $B$ “from below”. Define a sequence of sets $\{C_n | n \in \mathbb{N}\}$ by:

$C_0 = B$;

$$C_{n+1} = C_n \cup f[C_n \times C_n] \cup g[C_n].$$

Let $\underline{C} = \bigcup\{C_n | n \in \mathbb{N}\}$. Notice that $C_n \subseteq C_{n+1}$. We call $\underline{C}$ the set generated from $B$ by $f$ and $g$.

**Theorem 1** $\overline{C} = \underline{C}$.

**Proof.** (“$\subseteq$”) It suffices to show that $\underline{C}$ is closed under $f$ and $g$. Assume that $a, b \in \underline{C}$. Then $a \in C_m$ and $b \in C_n$ for some $m, n \in \mathbb{N}$. Let us assume $m \leq n$. Then both $f(a,b)$ and $g(a)$ are in $C_{n+1}$, hence in $\underline{C}$. Thus $\underline{C}$ is closed under $f$ and $g$. 

1
(“⊇”) We show that $C_n \subseteq \bar{C}$ by induction on $n$. Clearly, $C_0 = B \subseteq \bar{C}$. Assume $C_n \subseteq \bar{C}$. Then $f[C_n \times C_n]$ and $g[C_n]$ are contained in $\bar{C}$, since $\bar{C}$ is closed under $f$ and $g$. Thus $C_{n+1} \subseteq \bar{C}$. By Induction Principle, $C_n \subseteq \bar{C}$ for all $n \in \mathbb{N}$. Therefore, $\mathcal{C} \subseteq \bar{C}$. 

**Theorem 2** Let $P(x)$ be a property. Assume that

1. $P(a)$ holds for all $a \in B$.
2. For each $a, b \in A$, $P(a)$ and $P(b)$ hold implies both $P(f(a, b))$ and $P(g(a))$ hold.

Then $P(x)$ holds for all $x \in \bar{C}$.

We also use definition by recursion.

We say that $C$ is *freely* generated from $B$ by $f$ and $g$ if $C$ is generated from $B$ and

1. $f$ and $g$ are one-to-one, and
2. The range of $f$, the range of $g$, and the set $B$ are pairwise disjoint.

**Theorem 3** Assume that $C$ is freely generated from $B$ by $f$ and $g$. Further assume that $V$ is a set and $F, G$ and $h$ functions such that $h : B \to V$, $F : V \times V \to V$, and $G : V \to V$. Then there is a unique function $\bar{h} : C \to V$ such that

(1) For $x$ in $B$, $\bar{h}(x) = h(x)$.

(2) For $x, y$ in $C$, $\bar{h}(f(x, y)) = F(\bar{h}(x), \bar{h}(y))$ and $\bar{h}(g(x)) = G(\bar{h}(x))$.

**Proof.** (Sketch) The idea is to let $\bar{h}$ be the union of many approximating functions. A function $v$ is called **acceptable** if it meets the conditions imposed on $\bar{h}$ by (1) and (2). Let $K$ be the collection of all acceptable functions, and let $\bar{h}$ be the union of $K$. Thus

$$\langle x, y \rangle \in \bar{h} \text{ iff } v(x) = y \text{ for some } v \in K.$$
We can show that \( \bar{h} \) satisfies all conditions by proving the following claims (more details can be found on [Enderton]).

**Claim 1.** \( \bar{h} \) is a function.

**Claim 2.** \( \bar{h} \) is defined throughout \( C \).

**Claim 3.** \( \bar{h} \) is an acceptable function, i.e., it satisfies (1) and (2).

**Claim 4.** \( \bar{h} \) is unique.  

\[ \square \]

# 2 Sentential Logic

## 2.1 The Language of Sentential Logic

The *language of sentential logic* consists of the following three parts:

1. A countable set of *sentential symbols* (or *proposition symbols*): \( A_1, A_2, \ldots \)

2. Five *sentential connectives*: negation \( \neg \) (read as “not”), conjunction \( \land \) (read as “and”), disjunction \( \lor \) (read as “or”), conditional \( \rightarrow \) (read as “implies” or “if... then...”) and biconditional \( \leftrightarrow \) (read as “if and only if”).

3. Parentheses: left parenthesis (, and right parenthesis ).

An *expression* is a string of symbols. For example, \((\neg A_1)\), \((A_1 \lor A_2)\) are expressions. We now give a formal definition of *well-formed formulas* (wffs). The informal idea is that wffs are grammatically correct expressions.

1. Every sentence symbol \( A_i \) is a wff.

2. If \( \alpha \) and \( \beta \) are wffs, then so are \((\neg \alpha), (\alpha \land \beta), (\alpha \lor \beta), (\alpha \rightarrow \beta) \) and \((\alpha \leftrightarrow \beta)\).

3. No expression is a wff unless it is compelled to be one by (a) or (b).
If we define an operation \( \mathcal{E}_\wedge \) on expressions by \( \mathcal{E}_\wedge (\alpha) = (\neg \alpha) \), and \( \mathcal{E}_\vee, \mathcal{E}_\rightarrow, \) and \( \mathcal{E}_\leftrightarrow \), similarly, then the set of wffs is the smallest set containing sentential symbols and closed under those 5 operations. Equivalently, we say \( \alpha \) is a wff iff \( \alpha \) can be built from sentential symbols by applying some of the 5 operations finitely many times.

### 2.2 Truth Assignments

Let us fix a set of truth values \( \{T, F\} \). A truth assignment \( v \) for a set \( S \) of sentence symbols is a function

\[
v : S \rightarrow \{T, F\}.
\]

Let \( \bar{S} \) be the set of wffs generated from \( S \) by the five formula-building operations. We have an extension \( \bar{v} \) of \( v \):

\[
\bar{v} : \bar{S} \rightarrow \{T, F\},
\]

such that,

1. For any \( A \in S \), \( \bar{v}(A) = v(A) \).
2. \[
\bar{v}(\neg \alpha) = \begin{cases} 
T & \text{if } \bar{v}(\alpha) = F, \\
F & \text{otherwise.}
\end{cases}
\]
3. \[
\bar{v}(\alpha \land \beta) = \begin{cases} 
T & \text{if } \bar{v}(\alpha) = T \text{ and } \bar{v}(\beta) = T, \\
F & \text{otherwise.}
\end{cases}
\]
4. \[
\bar{v}(\alpha \lor \beta) = \begin{cases} 
T & \text{if } \bar{v}(\alpha) = T \text{ or } \bar{v}(\beta) = T \text{ (or both),} \\
F & \text{otherwise.}
\end{cases}
\]

4
(4) \[ \bar{v}((\alpha \rightarrow \beta)) = \begin{cases} F & \text{if } \bar{v}(\alpha) = T \text{ and } \bar{v}(\beta) = F, \\ T & \text{otherwise.} \end{cases} \]

(5) \[ \bar{v}((\alpha \leftrightarrow \beta)) = \begin{cases} T & \text{if } \bar{v}(\alpha) = \bar{v}(\beta), \\ F & \text{otherwise.} \end{cases} \]

We can also represent \( \bar{v} \) by a truth table.

<table>
<thead>
<tr>
<th>( \alpha )</th>
<th>( \beta )</th>
<th>( \neg \alpha )</th>
<th>( \alpha \land \beta )</th>
<th>( \alpha \lor \beta )</th>
<th>( \alpha \rightarrow \beta )</th>
<th>( \alpha \leftrightarrow \beta )</th>
</tr>
</thead>
<tbody>
<tr>
<td>T</td>
<td>T</td>
<td>F</td>
<td>T</td>
<td>T</td>
<td>T</td>
<td>T</td>
</tr>
<tr>
<td>T</td>
<td>F</td>
<td>F</td>
<td>F</td>
<td>T</td>
<td>F</td>
<td>F</td>
</tr>
<tr>
<td>F</td>
<td>T</td>
<td>T</td>
<td>F</td>
<td>T</td>
<td>T</td>
<td>F</td>
</tr>
<tr>
<td>F</td>
<td>F</td>
<td>T</td>
<td>F</td>
<td>F</td>
<td>T</td>
<td>T</td>
</tr>
</tbody>
</table>

Example. Suppose that \( v(A) = v(B) = T \) and \( v(C) = F \). Let \( \alpha \) be the wff \(((B \rightarrow (A \rightarrow C)) \leftrightarrow ((B \land A) \rightarrow C))\).

What is \( \bar{v}(\alpha) \)?

**Theorem 4** For any truth assignment \( v \) for a set \( S \) there is a unique function \( \bar{v} : \hat{S} \rightarrow \{T, F\} \) meeting the aforementioned conditions (0) – (5).

Theorem 4 follows from the Recursion Theorem in Preliminary and the unique Readability Theorem, which we will see later.

We say that a truth assignment \( v \) satisfies \( \phi \) iff \( \bar{v}(\phi) = T \).

**Definition 1** \( \Sigma \) tautologically implies \( \tau \) (\( \Sigma \models \tau \)) iff every truth assignment satisfying every member of \( \Sigma \) also satisfies \( \tau \).
Examples. \(|\{\alpha \land \beta\}\} \models \alpha\). Is \(|\{A, \neg A\}\} \models B\)?

We say that \(\tau\) is a tautology \((\models \tau)\) iff \(\emptyset \models \tau\). In other words, \(\tau\) is satisfied by every truth assignment (Can you justify this?).

If \(\Sigma = \{\sigma\}\), then we write \(\sigma \models \tau\) instead of \(|\{\sigma\}\} \models \tau\). If both \(\sigma \models \tau\) and \(\tau \models \sigma\), then we say taht \(\sigma\) and \(\tau\) are tautologically equivalent, written \(\sigma \equiv \tau\).

A Selected List of Tautologies

1. Associative, commutative and distributive laws.

2. Negation:

\[ ((\neg (\neg A)) \leftrightarrow A). \]

De Morgan’s laws.

3. Other:

Excluded middle: \((A \lor (\neg A))\).

Contradiction: \((\neg (A \land (\neg A)))\).

Contraposition: \((A \rightarrow B) \leftrightarrow ((\neg B) \rightarrow (\neg A)))\).

Recursion Theorem (revisit)

2.3 Unique Readability

We now prove that there is no ambiguity in analyzing wffs. More precisely, the set of wffs is freely generated from sentential symbols by the five formula-building operators.

Lemma 1 Every wff has the same number of left as right parentheses. Moreover, any proper prefix of a wff contains an excess of left parentheses. Thus no proper prefix of wff is a wff.

Proof. (Sketch) It suffices to check that the set of wffs having the required property is an inductive set. \(\square\)
Theorem 5 (Unique Readability Theorem) The five formula-building operations, when restricted to the set of wffs,

(a) have ranges which are disjoint from each other and from the set of sentential symbols, and

(b) are one-to-one.

Proof. (Sketch) We only show that the restriction of $E_\land$ is one-to-one as an example, other parts are similar. Suppose that $\alpha \land \beta = (\gamma \land \delta)$. Then $\alpha \land \beta = \gamma \land \delta$. Then $\alpha = \gamma$, otherwise one is a proper prefix of the other. Hence $\beta = \delta$. $\square$

Omitting Parentheses

We now adopt the following conventions:

1. The outmost parentheses are often omitted.
2. The negation symbol applies to as little as possible.
3. When (2) is obeyed, the conjunction and disjunction symbols apply to as little as possible. For example, $A \land B \rightarrow \neg C \lor D$ is $(A \land B) \rightarrow ((\neg C) \lor D))$.
4. For repeating connectives, grouping is to the right. For example, $\alpha \rightarrow \beta \rightarrow \gamma$ is $\alpha \rightarrow (\beta \rightarrow \gamma)$.

2.4 Sentential Connectives

We say a function $B$ a $k$-place Boolean function, if $B$ is from $\{T,F\}^k$ into $\{T,F\}$. If $\alpha$ is a wff whose sentence symbols are at most $A_1,\ldots,A_n$, then $\alpha$ defines an $n$-place Boolean function $B^n_\alpha$ by

$$B^n_\alpha = \text{the truth value given to } \alpha \text{ when }$$

$$\ A_1,\ldots,A_n \text{ are given the values } X_1,\ldots,X_n.$$

We call $B^n_\alpha$ the Boolean function realized by $\alpha$. 

7
Theorem 6 Let $G$ be an $n$-place Boolean function, $n \geq 1$. We can find a wff $\alpha$ such that $G = B^n_\alpha$, i.e., such that $\alpha$ realizes the function $G$.

Proof. Case 1: ran $G = \{F\}$. Let $\alpha = A \land \neg A$.

Case 2: Otherwise there are $k$ points at which $G$ has the value $T$, $k > 0$. List these:

$\bar{X}_1 = \langle X_{11}, X_{12}, \ldots, X_{1n} \rangle$,

$\bar{X}_2 = \langle X_{21}, X_{22}, \ldots, X_{2n} \rangle$,

$\ldots$

$\bar{X}_k = \langle X_{k1}, X_{k2}, \ldots, X_{kn} \rangle$.

Let

$$\beta_{ij} = \begin{cases} A_j & \text{if } X_{ij} = T, \\ \neg A_j & \text{otherwise,} \end{cases}$$

$$\gamma_i = \beta_{i1} \land \cdots \land \beta_{in},$$

$$\alpha = \gamma_1 \lor \gamma_2 \lor \cdots \lor \gamma_k.$$ 

We claim that $G = B^n_\alpha$. Note first that $B^n_\alpha(\bar{X}_i) = T$ for $1 \leq i \leq k$. On the other hand, only one truth assignment for $\{A_1, \ldots, A_n\}$ can satisfy $\gamma_i$, hence only $k$ such truth assignment can satisfy $\alpha$. Hence $B^n_\alpha(\bar{Y}) = F$ for the $2^n - k$ other $n$-tuple $Y$. Thus in all cases, $B^n_\alpha(\bar{Y}) = G(\bar{Y})$. \hfill \Box

We say that $\alpha$ is in disjunctive normal form if $\alpha = \gamma_1 \lor \cdots \lor \gamma_k$, where $\gamma_i = \beta_{i1} \land \cdots \land \beta_{in}$, and each $\beta_{ij}$ is a sentence symbol or a negation of a sentence symbol.

Corollary 1 For any wff $\varphi$, we can find a tautologically equivalent wff in disjunctive normal form.

Let $C$ be a set of connectives. We say $C$ is complete if every Boolean function can be realised by a wff using only connectives from $C$. For example, we have just proved that $\{\neg, \lor, \land\}$ is complete.
Corollary 2 Both $\{\neg, \land\}$ and $\{\neg, \lor\}$ are complete.

Proof. Repeatedly apply De Morgan’s Law. \hfill $\Box$

Example. $\{\land, \rightarrow\}$ is not complete.

Proof. The idea is that with these connectives, if the sentence symbols are assigned $T$, then the entire formula is assigned $T$. In particular, there is nothing tautologically equivalent to $\neg A$. (Formally, we can show by induction that for any wff $\alpha$ using only these connectives and having $A$ as its only sentence symbol, we have $A \models \alpha$.) \hfill $\Box$

2.5 An Axiom System for Sentential Logic

In this section, we will formalise the concept “proof”. We first select a set $\Lambda$, often infinite, of wff’s called “axioms”. We also have a set of “inference rules”, which tells us how to get a new wff from certain others effectively. Then for a set $\Gamma$ of “hypothesis” or “premises”, the ”theorems” of $\Gamma$ are the wff’s which can be obtained from $\Gamma \cup \Lambda$ by use of the rules of inference some finite number of times. If $\varphi$ is a theorem of $\Gamma$, then a sequence of formulas which records how $\varphi$ was obtained from $\Gamma \cup \Lambda$ is called a “proof” of $\varphi$ from $\Gamma$.

We now introduce an axiom system $L$ for sentential logic. For simplicity, we assume that $\neg$ and $\rightarrow$ are the only primitive connectives, and $\alpha \land \beta$, $\alpha \lor \beta$ and $\alpha \leftrightarrow \beta$ are abbreviations of $\neg(\alpha \to \neg \beta)$, $\neg \alpha \to \beta$ and $(\alpha \to \beta) \land (\beta \to \alpha)$ respectively.

The axioms $\Lambda$ for $L$ are:

(A1) $\alpha \to (\beta \to \alpha)$;

(A2) $(\alpha \to (\beta \to \gamma)) \to ((\alpha \to \beta) \to (\alpha \to \gamma))$;

(A3) $(\neg \beta \to \neg \alpha) \to ((\neg \beta \to \alpha) \to \beta)$.

where $\alpha$, $\beta$ and $\gamma$ are wff’s.
There is only one inference rule for $L$, which is called *modus ponens* (MP), that is, from $\alpha$ and $\alpha \rightarrow \beta$, we may infer $\beta$.

**Definition 2** A deduction (or proof) of $\varphi$ from $\Gamma$ is a sequence $\langle \alpha_0, \ldots, \alpha_n \rangle$ of wffs, such that $\alpha_n = \varphi$ and for each $i \leq n$ either

(a) $\alpha_i$ is in $\Gamma \cup \Lambda$ or

(b) for some $j, k < i$, $\alpha_i$ is obtained from $\alpha_j$ and $\alpha_k$ by MP (i.e., $\alpha_k = \alpha_j \rightarrow \alpha_i$).

$\varphi$ is a theorem of $\Gamma$, written $\Gamma \vdash \varphi$, if there is a deduction of $\varphi$ from $\Gamma$.

Facts: (Can you verify them?)

1. If $\Gamma \subseteq \Delta$ and $\Gamma \vdash \alpha$, then $\Delta \vdash \alpha$.

2. $\Gamma \vdash \alpha$ iff there is a finite subset $\Gamma_0$ of $\Gamma$ such that $\Gamma_0 \vdash \alpha$.

3. If $\Delta \vdash \alpha$ and for each $\beta \in \Delta$ $\Gamma \vdash \beta$, then $\Gamma \vdash \alpha$.

**Lemma 2** For all wffs $\alpha$, $\vdash \alpha \rightarrow \alpha$.

**Proof.** We have the following deduction: (Can you specify the reasons in each step?)

1. $(\alpha \rightarrow (\alpha \rightarrow \alpha) \rightarrow \alpha) \rightarrow (\alpha \rightarrow (\alpha \rightarrow \alpha)) \rightarrow (\alpha \rightarrow \alpha)$

2. $\alpha \rightarrow (\alpha \rightarrow \alpha) \rightarrow \alpha$

3. $(\alpha \rightarrow (\alpha \rightarrow \alpha)) \rightarrow (\alpha \rightarrow \alpha)$

4. $\alpha \rightarrow (\alpha \rightarrow \alpha)$

5. $\alpha \rightarrow \alpha$. □

**Theorem 7** (Deduction Theorem) If $\Gamma$ is a set of wffs, and $\alpha$ and $\beta$ are wffs, and $\Gamma \cup \{\alpha\} \vdash \beta$, then $\Gamma \vdash \alpha \rightarrow \beta$. In particular, if $\alpha \vdash \beta$ then $\vdash \alpha \rightarrow \beta$.  

10
Proof. Let $\langle \beta_1, \ldots, \beta_n \rangle$ be a deduction sequence of $\beta$ from $\Gamma \cup \{\alpha\}$, where $\beta_n = \beta$. We prove, by induction on $i$ that $\Gamma \vdash \alpha \rightarrow \beta_i$, for $1 \leq i \leq n$.

For $i = 1$, $\beta_1$ is either in $\Gamma$ or an axiom or $\alpha$ itself. By (A1), $\beta_1 \rightarrow (\lambda \rightarrow \beta_1)$ is an axiom. Hence in the first two cases $\Gamma \vdash \alpha \rightarrow \beta_1$ by MP. For the third case, we just apply previous lemma.

Now assume that $\Gamma \vdash \alpha \rightarrow \beta_k$ for all $k < i$. Then either $\beta_i$ is in $\Gamma$ or an axiom or $\alpha$ itself, or $\beta_i$ follows from $\beta_j$ and $\beta_k = \beta_j \rightarrow \beta_k$ by MP for some $j, l < i$. In the first 3 cases, $\Gamma \vdash \alpha \rightarrow \beta_i$ as in the case $i = 1$ above. In the last case, we have, by induction hypotheses, $\Gamma \vdash \alpha \rightarrow \beta_j$ and $\Gamma \vdash \alpha \rightarrow (\beta_j \rightarrow \beta_i)$. By (A2)

$$(\alpha \rightarrow (\beta_j \rightarrow \beta_i)) \rightarrow (\alpha \rightarrow \beta_j) \rightarrow (\alpha \rightarrow \beta_i)$$

is an axiom. Hence by applying MP twice, we get $\Gamma \vdash \alpha \rightarrow \beta_i$. \hfill \Box

Corollary 3  
1. $\{\alpha \rightarrow \beta, \beta \rightarrow \gamma\} \vdash \alpha \rightarrow \gamma$.

2. $\{\alpha \rightarrow (\beta \rightarrow \gamma), \beta\} \vdash \alpha \rightarrow \gamma$.

We now establish the connection between the theorems in $L$ and tautologies.

**Theorem 8 (Soundness Theorem for sentential logic)** Every theorem of $L$ is a tautology. In other words, if $\vdash \alpha$ then $\models \alpha$.

Proof. (Sketch) First we verify that all the axioms are tautologies. Then we check that modus ponens leads from tautologies to tautologies. \hfill \Box

To show that every tautology is a theorem, we need the following facts and a technical lemma.

Facts:

1. If $\Gamma \vdash \alpha$ then $\Gamma \vdash \beta \rightarrow \alpha$ for any wff $\beta$.

2. $\vdash \neg \alpha \rightarrow (\alpha \rightarrow \beta)$.

3. If $\Sigma \vdash \alpha$ and $\Sigma \vdash \neg \beta$ then $\Sigma \vdash \neg (\alpha \rightarrow \beta)$. 

11
Lemma 3 Let \( \alpha \) be a wff and \( A_1, \ldots, A_k \) be the sentence symbols that occur in \( \alpha \). For a given truth assignment \( v \) of \( A_1, \ldots, A_k \), let \( A'_i \) be \( A_i \) if \( v(A_i) = T \) and be \( \neg A_i \) otherwise. Also let \( \alpha' \) be \( \alpha \) if \( \bar{v}(\alpha) = T \) and \( \neg \alpha \) otherwise. Then

\[
\{A'_1, \ldots, A'_k\} \vdash \alpha'.
\]

Proof. Let \( X \) be the set of all wffs \( \alpha \) satisfying the above stated property. We show that \( X \) is an inductive set. For notation simplicity we let \( \Sigma \) denote the set of \( \{A'_1, \ldots, A'_k\} \) where \( A_1, \ldots, A_k \) are all relevant sentential symbols, and we write \( v \) instead of \( \bar{v} \).

If \( \alpha \) is a sentential symbol, then \( \alpha \in X \) (Why?).

Suppose that \( \alpha \) is in \( X \) and \( \beta = \neg \alpha \). We show \( \beta \) is in \( X \). Let \( v \) be a truth assignment.

Case 1: \( v(\beta) = T \). Then \( v(\alpha) = F \). So \( \alpha' = \neg \alpha \). By induction hypothesis, \( \Sigma \vdash \alpha' \). Hence \( \Sigma \vdash \beta' \), because \( \beta' \equiv \beta = \neg \alpha = \alpha' \).

Case 2: \( v(\beta) = F \). Then \( v(\alpha) = T \). So \( \alpha' \equiv \alpha \). By induction hypothesis, \( \Sigma \vdash \alpha \). By a homework problem, we have \( \vdash \alpha \rightarrow \neg \alpha \). Hence \( \Sigma \vdash \beta' \), because \( \beta' \equiv \neg \alpha \).

Suppose that \( \alpha \) and \( \beta \) are in \( X \) and \( \gamma = \alpha \rightarrow \beta \). We show that \( \gamma \) is in \( X \). Let \( v \) be a truth assignment. There are 3 cases.

Case 1: \( v(\alpha) = F \). Hence \( v(\gamma) = T \). By induction hypothesis, \( \Sigma \vdash \neg \alpha \) because \( \alpha' \equiv \neg \alpha \). By fact (2) and MP we have \( \Sigma \vdash \alpha \rightarrow \beta \), which is \( \gamma' \).

Case 2: \( v(\beta) = T \). Hence \( v(\gamma) = T \). By induction hypothesis, \( \Sigma \vdash \beta \), because \( \beta' \equiv \beta \). By fact (1), we have \( \Sigma \vdash \alpha \rightarrow \beta \) which is \( \gamma' \).

Case 3: \( v(\alpha) = T \) and \( v(\beta) = F \). Hence \( v(\gamma) = F \). By induction hypothesis, \( \Sigma \vdash \alpha \) and \( \Sigma \vdash \neg \beta \). By fact (3), we have \( \Sigma \vdash \neg (\alpha \rightarrow \beta) \), which is \( \gamma' \). \( \square \)

Theorem 9 (Completeness Theorem for sentential logic) If \( \models \alpha \), then \( \vdash \alpha \), in other words, every tautology is a theorem.

Proof. Assume that \( \alpha \) is a tautology and let \( A_1, \ldots, A_k \) be the sentence symbols in \( \alpha \). By Lemma 3, for any truth assignment we have \( \{A'_1, \ldots, A'_k\} \vdash \alpha \).
(because \( \alpha' \) is \( \alpha \)). Thus \( \{ A_1', \ldots, A_k', A_k \} \vdash \alpha \) and \( \{ A_1', \ldots, A_k', \lnot A_k \} \vdash \alpha \). Thus by Deduction Theorem, both \( \{ A_1', \ldots, A_k' \} \vdash A_k \rightarrow \alpha \) and \( \{ A_1', \ldots, A_k' \} \vdash \lnot A_k \rightarrow \alpha \). Notice that for any wffs \( \gamma \) and \( \delta \)
\[
\vdash (\gamma \rightarrow \delta) \rightarrow (\lnot \gamma \rightarrow \delta) \rightarrow \delta,
\]
(can you prove it?). So by applying MP twice, we have \( \{ A_1', \ldots, A_k' \} \vdash \alpha \). Repeating the same argument, we have \( \vdash \alpha \). \( \square \)

We now prove Compactness Theorem. We call a set \( \Sigma \) of wffs \textit{satisfiable} iff there is a truth assignment which satisfies every member of \( \Sigma \). We call \( \Sigma \) \textit{finitely satisfiable} iff every finite subset of \( \Sigma \) is satisfiable.

**Theorem 10 (Compactness Theorem)** A set of wffs is satisfiable iff it is finitely satisfiable.

**Proof.** We only show the nontrivial direction. Assume that \( \Sigma \) is finitely satisfiable. We first extend \( \Sigma \) to a maximal such set \( \Delta \). Let \( \alpha_1, \alpha_2, \ldots \) be a fixed enumeration of all the wffs. Define by recursion on natural numbers:

\[
\Delta_0 = \Sigma,
\]

\[
\Delta_{n+1} = \begin{cases} 
\Delta_n \cup \{ \alpha_{n+1} \} & \text{if it is finitely satisfiable,} \\
\Delta_n \cup \{ \lnot \alpha_{n+1} \} & \text{otherwise.}
\end{cases}
\]

Then each \( \Delta_n \) is finitely satisfiable (homework). Let \( \Delta = \bigcup_n \Delta_n \). Then

(1) \( \Sigma \subseteq \Delta \) and

(2) for any wff \( \alpha \) either \( \alpha \in \Delta \) or \( \lnot \alpha \in \Delta \). Furthermore

(3) \( \Delta \) is finitely satisfiable. (Why?)

We now use \( \Delta \) to make a truth assignment satisfying \( \Sigma \). Define a truth assignment \( v \) by \( v(A) = T \) iff \( A \in \Delta \) for any sentence symbol \( A \). Then for any wff \( \varphi \), we have that \( v \) satisfies \( \varphi \) iff \( \varphi \in \Delta \) (homework). Since \( \Sigma \subseteq \Delta \), \( v \) must satisfy every member of \( \Sigma \). \( \square \)

13
**Corollary 4** If $\Sigma \models \tau$ then there is a finite $\Sigma_0 \subseteq \Sigma$ such that $\Sigma_0 \models \tau$.

**Proof.** We use the fact that $\Sigma \models \tau$ iff $\Sigma \cup \{\neg \tau\}$ is unsatisfiable.

Suppose that for all finite $\Sigma_0 \subseteq \Sigma$, $\Sigma_0 \nvdash \tau$. Then $\Sigma_0 \cup \{\neg \tau\}$ is satisfiable for every finite $\Sigma_0 \subseteq \Sigma$, that is, $\Sigma \cup \{\neg \tau\}$ is finitely satisfiable. By Compactness Theorem, $\Sigma \cup \{\neg \tau\}$ is satisfiable, a contradiction. \qed

## 3 First-Order Logic

### 3.1 First-Order Languages

A first-order language $L$ consists of

1. **parentheses**: (,);
2. **sentential connectives**: $\neg$ and $\rightarrow$;
3. **quantifier symbol**: $\forall$;
4. **variables**: $v_1, v_2, \ldots$;
5. **constant symbols**: some set (possibly empty) of symbols;
6. **function symbols**: for each positive integer $n$, some set (possibly empty) of symbols, called $n$-place (or $n$-ary) function symbols;
7. **predicate symbols**: for each positive integer $n$, some set (possibly empty) of symbols, called $n$-place predicates;
8. **equality symbol** (optional): $\approx$.

**Examples.**

1. The language of set theory is $L = \{\approx, \in\}$ (We often only specify the parameters from (4) to (7)).
2. The language of elementary number theory is $L = \{\approx, <, 0, S, +, \cdot, E\}$. (Can you tell which ones are predicates? And functions?)
Definition 3 Let $L$ be a first-order language. The set $T$ of terms is the smallest subset of expressions such that

1. Each variable $v_i$ is a term;
2. Each constant is a term;
3. If $t_1, \ldots, t_n$ are terms and $f$ is an $n$-ary function symbol, then $f(t_1, \ldots, t_n)$ is also a term.

Example. $S0$, $+v_1SS0$ and $ES0 + 0SS0$ are terms in the language of elementary number theory.

Definition 4 Let $L$ be a first-order language. The set of all well-formed formulas (or wffs, or formulas) is the smallest subset of expressions such that

1. If $t_1, \ldots, t_n$ are terms and $P$ is an $n$-place predicate, then $Pt_1, \ldots, t_n$ is a wff. We shall call wffs of this form atomic formulas. In particular, $\approx t_1t_2$ is an atomic formula;
2. If $\alpha$ and $\beta$ are wffs, then so are $(\neg \alpha)$ and $(\alpha \rightarrow \beta)$;
3. If $\alpha$ is a wff, then so is $\forall v_1 \alpha$.

Remarks on notations:

1. We introduce $\lor$, $\land$ and $\leftrightarrow$ as abbreviations for $((\neg \alpha) \rightarrow \beta)$, $(\neg(\alpha \rightarrow (\neg \beta)))$ and $((\alpha \rightarrow \beta) \land (\beta \rightarrow \alpha))$ respectively.
2. $\exists x \alpha$ abbreviates $(\neg \forall x(\neg \alpha))$.
3. $u \approx t$ abbreviates $\approx ut$ and similarly for other two-place predicates. $u \not\equiv t$ abbreviates $(\neg \approx ut)$.
4. As did in sentential logic, we often omit the outmost parentheses. When one connective is used repeatedly, the expression is grouped to the right. $\neg$, $\forall$ and $\exists$ apply to as little as possible and after that $\land$ and $\lor$ apply to as little as possible.
5. We usually (with exceptions) use uppercase letters, e.g. $P$, for predicates; $v_i$, $v$, $x, y, z$ for variables; $f, g, h, F$ for functions; $a, b, c$ for constants; $t$ for terms; lowercase Greek letters, e.g. $\alpha, \beta, \varphi, \sigma, \tau$ for formulas; and uppercase Greek letters, e.g., $\Gamma, \Delta, \Sigma$, for sets of formulas.

Consider a variable $x$. We define, for each wff $\alpha$, what it means for $x$ to occur free in $\alpha$ by recursion, as follows.

1. For atomic $\alpha$, $x$ occurs free in $\alpha$ iff $x$ occurs in $\alpha$.
2. $x$ occurs free in $(\neg \alpha)$ iff $x$ occurs free in $\alpha$.
3. $x$ occurs free in $(\alpha \to \beta)$ iff occurs free in $\alpha$ or in $\beta$.
4. $x$ occurs free in $\forall v_i \alpha$ iff $x$ occurs free in $\alpha$ and $x \neq v_i$.

If no variable occurs free in the wff $\alpha$, then $\alpha$ is a sentence.

### 3.2 Truth and Models

We now define the structure for a first-order language. Informally, a structure will tell us: What collection of things the universal quantifier symbol refers to and what the other parameters denote.

**Definition 5** A structure $\mathfrak{A}$ for a first-order language is a function whose domain is the set of parameters and such that

1. $\mathfrak{A}$ assigns to the quantifier symbol $\forall$ a nonempty set $|\mathfrak{A}|$, called the universe of $\mathfrak{A}$.
2. $\mathfrak{A}$ assigns to each $n$-place predicate symbol $P$ an $n$-ary relation $P^\mathfrak{A} \subseteq |\mathfrak{A}|^n$; i.e., $P^\mathfrak{A}$ is a set of $n$-tuples of members of the universe.
3. $\mathfrak{A}$ assigns to each constant symbol $c$ a member $c^\mathfrak{A}$ of the universe $|\mathfrak{A}|$.
4. $\mathfrak{A}$ assigns to each $n$-place function symbol $f$ an $n$-ary operation $f^\mathfrak{A}$ on $|\mathfrak{A}|$; i.e., $f^\mathfrak{A} : |\mathfrak{A}|^n \to |\mathfrak{A}|$. 

16
Example. Consider the language for set theory, whose only parameter (other than $\forall$) is $\in$. Take the structure $\mathfrak{A}$ with $\mathfrak{A}$ to be the set of natural numbers, and $\in^\mathfrak{A}$ to be the set of pairs $\langle m, n \rangle$ such that $m < n$. What are the translations of the sentences $\exists x \forall y \neg y \in x$ and

$$\forall x \forall y \exists t (t \in z \to t \approx x \lor \neg y)$$

Now we want to define “$\sigma$ is true in $\mathfrak{A}$”, written $\models_{\mathfrak{A}} \sigma$, for sentence $\sigma$ and structure $\mathfrak{A}$. We start with a more general notion involving wffs. Let $\varphi$ be a wff of our language, $\mathfrak{A}$ a structure for the language, $s : V \to |\mathfrak{A}|$ a function from the set $V$ of all variables into the universe $|\mathfrak{A}|$ of $\mathfrak{A}$. We will define what it means for $\mathfrak{A}$ to satisfy $\varphi$ with $s$, written $\models_{\mathfrak{A}} \varphi[s]$. The intuitive version is $\models_{\mathfrak{A}} \varphi[s]$ iff the translation of $\varphi$ determined by $\mathfrak{A}$, where the variable $x$ is translated as $s(x)$ wherever it occurs free, is true. Formally we have:

**Definition 6**

1. **Terms.** We define the extension $\bar{s} : T \to |\mathfrak{A}|$ by recursion as follows:

   (1) For each variable $x$, $\bar{s}(x) = s(x)$.
   
   (2) For each constant symbol $c$, $\bar{s}(c) = c^\mathfrak{A}$.
   
   (3) If $t_1, \ldots, t_n$ are terms and $f$ is an $n$-place function symbol, then

   $$\bar{s}(f t_1 \ldots t_n) = f^\mathfrak{A}(\bar{s}(t_1), \ldots, \bar{s}(t_n)).$$

2. **Atomic formulas.**

   (1) $\models_{\mathfrak{A}} t_1 t_2 [s]$ iff $\bar{s}(t_1) = \bar{s}(t_2)$.
   
   (2) For an $n$-place predicate parameter $P$, $\models_{\mathfrak{A}} P t_1 \ldots t_n [s]$ iff $\langle \bar{s}(t_1), \ldots, \bar{s}(t_n) \rangle \in P^\mathfrak{A}$.

3. **Other wffs.**

   (1) For atomic formulas, the definition is above.
   
   (2) $\models_{\mathfrak{A}} \neg \varphi[s]$ iff $\not\models_{\mathfrak{A}} \varphi[s]$.
   
   (3) $\models_{\mathfrak{A}} (\varphi \rightarrow \psi)[s]$ iff either $\not\models_{\mathfrak{A}} \varphi[s]$ or $\models_{\mathfrak{A}} \psi[s]$.
\( (4) \models_{\mathcal{A}} \forall \varphi[s] \text{ iff for every } d \in |\mathcal{A}|, \text{ we have } \models_{\mathcal{A}} \varphi[s(x|d)], \)
\[
s(x|d)(y) = \begin{cases} 
  s(y) & \text{if } y \neq x, \\
  d & \text{if } y = x.
\end{cases}
\]

**Definition 7** Let \( \Gamma \) be a set of wffs, \( \varphi \) a wff. Then \( \Gamma \) logically implies \( \varphi \), \( \Gamma \models \varphi \), iff for every structure \( \mathcal{A} \) for the language and every function \( s : V \rightarrow |\mathcal{A}| \) such that \( \mathcal{A} \) satisfies every member of \( \Gamma \) with \( s \), \( \mathcal{A} \) also satisfies \( \varphi \) with \( s \).

From now on, the symbol \( \models \) will be used only for logical implication, though it was used for tautological implication. As before, we write \( \gamma \models \varphi \) instead of \( \{\gamma\} \models \varphi \). We say that \( \varphi \) and \( \psi \) are logically equivalent (\( \varphi \equiv \psi \)) iff \( \varphi \models \psi \) and \( \psi \models \varphi \). A wff \( \varphi \) is valid iff \( \emptyset \models \varphi \) (written “\( \models \varphi \)”). Thus \( \varphi \) is valid iff for every \( \mathcal{A} \) and every \( s : V \rightarrow |\mathcal{A}| \), \( \mathcal{A} \) satisfies \( \varphi \) with \( s \).

**Theorem 11** Assume that \( s_1, s_2 \) are functions from \( V \) into \( |\mathcal{A}| \), which agree at all variables (if any) which occur free in the wff \( \varphi \). Then \( \models_{\mathcal{A}} \varphi[s_1] \) iff \( \models_{\mathcal{A}} \varphi[s_2] \).

**Proof.** Fix a structure \( \mathcal{A} \). We show by induction on wff \( \varphi \) that if two functions \( s_1 \) and \( s_2 \) agree on the variables free in \( \varphi \) with \( s_1 \), then \( \mathcal{A} \) satisfies \( \varphi \) iff it does so with \( s_2 \).

*Case 1:* \( \varphi \) is an atomic formula \( Pt_1 \ldots t_n \). Then we can show that \( s_1(t_i) = s_2(t_i) \) for each \( i \). (Can you prove it?) Hence \( \mathcal{A} \models Pt_1 \ldots t_n \) with \( s_1 \) if it does so with \( s_2 \).

*Case 2 and 3:* \( \varphi \) has the form \( \neg \alpha \) or \( \alpha \rightarrow \beta \): Skipped.

*Case 4:* \( \varphi \) is \( \forall x \psi \). Then the variables free in \( \varphi \) are those free in \( \psi \) except \( x \). Thus for any \( d \in |\mathcal{A}| \), \( s_1(x|d) \) and \( s_2(x|d) \) agree at all variables free in \( \psi \). By induction hypothesis, \( \mathcal{A} \) satisfies \( \psi \) with \( s_1(x|d) \) iff it does so with \( s_2(x|d) \). So \( \mathcal{A} \) satisfies \( \varphi \) with \( s_1 \) iff it does so with \( s_2 \). \( \square \)

**Corollary 5** For a sentence \( \sigma \), either

(a) \( \mathcal{A} \) satisfies \( \sigma \) with every function \( s \) from \( V \) into \( |\mathcal{A}| \), or
(b) $\mathfrak{A}$ does not satisfy $\sigma$ with any such function.

If alternative (a) holds, then we say that $\sigma$ is true in $\mathfrak{A}$ ($\models_\mathfrak{A} \sigma$) or that $\mathfrak{A}$ is a model of $\sigma$.

**Example.** Assume that our language has the parameters $\forall$, $P$ (a two-place predicate symbol), $f$ (a one-place function symbol) and $c$ (a constant symbol). Let $\mathfrak{A}$ be the structure

$$\mathfrak{A} = (\mathbb{N}, \leq, s, 0).$$

Let $s : V \to \mathbb{N}$ be the function for which $s(v_i) = i - 1$; i.e., $s(v_1) = 0$, $s(v_2) = 1$, etc. What are $\bar{s}(f f v_3)$ and $\bar{s}(f f c)$? Does $\mathfrak{A}$ satisfy the following formula with $s$?

1. $Pcv_1[s]$;
2. $\forall v_1 Pcv_1$;
3. $\forall v_1 P v_2 v_1$?

**Example.** Prove or disprove the following:

1. $\forall v_1 Qv_1 \models Qv_1$;
2. $Qv_1 \models \forall v_1 Qv_1$.

**Definability of a class of structures**

For a set $\Sigma$ of sentences, let $\text{Mod } \Sigma$ be the class of all models of $\Sigma$. For a single sentence $\tau$ we write ”$\text{Mod } \tau$” instead of ”$\text{Mod } \{\tau\}$”. A class $K$ of structures for a language is an elementary class (EC) iff $K$ is Mod $\tau$ for some sentence $\tau$. $K$ is an elementary class in the wider sense (EC$_\Delta$) iff $K$ is Mod $\Sigma$ for some set $\Sigma$ of sentences.

**Examples.**

1. Let the language $L = \{ \approx, P \}$ where $P$ is a two-place predicate symbol. Let $\tau$ be the conjunction of the three sentences

$$\forall x \forall y \forall z (xPy \rightarrow yPz \rightarrow xPz);$$
\[ \forall x \forall y (x P y \lor x \approx y \lor y P x); \]
\[ \forall x \forall y (x P y \rightarrow \neg y P x). \]

Then any model of \( T \) is a linearly ordered set. Hence the class of nonempty linearly ordered sets is an elementary class.

2. Assume that the language has \( \approx \) and a two-place function symbol \( \circ \). The class of all groups is an elementary class, being the class of all models of the conjunction of the group axioms (What are they?). Let

\[ \lambda_2 = \exists x \exists y (x \neq y), \]
\[ \lambda_3 = \exists x \exists y \exists z (x \neq y \land y \neq z \land z \neq x \neq z), \]

\[ \ldots \]

(\( \lambda_n \) translates “There are at least \( n \) things.”) Then the class of all infinite groups is \( EC_{\Delta} \), because it is the class of models of group axioms and \( \{\lambda_2, \lambda_3, \ldots\} \). Later we will see that the class of infinite groups is not \( EC \).

**Definability within a structure**

Consider a fixed structure \( \mathfrak{A} \). Suppose that \( \varphi \) is a formula such that all variables occurring free in \( \varphi \) are among \( v_1, \ldots, v_k \). Then for elements \( a_1, \ldots, a_k \) of \( |\mathfrak{A}| \),

\[ \models \mathfrak{A} \varphi \[ a_1, \ldots, a_k \] \]

means that \( \mathfrak{A} \) satisfies \( \varphi \) with some (and hence with any) functions \( s : V \to |\mathfrak{A}| \) for which \( s(v_i) = a_i \), \( 1 \leq i \leq k \). We call the \( k \)-ary relation

\[ \{ \langle a_1, \ldots, a_k \rangle : \models \mathfrak{A} \varphi \[ a_1, \ldots, a_k \] \} \]

the relation \( \varphi \) *defines* in \( \mathfrak{A} \). We say a \( k \)-ary relation on \( |\mathfrak{A}| \) *definable* if there is a formula \( \varphi \) which defines it in \( \mathfrak{A} \).

**Example.** Let \( L = \{0, S, +, \cdot\} \) be part of language for number theory. Let \( \mathfrak{A} \) be the structure with universe \( \mathbb{N} \), and other natural interpretation. Then the ordering relation \( \{ \langle m, n \rangle : m < n \} \) is definable (How?). For any natural number \( n \) \( \{n\} \) is definable (How?). And the set of prime numbers is definable in \( \mathfrak{A} \) as well (How?).

**Homomorphisms**
Definition 8 Let \( \mathfrak{A}, \mathfrak{B} \) be structures for a fixed language. A homomorphism 
\( h \) of \( \mathfrak{A} \) into \( \mathfrak{B} \) is a function \( h : |\mathfrak{A}| \to |\mathfrak{B}| \) such that

(a) For each \( n \)-place predicate \( P \) (other than \( \approx \)), and each \( a_1, \ldots, a_n \) in 
\[ |\mathfrak{A}|, \]
\[ \langle a_1, \ldots, a_n \rangle \text{ran} \in P^{\mathfrak{A}} \iff \langle h(a_1), \ldots, h(a_n) \rangle \in P^{\mathfrak{B}}. \]

(b) For each \( n \)-place function symbol \( f \) and each such \( n \)-tuple,
\[ h(f^{\mathfrak{A}}(a_1, \ldots, a_n)) = f^{\mathfrak{B}}(h(a_1), \ldots, h(a_n)). \]

(c) For each constant symbol \( c \), \( h(c^{\mathfrak{A}}) = c^{\mathfrak{B}}. \)

In the above definition, if \( h \) is one-to-one, then \( h \) is called an isomorphism 
of \( \mathfrak{A} \) into \( \mathfrak{B} \). If \( h \) is bijective, then \( h \) is called an isomorphism of \( \mathfrak{A} \) onto \( \mathfrak{B} \), 
and \( \mathfrak{A} \) is said to be isomorphic to \( \mathfrak{B} \), denoted by \( \mathfrak{A} \cong \mathfrak{B} \).

Theorem 12 (Homomorphism Theorem) Let \( h \) be a homomorphism of 
\( \mathfrak{A} \) into \( \mathfrak{B} \), and \( s : V \to |\mathfrak{A}|. \)

(a) For any term \( t \), \( h(\bar{s}(t)) = \overline{h \circ s(t)}. \)

(b) For any quantifier-free formula \( \alpha \) not containing the equality symbol,
\[ \models_{\mathfrak{A}} \alpha[\bar{s}] \iff \models_{\mathfrak{B}} \alpha[h \circ s]. \]

(c) If \( h \) is one-to-one, then in part (b) we may delete the restriction “not 
containing the equality symbol”.

(d) If \( h \) is a homomorphism of \( \mathfrak{A} \) onto \( \mathfrak{B} \), then in (b) we may delete the 
restriction “quantifier-free”.

Proof. (a) We do an induction on term \( t \). If \( t \) is a constant \( c \), then
\[ h(\bar{s}(c)) = h(c^{\mathfrak{A}}), \] which is \( c^{\mathfrak{B}} = \overline{h \circ s(c)} \), since \( h \) is a homomorphism. If \( t \) is a 
variable \( v \), then both sides are \( h(\bar{s}(v)). \) If \( t \) is \( f(t') \) for some unary function 
symbol \( f \), (the same proof will work for \( n \)-ary functions), then
\[
\begin{align*}
h(\bar{s}(f(t'))) &= h(f^{\mathfrak{A}}(\bar{s}(t'))) \\
&= f^{\mathfrak{B}}(h(\bar{s}(t'))) \\
&= f^{\mathfrak{B}}(h \circ s(t')) \\
&= \overline{h \circ s(f(t')).}
\end{align*}
\]
(b) Let \( \alpha \) be any quantifier free formula without \( \approx \), we show that \( \models_{\bar{\alpha}} \alpha[s] \) iff \( \models_{\bar{\alpha}} \alpha[h \circ s] \) by induction on \( \alpha \).

If \( \alpha \) is an atomic formula, say, \( Pt \) for some unary predicate \( P \) (the same proof works for \( n \)-ary predicates as well). Then
\[
\models_{\bar{\alpha}} Pt[s] \iff \bar{s}(t) \in P^\alpha \\
\implies h(\bar{s}(t)) \in P^{\bar{\alpha}} \\
\implies h \circ \bar{s}(t) \in P^{\bar{\alpha}} \\
\iff \models_{\bar{\alpha}} Pt[h \circ s].
\]

For the inductive case, when \( \alpha = \alpha' \) or \( \alpha = \beta \rightarrow \gamma \), the proof is routine.

(c) Regardless \( h \) is one-to-one or not, we have
\[
\models_{\bar{\alpha}} u \approx t[s] \iff \bar{s}(u) = \bar{s}(t) \\
\implies h(\bar{s}(u)) = h(\bar{s}(t)) \\
\implies h \circ \bar{s}(u) = h \circ \bar{s}(t) \\
\iff \models_{\bar{\alpha}} u \approx t[h \circ s].
\]

If \( h \) is one-to-one then the \( \Rightarrow \) above can be reversed.

(d) Regardless \( h \) is onto or not, we have
\[
\models_{\bar{\alpha}} \forall x \beta[s] \iff \text{for any } d \in |\mathfrak{A}| \models_{\bar{\alpha}} \beta[s(x|d)] \\
\iff \text{for any } d \in |\mathfrak{A}| \models_{\bar{\alpha}} \beta[h \circ (s(x|d))] \\
\iff \text{for any } e \in |\mathfrak{B}| \models_{\bar{\beta}} [(h \circ s)(x|e)] \\
\iff \models_{\bar{\beta}} \forall x \beta[h \circ s].
\]

If \( h \) is onto then the \( \Leftarrow \) above can be reversed. \( \square \)

We say two structures \( \mathfrak{A} \) and \( \mathfrak{B} \) are elementarily equivalent (written \( \mathfrak{A} \equiv \mathfrak{B} \)) iff for any sentence \( \sigma \), \( \models_{\mathfrak{A}} \sigma \) iff \( \models_{\mathfrak{B}} \sigma \). By the homomorphism theorem we know that two isomorphic models are elementarily equivalent. Later we will see that they are elementarily equivalent structures which are not isomorphic.

An automorphism of the structure \( \mathfrak{A} \) is an isomorphism of \( \mathfrak{A} \) onto \( \mathfrak{A} \). As a consequence of the homomorphism theorem, we can show that an automorphism must preserve the definable relations:

22
**Corollary 6**  Let \( h \) be an automorphism of the structure \( \mathfrak{A} \), and let \( R \) be an \( n \)-ary relation on \( |\mathfrak{A}| \) definable in \( \mathfrak{A} \). Then for any \( a_1, \ldots, a_n \) in \( |\mathfrak{A}| \),

\[
\langle a_1, \ldots, a_n \rangle \in R \iff \langle h(a_1), \ldots, h(a_n) \rangle \in R.
\]

**Proof.** Let \( \varphi \) be the formula which defines \( R \) in \( \mathfrak{A} \). By Homomorphism Theorem, we have

\[
\models_{\mathfrak{A}} \varphi [ a_1, \ldots, a_n ] \iff \models_{\mathfrak{A}} \varphi [ h(a_1), \ldots, h(a_n) ]
\]

by taking \( s : V \to |\mathfrak{A}| \) to be \( s(v_i) = a_i \), where \( v_i \) are the free variables in \( \varphi \) (1 \( \leq i \leq n \)). Hence

\[
\langle a_1, \ldots, a_n \rangle \in R \iff \langle h(a_1), \ldots, h(a_n) \rangle \in R,
\]

that is what we want. \( \square \)

**Example.** Consider the structure \((\mathbb{R}, <)\) consisting of the set of real numbers with its usual ordering. Then \( h : \mathbb{R} \to \mathbb{R} \) defined by \( h(x) = x^3 \) is an automorphism (Why?). We can use \( h \) to show that \( \mathbb{N} \) is not definable in this structure (How?).

### 3.3 A Deductive Calculus

Let \( \Lambda \) be a set of axioms and \( \Gamma \) a set of wffs. Recall that a deduction of \( \varphi \) from \( \Gamma \) is a sequence \( \langle \alpha_0, \ldots, \alpha_n \rangle \) of formulas such that \( \alpha_n = \varphi \) and for each \( i \leq n \) either \( \alpha_i \) is in \( \Gamma \cup \Lambda \), or for some \( j, k < i \), \( \alpha_i \) is obtained by modus ponens from \( \alpha_j \) and \( \alpha_k \) (i.e., \( \alpha_k = \alpha_j \to \alpha_i \)). We say \( \varphi \) is a theorem of \( \Gamma \) (written \( \Gamma \vdash \varphi \)), if there is a deduction of \( \varphi \) from \( \Gamma \). It is easy to see that the set of theorems of \( \Gamma \) is the set of formulas generated from \( \Gamma \cup \Lambda \) by modus ponens.

We now give the set \( \Lambda \) of logical axioms for first-order logic. The axioms are arranged in six groups. We say that a wff \( \varphi \) is a generalization of \( \psi \) iff for some \( n \geq 0 \) and some variable \( x_1, \ldots, x_n \),

\[
\varphi = \forall x_1 \ldots \forall x_n \psi.
\]

The logical axioms are all generalizations of wffs of the following forms, where \( x \) and \( y \) are variables and \( \alpha \) and \( \beta \) are wffs:

23
1. Tautologies;
2. $\forall x \alpha \rightarrow \alpha^x_t$, where $t$ is substitutable for $x$ in $\alpha$;
3. $\forall x (\alpha \rightarrow \beta) \rightarrow (\forall x \alpha \rightarrow \forall x \beta)$;
4. $\alpha \rightarrow \forall x \alpha$, where $x$ does not occur free in $\alpha$.

And if the language includes equality, then we add

5. $x \approx x$;
6. $x \approx y \rightarrow (\alpha \rightarrow \alpha')$, where $\alpha$ is atomic and $\alpha'$ is obtained from $\alpha$ by replacing $x$ in zero or more (but not necessarily all) places by $y$.

We now take a closer look at groups 1 and 2.

**Substitution**

In axiom group 2, $\alpha^x_t$ is the expression obtained from the formula $\alpha$ by replacing the variable $x$, whenever it occurs free in $\alpha$, by the term $t$. It can also be defined by recursion, (Can you do it?).

**Example.** Let $\alpha$ be $\neg \forall y x \approx y$. Then $\forall x \alpha \rightarrow \alpha^x_2$ is

$$\forall x \neg \forall y x \approx y \rightarrow \neg \forall y z \approx y.$$ 

And $\forall x \alpha \rightarrow \alpha^x_3$ is

$$\forall x \neg \forall y x \approx y \rightarrow \neg \forall y y \approx y.$$ 

The last example makes us add conditions on the substitution. Informally, we say that a term $t$ is not substitutable for $x$ in $\alpha$ if there is some variable $y$ in $t$ that is captured by a $\forall y$ quantifier in $\alpha^x_t$. Formally, we define the phrase “$t$ is substitutable for $x$ in $\alpha$ as follows:

1. For atomic $\alpha$, $t$ is always substitutable for $x$ in $\alpha$.
2. $t$ is substitutable for $x$ in $\neg \alpha$ iff it is substitutable for $x$ in $\alpha$. $t$ is substitutable for $x$ in $\alpha \rightarrow \beta$ iff it is substitutable for $x$ in both $\alpha$ and $\beta$. 

24
3. \( t \) is substitutable for \( x \) in \( \forall y \alpha \) iff either

(a) \( x \) does not occur free in \( \forall y \alpha \), or

(b) \( y \) does not occur in \( t \) and \( t \) is substitutable for \( x \) in \( \alpha \).

**Tautologies**

Now we look at axiom group 1. We call a first-order formula a *tautology* if it can be obtained from a tautology of sentential logic by replacing each sentence symbol by a first-order formula. For example,

\[
(\forall y \neg P y \rightarrow \neg P x) \rightarrow (P x \rightarrow \neg \forall y \neg P y)
\]

is a tautology (Why?).

There is another way of looking at axiom group 1. We say that a wff is a *prime* formula iff it is either an atomic formula or of the form \( \forall x \alpha \). Now for any wff, we can replace the prime subformulas by sentence symbols. If the resulting formula is a sentential tautology, then the original wff is in axiom group 1.

**Theorem 13** \( \Gamma \vdash \varphi \) iff \( \Gamma \cup \Lambda \) tautologically implies \( \varphi \).

**Proof.** (\( \Rightarrow \)): Suppose that we have a truth assignment \( v \) which satisfies every member of \( \Gamma \cup \Lambda \). By induction we can see that \( v \) satisfies any theorem of \( \Gamma \), because \( \{ \alpha, \alpha \rightarrow \beta \} \) tautologically implies \( \beta \).

(\( \Leftarrow \)): Assume that \( \Gamma \cup \Lambda \) tautologically implies \( \varphi \). Then by the corollary to the compactness theorem for sentential logic, there is a finite subset \( \{ \gamma_1, \ldots, \gamma_m, \lambda_1, \ldots, \lambda_n \} \) which tautologically implies \( \varphi \). Consequently,

\[
\gamma_1 \rightarrow \cdots \rightarrow \gamma_m \rightarrow \lambda_1 \rightarrow \cdots \rightarrow \lambda_n \rightarrow \varphi
\]

is a tautology and hence is in \( \Lambda \). By applying modus ponens \( m + n \) times, we obtain \( \varphi \). \( \square \)

**Deductions and Metatheorems**

The following generalization theorem reflects our informal feeling that if we can prove a statement \( \alpha(x) \) without any special assumption about \( x \), we then are entitled to say that “since \( x \) was arbitrary, we have \( \forall x \alpha(x) \).”
Theorem 14 (Generalization Theorem) If $\Gamma \vdash \varphi$ and $x$ does not occur free in any formula in any formula in $\Gamma$, then $\Gamma \vdash \forall x \varphi$.

Proof. Let $\langle \varphi_1, \ldots, \varphi_n \rangle$ be a deduction sequence of $\varphi$, which is $\varphi_n$. We do induction on $i \leq n$ to show that $\Gamma \vdash \forall x \varphi_i$.

Case 1: $\varphi_i$ is a logical axiom. Then $\forall x \varphi_i$ is also a logical axiom (Why?). And so $\Gamma \vdash \varphi_i$. Notice that in this case, $x$ can occur free in $\varphi_i$, but it does not matter.

Case 2: $\varphi_i$ is in $\Gamma$. Then $x$ does not occur free in $\varphi_i$. Hence $\varphi_i \rightarrow \forall x \varphi_i$ is in axiom group 4. Thus by MP we get $\Gamma \vdash \forall x \varphi_i$.

Case 3: $\varphi_i$ is obtained by MP from $\varphi_j$ and $\varphi_k = \varphi_j \rightarrow \varphi_i$, for some $j, k < i$. Then by induction hypothesis, $\Gamma \vdash \forall x \varphi_j$ and $\Gamma \vdash \forall x (\varphi_i \rightarrow \varphi_j)$. Applying MP twice to the axiom

$$\forall x (\varphi_j \rightarrow \varphi_i) \rightarrow (\forall x \varphi_j \rightarrow \forall x \varphi_i),$$

we get $\Gamma \vdash \forall x \varphi_i$. \qed

Example. $\forall x \forall y \alpha \vdash \forall y \forall x \alpha$. (Can you prove it?)

Lemma 4 (Rule T) If $\Gamma \vdash \alpha_1, \ldots, \Gamma \vdash \alpha_n$ and $\\{\alpha_1, \ldots, \alpha_n\}$ tautologically implies $\beta$, then $\Gamma \vdash \beta$.

Proof. $\alpha_1 \rightarrow \ldots \rightarrow \alpha_n \rightarrow \beta$ is a tautology, and hence a logical axiom. Apply MP $n$ times. \qed

Theorem 15 (Deduction Theorem) $\Gamma \cup \{\gamma\} \vdash \varphi$ iff $\Gamma \vdash (\gamma \rightarrow \varphi)$.

Proof. Similar to the proof of Deduction Theorem for sentential logic. \qed

Corollary 7 (Contraposition) $\Gamma \cup \{\varphi\} \vdash \neg \psi$ iff $\Gamma \cup \{\psi\} \vdash \neg \varphi$.

We say that a set of formulas is inconsistent iff for some $\beta$, both $\beta$ and $\neg \beta$ are theorems of the set. In this event, any formula $\alpha$ is a theorem of the set (Why?).

26
Corollary 8 (Reductio ad absurdum) If $\Gamma \cup \{ \varphi \}$ is inconsistent, then $\Gamma \vdash \neg \varphi$.

The proofs of both corollary are skipped.

**Example.** $\vdash \exists x \forall y \varphi \rightarrow \forall y \exists x \varphi$.

**Proof.** We work backward to find a proof.

It suffices to show that $\exists x \forall y \varphi \vdash \forall y \exists x \varphi$ (Why?).

It suffices to show that $\exists x \forall y \varphi \vdash \exists x \varphi$ (Why?), which is $\neg \forall x \neg \forall y \varphi \vdash \neg \forall x \neg \varphi$.

It suffices to show that $\forall x \neg \varphi \vdash \forall x \neg \forall y \varphi$ (Why?).

It suffice to show that $\forall x \neg \varphi \vdash \neg \forall y \varphi$ (Why?).

It suffices to show that $\{ \forall x \neg \varphi, \forall y \varphi \}$ is inconsistent (Why?), which can be done easily (How?). \qed

Some useful tips for proving $\varphi$ from $\Gamma$:

1. Suppose that $\varphi$ is $(\psi \rightarrow \theta)$. Then it will suffice to show that $\Gamma \cup \{ \psi \} \vdash \theta$.

2. Suppose that $\varphi$ is $\forall x \psi$. If $x$ does not occur free in $\Gamma$, then it will suffice to show that $\Gamma \rightarrow \psi$. Even if $x$ occurs free in $\Gamma$, there will be a variable $y$ such that $\Gamma \vdash \forall y \psi^x_y$ and $\forall y \psi^x_y \vdash \forall x \psi$. (See homework.)

3. Suppose that $\varphi$ is the negation of another formula.

   3a. If $\varphi$ is $\neg (\psi \rightarrow \theta)$, then it will suffice to show that $\Gamma \vdash \psi$ and $\Gamma \vdash \neg \theta$.

   3b. If $\varphi$ is $\neg \neg \psi$, then it will suffice to show that $\Gamma \vdash \psi$.

   3c. If $\varphi$ is $\neg \forall x \psi$. Try to find some term $t$ substitutable for $x$ in $\psi$ and $\Gamma \vdash \neg t$, which will suffice, but not always possible. Try to use contraposition. And try reductio ad absurdum.

**Theorem 16 (Generalization on Constants)** Assume that $\Gamma \vdash \varphi$ and that $c$ is a constant symbol which does not occur in $\Gamma$. Then there is a variable $y$, which does not occur in $\varphi$, such that $\Gamma \vdash \varphi^c_y$. Furthermore, there is a deduction of $\forall y \varphi^c_y$ from $\Gamma$ in which $c$ does not occur.
Proof. Let \( \langle \alpha_0, \ldots, \alpha_n \rangle \) be a deduction of \( \varphi \) from \( \Gamma \), let \( y \) be a variable which does not occur in any of the \( \alpha_i \). We claim that

\[
\langle (\alpha_0)_y^c, \ldots, (\alpha_n)_y^c \rangle \tag{*}
\]

is a deduction of \( \varphi_y^c \) from \( \Gamma \).

Case 1: \( \alpha_k \) is from \( \Gamma \). Then \( (\alpha_k)_y^c \) is \( \alpha_k \), because \( c \) does not occur in \( \alpha_k \).

Case 2: \( \alpha_k \) is an axiom. Then \( (\alpha_k)_y^c \) is still a logical axiom. For example, if \( \alpha_k \) is of the form \( \forall x \beta \rightarrow \beta^c_t \), then \( (\alpha_k)_y^c \) is \( \forall x (\beta)_y^c \rightarrow (\beta^c_t)_y^c \), notice that \( (\beta^c_t)_y^c = (\beta^c_y)_y^c \), so it is still a logical axiom. Other groups of axioms are easily verified.

Case 3: \( \alpha_k \) is obtained from \( \alpha_i \) and \( \alpha_j = \alpha_i \) rar \( \alpha_k \) by MP for some \( i, j < k \). Then \( (\alpha_j)_y^c = (\alpha_i)_y^c \rightarrow (\alpha_k)_y^c \). Hence \( (\alpha_k)_y^c \) is obtained from \( (\alpha_i)_y^c \) and \( (\alpha_j)_y^c \) by MP.

Therefore the claim is established. Furthermore, let \( \Gamma_0 \) be the finite subset of \( \Gamma \) used in \( (*) \). We have \( \Gamma_0 \models \varphi_y^c \) and \( y \) does not occur in \( \Gamma_0 \). So by Generalization Theorem, \( \Gamma_0 \models \forall y \varphi_y^c \), hence \( \Gamma \models \forall y \varphi_y^c \). Since the proof of the Generalization Theorem does not introduce any new constants, there is a deduction of \( \forall y \varphi_y^c \) from \( \Gamma \) in which \( c \) does not occur. \( \square \)

Lemma 5 (Re-replacement Lemma) If \( y \) does not occur at all in \( \varphi \), then \( x \) is substitutable for \( y \) in \( \varphi_y^x \) and \( (\varphi_y^x)_x = \varphi \).

Proof. Homework. \( \square \)

Corollary 9 Assume that \( \Gamma \models \varphi_x^c \), where the constant symbol \( c \) does not occur in \( \Gamma \) or in \( \varphi \). Then \( \Gamma \models \forall x \varphi \), and there is a deduction of \( \forall x \varphi \) from \( \Gamma \) in which \( c \) does not occur.

Proof. By the Generalization on Constant Theorem, we have a deduction without \( c \) from \( \Gamma \) of \( \forall y (\varphi_x^c)_y \) which is \( \forall y \varphi_y^c \) since \( c \) does not occur in \( \varphi \). By Re-replacement Lemma and Generalization Theorem, we have \( \forall y \varphi_y^c \models \forall x \varphi \). (Why?) \( \square \)

Corollary 10 (Rule EI) Assume that the constant symbol \( c \) does not occur in \( \varphi \), \( \psi \) or \( \Gamma \), and that \( \Gamma \cup \{ \varphi_x^c \} \models \psi \). Then \( \Gamma \cup \{ \exists x \varphi \} \models \psi \) and there is a deduction of \( \psi \) from \( \Gamma \cup \{ \exists x \varphi \} \) in which \( c \) does not occur.

28
Proof. Contraposition, previous corollary, contraposition. □

"EI" stands for "existential instantiation". It is a formal way of saying: "We know there is an $x$ such that $P(x)$. So call it $c$. Now from $P(c)$ we can prove $\psi$.

Example. Revisit $\vdash \exists x \forall y \varphi \rightarrow \forall y \exists x \varphi$. (See p.117 on Enderton)

**Theorem 17 (Existence of Alphabetic Variants)** Let $\varphi$ be a formula, $t$ a term, and $x$ a variable. Then we can find a formula $\varphi'$ (which differs from $\varphi$ only in the choice of quantified variables) such that

(a) $\varphi \vdash \varphi'$ and $\varphi' \vdash \varphi$;

(b) $t$ is substitutable for $x$ in $\varphi'$.

**Proof.** Fix $t$ and $x$, we construct $\varphi'$ by recursion on $\varphi$. For atomic $\varphi$ we take $\varphi' = \varphi$ and then $(\neg \varphi)' = (\neg \varphi')$, $(\varphi \rightarrow \psi)' = (\varphi' \rightarrow \psi')$.

The remaining case is to find $(\forall y \varphi)'$. Choose a variable $z$ which does not occur in $\varphi'$ or $t$ or $x$. Then define $(\forall y \varphi)' = \forall z(\varphi')_z$, (b) is true by induction hypothesis and the choice of $z$ (Why?).

To verify that (a) is holds, we calculate as follows.

By induction hypothesis, we have $\varphi \vdash \varphi'$. So $\forall y \varphi \vdash \forall y \varphi'$. Now $\forall y \varphi' \vdash (\varphi')^y_z$ since $z$ does not occur in $\varphi'$. By generalization, we have $\forall y \varphi' \vdash \forall z(\varphi')_z$. Hence $\forall y \varphi \vdash \forall z(\varphi')_z$.

In the other direction, $\forall z(\varphi')_z \vdash (\varphi')^z_z$, which is $\varphi'$ by Re-placement Lemma. Now by induction hypothesis, $\varphi' \vdash \varphi$. Thus $\forall z(\varphi')_z \vdash \varphi$. By Generalization Theorem, $\forall z(\varphi')_z \vdash \forall y \varphi$. The last step uses the fact that $y$ does not occur free in $(\varphi')_z^y$ unless $y = z$, and so does not occur free in $\forall z(\varphi')_z^y$ in any case. □

We now list some properties involving $\approx$, which we will use later. We leave all proofs as exercises.

Eq1: $\forall x \ x \approx x$. 

29
Eq2: $\forall x \forall y (x \approx y \rightarrow y \approx x)$.

Eq3: $\forall x \forall y \forall z (x \approx y \rightarrow y \approx z \rightarrow x \approx z)$.

The above facts say that $\approx$ is an equivalence relation. The rest says that $\approx$ is compatible with the predicate and function symbols. Though we state them for 2-variable symbols $P$ and $f$, they can be generalized to $n$-variable symbols easily.

Eq4: $\forall x_1 \forall x_2 \forall y_1 \forall y_2 (x_1 \approx y_1 \rightarrow x_2 \approx y_2 \rightarrow Px_1x_2 \rightarrow Py_1y_2)$.

Eq5: $\forall x_1 \forall x_2 \forall y_1 \forall y_2 (x_1 \approx y_1 \rightarrow x_2 \approx y_2 \rightarrow fx_1x_2 \approx fy_1y_2)$.

### 3.4 Soundness and Completeness Theorems

**Theorem 18 (Soundness Theorem)** Let $\Gamma$ be a set of formulas and $\varphi$ be a formula. If $\Gamma \vdash \varphi$, then $\Gamma \models \varphi$.

Recall that if $\Gamma \models \psi$ and $\Gamma \models \psi \rightarrow \varphi$ then $\Gamma \models \varphi$. It suffices to verify that every axiom is valid (Why?).

By the fact (previous HW) that $\theta$ is valid iff $\forall x \theta$ is valid, we know that any generalization of a valid formula is valid. So we only need to look at the axioms which is not a generalization of a formula. Now we look at each axiom group.

For group 3, $\forall x (\alpha \rightarrow \beta) \rightarrow \forall \alpha \rightarrow \forall x \beta$, we can use the fact $\{\forall x (\alpha \rightarrow \beta), \forall x \alpha\} \models \forall x \beta$, which is a previous HW.

For group 4, $\alpha \rightarrow \forall x \alpha$ where $x$ does not occur free in $\alpha$, we again use a previous HW problem which says that if $x$ does not occur free in $\alpha$ then $\alpha \models \forall x \alpha$.

For group 5, $x \approx x$, easy (Exercise).

For group 1, tautologies, See homework.

For group 6, $x \approx y \rightarrow \alpha \rightarrow \alpha'$ where $\alpha$ is atomic and $\alpha'$ is obtained from $\alpha$ by replacing $x$ at some places by $y$. It suffices to show that $\{x \approx y, \alpha\} \models \alpha'$. Take any $A, s$ such that $\models_{A} x \approx y[s]$, i.e., $s(x) = s(y)$. By an easy induction
on $t$, we can show that $\bar{s}(t) = \bar{s}(t')$ where $t'$ is obtained from $t$ by replacing $x$ at some places by $y$. If $\alpha$ is $t_1 \approx t_2$, then $\alpha'$ must be $t_1' \approx t_2'$, thus $|=_{\alpha} \alpha[s]$ iff $\bar{s}(t_1) = \bar{s}(t_2)$ iff $\bar{s}(t_1') = \bar{s}(t_2')$ iff $|=_{\alpha'} \alpha'[s]$. Similarly we can prove when $\alpha$ is $Pt_1 \ldots t_n$.

So it remains to check group 2, substitution. We first prove a lemma.

**Lemma 6 (Substitution Lemma)** If the term $t$ is substitutable for the variable $x$ in the wff $\varphi$, then $|=_{\alpha} \varphi^x_t[s]$ iff $|=_{\alpha} \varphi[s(x|\bar{s}(t))].$

**Proof.** We use induction on $\varphi$.

*Case 1:* $\varphi$ is atomic. First observe that for any terms $u$ and $t$, $\bar{s}(u^x_t) = s(x|\bar{s}(t))(u)$, which can be proved by an induction on $u$. For simplicity, let us assume that $\varphi$ is $Pu$ for some term $u$, then $|=_{\alpha} Pu^x_t[s]$ iff $\bar{s}(u^x_t) \in P^x$, iff $s(x|\bar{s}(t))(u) \in P^x$ (by the observation) iff $|=_{\alpha} Pu[s(x|\bar{s}(t))].$

*Case 2:* $\varphi$ is $\neg \psi$ or $\psi \rightarrow \theta$. Easy.

*Case 3:* $\varphi$ is $\forall y \psi$, and $x$ does not occur free in $\varphi$. Just observe that $s$ and $s(x|\bar{s}(t))$ agree on all variables which occur free in $\varphi$ and $\varphi^x_t$ is just $\varphi$.

*Case 4:* $\varphi$ is $\forall y \psi$ and $x$ does occur free in $\varphi$. Because $t$ is substitutable for $x$ in $\varphi$, we know that $y$ does not occur free in $t$ and $t$ is substitutable for $x$ in $\psi$. Therefore, for any $d$ in $[\mathcal{A}]$, $\bar{s}(t) = s(y|d)(t)$. Since $x \neq y$, $\varphi^x_t = \forall y \psi^x_t$. Hence $|=_{\alpha} \varphi^x_t[s]$ iff for every $d$, $|=_{\alpha} \psi^x_t[s(y|d)]$, iff for every $d$, $|=_{\alpha} \psi[s(y|d)(x|\bar{s}(t))]|$ by induction hypothesis iff $|=_{\alpha} \varphi[s(x|\bar{s}(t))].$

So by induction the lemma holds for all $\varphi$. \qed

Now we go back to axiom group 2. Suppose that $t$ is substitutable for $x$ in $\varphi$, and $|=_{\alpha} \forall x \varphi[s]$. We need to show that $|=_{\alpha} \varphi^x_t[s]$. We know that for any $d$ in $[\mathcal{A}]$, $|=_{\alpha} \varphi[s(x|d)]$. In particular, let $d = \bar{s}(t)$, we have $|=_{\alpha} \varphi[s(x|\bar{s}(t))].$ By Substitution Lemma, $|=_{\alpha} \varphi^x_t[s]$. That establishes the Soundness Theorem. The following corollaries follows easily from the Soundness Theorem. (Can you prove them?)

**Corollary 11** If $\vdash (\varphi \leftrightarrow \psi)$, then $\varphi$ and $\psi$ are logically equivalent.

**Corollary 12** If $\Gamma$ is satisfiable, that is, there are $\mathcal{A}$ and $s$ such that $\mathcal{A}$ satisfies every member of $\Gamma$ with $s$, then $\Gamma$ is consistent.
We now prove the converse of the soundness theorem.

**Theorem 19 (Completeness Theorem, Gödel, 1930)**  
(a) If $\Gamma \vdash \varphi$, then $\Gamma \models \varphi$.

(b) Any consistent set of formulas is satisfiable.

**Overview.** Because (a) and (b) are equivalent (HW), we only prove (b). We prove only for a countable language. There are 6 steps. In steps 1-3 we extend $\Gamma$ to a set $\Delta$ of formulas for which

(i) $\Gamma \subseteq \Delta$.

(ii) $\Delta$ is consistent and is maximal in the sense that for any formula $\alpha$ either $\alpha \in \Delta$ or $(\neg \alpha) \in \Delta$.

(iii) For any formula $\varphi$ and variable $x$, there is a $c$ such that

$$(\neg \forall x \varphi \rightarrow \neg \varphi^c) \in \Delta.$$  

Then in step 4 we form a structure $\mathfrak{A}$ in which members of $\Gamma$ not containing $\approx$ can be satisfied. $|\mathfrak{A}|$ is the set of terms, and for a predicate symbol $P$,

$$\langle t_1, \ldots, t_n \rangle \in P^\mathfrak{A} \text{ iff } P t_1 \ldots t_n \in \Delta.$$  

Finally in step 5 and 6, we change $\mathfrak{A}$ so that formulas containing $\approx$ is also satisfied.

**Proof.** Let $\Gamma$ be a consistent set of wffs in a countable language.

*Step 1:* Expand the language $L$ to $L_C$ by adding a countably infinite set of new constant symbols $C = \{c_0, c_1, \ldots\}$. Then $\Gamma$ remains consistent in the new language.

Details: If not then for some $\beta$, there is a deduction (in $L_C$) of $\beta \land \neg \beta$ from $\Gamma$. This deduction contains only finitely many of the new constant symbols. By the theorem for generalization on constants, each can be replaced by a variable. We then have a deduction (in $L$) of $(\beta' \land \neg \beta')$ from $\Gamma$ (Do you know what $\beta'$ is?). This contradicts that $\Gamma$ is consistent.
Step 2: For each wff \( \varphi \) (in \( L_C \)) and each variable \( x \), we want to add to \( \Gamma \) the wff

\[ \neg \forall x \varphi \rightarrow \neg \varphi^x_c, \]

where \( c \) is one of the new constant symbols. The idea is that \( c \) would be a counterexample to \( \varphi \), if any. We can do this in such a way that \( \Gamma \) together with the set \( \Theta \) of all the added wffs is still consistent.

Details: Adopt a fixed enumeration of the pair \( \langle \varphi, x \rangle \), where \( \varphi \) is a wff (of \( L_C \)) and \( x \) is a variable:

\( \langle \varphi_1, x_1 \rangle, \langle \varphi_2, x_2 \rangle, \ldots \)

This is possible because the language is countable. Let \( \theta_1 \) be

\[ \neg \forall x_1 \varphi_1 \rightarrow \neg \varphi_1^{x_1} c_1, \]

where \( c_1 \) is the first of the new constant symbols not occurring in \( \varphi_1 \). Keep doing this we will get a set \( \Theta = \{ \theta_1, \theta_2, \ldots \} \). We claim that \( \Gamma \cup \Theta \) is consistent. If not, then (because deductions are finite) for some \( m \geq 0, \)

\[ \Gamma \cup \{ \theta_1, \ldots, \theta_{m+1} \} \]

is inconsistent. Take the least such \( m \). Then by RAA

\[ \Gamma \cup \{ \theta_1, \ldots, \theta_m \} \vdash \neg \theta_{m+1}. \]

Now \( \theta_{m+1} \) is

\[ \neg \forall x \varphi \rightarrow \neg \varphi^x_c \]

for some \( x, \varphi \) and \( c \). So by rule T, we obtain that \( \text{LHS} \vdash \neg \forall x \varphi \), and \( \text{LHS} \vdash \varphi^x_c \). Since \( c \) does not appear on \( \text{LHS} \), we have \( \text{LHS} \vdash \forall x \varphi \), contradicting the minimality of \( m \).

Step 3: We now extend the consistent set \( \Gamma \cup \Theta \) to a consistent set \( \Delta \) which is maximal in the sense that for any wff \( \varphi \) either \( \varphi \in \Delta \) or \( \neg \varphi \in \Delta \).

Details: See the proof Compactness Theorem in sentential logic. Notice also \( \Delta \) is deductively closed, because if \( \Delta \vdash \varphi \), then \( \Delta \not\vdash \neg \varphi \) by consistency, then \( \neg \varphi \not\in \Delta \), so \( \varphi \in \Delta \) by maximality.

So far we extended our language and added a set \( \Theta \), (sometimes called Henkin Axioms), and extended \( \Gamma \cup \Theta \) to a maximal consistent set \( \Delta \), which is automatically deductively closed.
Step 4. We now make from $\Delta$ a structure $\mathfrak{A}$ for the new language, but with \( \approx \) replaced by a new two-place predicate symbol $E$. We define $\mathfrak{A}$ as follows.

(a) $|\mathfrak{A}|$ = the set of all terms of $L_\mathcal{C}$.

(b) Define the binary relation $E^\mathfrak{A}$ by $\langle u, t \rangle \in E^\mathfrak{A}$ iff the formula $u \approx t$ is in $\Delta$.

(c) For each $n$-place predicate parameter $P$, define the $n$-ary relation $P^\mathfrak{A}$ by

\[
\langle t_1, \ldots, t_n \rangle \in P^\mathfrak{A} \iff Pt_1 \ldots t_n \in \Delta.
\]

(d) For each $n$-place function symbol $f$, let $f^\mathfrak{A}$ be the function defined by

\[
f^\mathfrak{A}(t_1, \ldots, t_n) = ft_1 \ldots t_n.
\]

(e) For a constant $c$, we take $c^\mathfrak{A} = c$.

Also define a function $s : V \to |\mathfrak{A}|$, namely the identity function. We claim that: for any term $t$, $s(t) = t$; for any formula $\varphi \models \varphi^*[s]$ iff $\varphi \in \Delta$, where $\varphi^*$ is obtained by replacing the equality symbol in $\varphi$ by $E$.

We now prove the claim. An easy induction on $t$ shows that $s(t) = t$. (Can you prove it?). We prove the second statement by induction on the number of connectives and quantifiers in $\varphi$.

Case 1: Atomic formula. True by definition. (Can you provide details?)

Case 2: Negation. Routine. (Can you prove it?)

Case 3: Suppose the claim is true for $\varphi$ and $\psi$, we want to show it is true for $\varphi \to \psi$.

\[
\models \mathfrak{A} (\varphi \to \psi)^*[s] \iff \not\models \mathfrak{A} \varphi^*[s] \text{ or } \models \mathfrak{A} \psi^*[s],
\]

\[
\iff \varphi \notin \Delta \text{ or } \psi \in \Delta,
\]

\[
\iff \neg \varphi \in \Delta \text{ or } \psi \in \Delta,
\]

\[
\Rightarrow \Delta \vdash (\varphi \to \psi)
\]

\[
\Rightarrow \varphi \notin \Delta \text{ or } [\varphi \in \Delta \text{ and } \Delta \vdash \psi]
\]

\[
\Rightarrow \neg \varphi \in \Delta \text{ or } \psi \in \Delta.
\]

which closes the loop.
Case 4: Suppose that the claim is true for $\varphi$, we show that $\models_\mathfrak{A} \forall x \varphi^*[s]$ iff $\forall x \varphi \in \Delta$ (Notice that $\forall x \varphi^*$ is $(\forall x \varphi)^*$).

$$\models_\mathfrak{A} \forall x \varphi^*[s] \Rightarrow \models_\mathfrak{A} \varphi^*[s(x|c)]$$
$$\Rightarrow \models_\mathfrak{A} (\varphi^*)^c_x[s] \quad \text{by the substitution lemma}$$
$$\Rightarrow \models_\mathfrak{A} (\varphi^*_c)^*[s]$$
$$\Rightarrow \varphi^*_c \in \Delta$$
$$\Rightarrow \neg \varphi^*_c \not\in \Delta$$
$$\Rightarrow (\neg \forall x \varphi) \not\in \Delta \quad \text{since } \theta \in \Delta \text{ and } \Delta \text{ is deductively closed}$$
$$\Rightarrow \forall x \varphi \in \Delta.$$ 

On the other hand, pick an alphabetic variant $\psi$ of $\varphi$ in which $t$ is substitutable, ($t$ is chosen as below) we have

$$\not\models_\mathfrak{A} \forall x \varphi^*[s] \Rightarrow \not\models_\mathfrak{A} \varphi^*[s(x|t)] \quad \text{for some } t, \text{ henceforth fixed}$$
$$\Rightarrow \not\models_\mathfrak{A} \psi^*[s(x|t)] \quad \text{by semantical equivalence of } \varphi^* \text{ and } \psi^* \text{ (Why?)}$$
$$\Rightarrow \not\models_\mathfrak{A} (\psi^*_t)^*[s] \quad \text{by the substitution lemma}$$
$$\Rightarrow \psi^*_t \not\in \Delta$$
$$\Rightarrow \forall x \psi \not\in \Delta$$
$$\Rightarrow \forall x \varphi \not\in \Delta \quad \text{by the syntactical equivalence of } \varphi \text{ and } \psi.$$ 

This establishes the claim.

So far we got a structure $\mathfrak{A}$, which satisfies all formulas in $\Delta$ not involving $\approx$. Assume now that $\approx$ is in the language, then $\mathfrak{A}$ may not work. For example, if $\Gamma$ contains the sentence $c \approx d$ (where $c$ and $d$ are distinct constant symbols), then we need a structure $\mathfrak{B}$ in which $c^\mathfrak{B} = d^\mathfrak{B}$. We obtain $\mathfrak{B}$ as the quotient structure $\mathfrak{A}/E$ of $\mathfrak{A}$ modulo $E^\mathfrak{A}$. For simplicity, we only consider unary predicates and functions, the general $n$-place ones are similar.

Step 5: $E^\mathfrak{A}$ is an equivalence relation on $|\mathfrak{A}|$. For each $t$ in $|\mathfrak{A}|$, let $[t]$ be its equivalence class. $E^\mathfrak{A}$ is a congruence relation for $\mathfrak{A}$, which means:

1. $E^\mathfrak{A}$ ia an equivalence relation on $|\mathfrak{A}|$.
2. $P^\mathfrak{A}$ is compatible with $E^\mathfrak{A}$ for each predicate symbol $P$:
   $$t \in P^\mathfrak{A} \text{ and } tE^\mathfrak{A} t' \Rightarrow t' \in P^\mathfrak{A}.$$
(3) $f^\mathfrak{A}$ is compatible with $E^\mathfrak{A}$ for each function symbol $f$:

$$tE^\mathfrak{A} t' \Rightarrow f^\mathfrak{A}(t) E^\mathfrak{A} f^\mathfrak{A}(t').$$

The verification is based on lemma about $\approx$ (Can you prove it?). We define the quotient structure $\mathfrak{A}/E$ as follows:

(a) $|\mathfrak{A}/E|$ is the set of all equivalence classes of members of $|\mathfrak{A}|$.

(b) For each predicate symbol $P$,

$$[t] \in P^\mathfrak{A}/E \iff t \in P^\mathfrak{A}.$$

(c) For each function symbol $f$,

$$f^\mathfrak{A}/E([t]) = [f^\mathfrak{A}(t)].$$

(d) For each constant $c$, $c^\mathfrak{A}/E = [c^\mathfrak{A}]$.

Let $h : |\mathfrak{A}| \rightarrow |\mathfrak{A}/E|$ be the natural map: $h(t) = [t]$. Then $h$ is a homomorphism of $\mathfrak{A}$ onto $\mathfrak{A}/E$ (Can you prove it?). And

$$[t]E^\mathfrak{A}/E [t'] \iff tE^\mathfrak{A} t' \iff [t] = [t'].$$

Finally,

$$\varphi \in \Delta \iff \models^\mathfrak{A} \varphi^s[s] \iff \models^\mathfrak{A}/E \varphi^s[h \circ s] \iff \models^\mathfrak{A}/E \varphi[h \circ s],$$

the last step being justified by the fact that $E^\mathfrak{A}/E$ is the equality relation on $|\mathfrak{A}/E|$.

**Step 6:** Restrict the structure $\mathfrak{A}/E$ to the original language. This restriction of $\mathfrak{A}/E$ satisfies every member of $\Gamma$ with $h \circ s$.

**Theorem 20 (Compactness Theorem)** (a) If $\Gamma \models \varphi$, then for some finite $\Gamma_0 \subseteq \Gamma$ we have $\Gamma_0 \models \varphi$.

(b) If every finite subset $\Gamma_0$ of $\Gamma$ is satisfiable, then $\Gamma$ is satisfiable.

**Proof.** Homework. □

36
We now look at some applications of Compactness Theorem.

**Theorem 21**  If a set $\Sigma$ of sentences has arbitrarily large finite models, then it has an infinite model.

**Proof.** For each integer $k \geq 2$, we can find a sentence $\lambda_k$ which translates, "There are at least $k$ things." For example,

$$
\lambda_2 = \exists v_1 \exists v_2 \: v_1 \neq v_2,
\lambda_3 = \exists v_1 \exists v_2 \exists v_3 \: (v_1 \neq v_2 \land v_2 \neq v_3 \land v_1 \neq v_3).
$$

Consider the set $\Sigma \cup \{\lambda_2, \lambda_3, \ldots \}$. By hypothesis any finite subset has a model. So by compactness the entire set has a model, which is infinite. $\square$

**Corollary 13** The class of all finite structures (for a fixed language) is not $EC_\Delta$. The class of all infinite structures is not $EC$.

Next we look at some nonstandard model of arithmetic.

**Example.** Consider the structure

$$
\mathfrak{A} = (\mathbb{N}, 0, S, <, +, \cdot).
$$

We claim that there is a countable structure $\mathfrak{B}$ elementarily equivalent to, but not isomorphic to, the structure $\mathfrak{A}$.

**Proof.** Expand the language by adding a new constant symbol $c$. Let

$$
\Sigma = \{0 < c, S0 < c, SS0 < c, \ldots \},
$$

and $\text{Th} \, \mathfrak{A}$, the theory of $\mathfrak{A}$, be the set of all sentences true in $\mathfrak{A}$, that is,

$$
\text{Th} \, \mathfrak{A} = \{ \sigma : \models_\mathfrak{A} \sigma \}.
$$

We claim that $\Sigma \cup \text{Th} \, \mathfrak{A}$ has a model. For any finite subset $\Sigma_0$ of $\Sigma \cup \text{Th} \, \mathfrak{A}$, $\Sigma_0$ is true in

$$
\mathfrak{A}_k = (\mathbb{N}, 0, S, <, +, \cdot, k)
$$

where $k = c^{\mathfrak{A}_k}$ for some large $k$. So by Compactness Theorem $\Sigma \cup \text{Th} \, \mathfrak{A}$ has a model, we can make it a countable model $\mathfrak{B}'$ (why?). Let $\mathfrak{B}$ be the restriction of $\mathfrak{B}'$ to the original language. Since $\mathfrak{B}$ is a model of $\text{Th} \, \mathfrak{A}$, we have $\mathfrak{A} \equiv \mathfrak{B}$. Moreover $\mathfrak{A}$ is not isomorphic to $\mathfrak{B}$, (can you prove it?). $\square$
Prenex Normal Form

Sometimes it is convenient to move all the quantifier symbols to the left of other symbols. Define a prenex formula to be one of the form (for some \( n \geq 0 \))

\[ Q_1 x_1 \ldots Q_n x_n \alpha, \]

where \( Q_i \) is \( \forall \) or \( \exists \) and \( \alpha \) is quantifier-free.

**Theorem 22 (Prenex Normal Form Theorem)** For any formula we can find a logically equivalent prenex formula.

**Proof.** We will make use of the following quantifier manipulation rules.

Q1a. \( \neg \forall x \alpha \models = \exists x \neg \alpha. \)

Q1b. \( \neg \exists x \alpha \models = \forall x \neg \alpha. \)

Q2a. \( (\alpha \rightarrow \forall x \beta) \models = \forall x (\alpha \rightarrow \beta) \) for \( x \) not free in \( \alpha. \)

Q2b. \( (\alpha \rightarrow \exists x \beta) \models = \exists x (\alpha \rightarrow \beta) \) for \( x \) not free in \( \alpha. \)

Q3a. \( (\forall x \alpha \rightarrow \beta) \models = \exists x (\alpha \rightarrow \beta) \) for \( x \) not free in \( \beta. \)

Q3b. \( (\exists x \alpha \rightarrow \beta) \models = \forall x (\alpha \rightarrow \beta) \) for \( x \) not free in \( \beta. \)

We now show by induction that every formula has an equivalent prenex formula.

1. For atomic \( \alpha \), it is already a prenex formula.

2. If \( \alpha \) is equivalent to the prenex \( \alpha' \), then \( \forall x \alpha \) is equivalent to the prenex \( \forall x \alpha' \).

3. If \( \alpha \) is equivalent to the prenex \( \alpha' \), then \( \neg \alpha \) is equivalent to \( \neg \alpha' \). Apply Q1 to \( \neg \alpha' \) to obtain a prenex formula.
4. Consider a formula of the form $\alpha \rightarrow \beta$. By induction hypothesis we have prenex formulas $\alpha'$ and $\beta'$ equivalent to $\alpha$ and $\beta$, respectively. By choosing suitable alphabetic variants, we may further assume that any variable which occurs quantified in one of the formulas $\alpha'$ and $\beta'$ does not occur at all in the other. We then use Q2 and Q3 to obtain a prenex formula equivalent to $\alpha' \rightarrow \beta'$ (and hence $\alpha \rightarrow \beta$).

\[\square\]

3.5 Models of Theories

Let us recall some basic facts about cardinals. Because we do not want to get into set theoretic details, our statements are informal. We say that two set $A$ and $B$ are equinumerous iff there is a one-to-one function mapping $A$ onto $B$. To each set $A$ we can assign a certain object, the cardinal number (or cardinality) of $A$. The smallest cardinals are those of finite set $0, 1, 2, \ldots$. The smallest infinite cardinal, which is the cardinality of $\mathbb{N}$, is called $\aleph_0$. The smallest cardinal larger than $\aleph_0$ is called $\aleph_1$. The cardinality of the real numbers $\mathbb{R}$ is called $2^{\aleph_0}$. Notice that $\aleph_0 < 2^{\aleph_0}$.

The operations of addition and multiplication can be extended from finite cardinals to all cardinals. To compute $\kappa + \lambda$, we choose disjoint set $A$ and $B$ of cardinality $\kappa$ and $\lambda$ respectively, then $\kappa + \lambda$ is the cardinality of $A \cup B$. And $\kappa \cdot \lambda$ is the cardinality of $A \times B$.

**Theorem 23 (Cardinal Arithmetic Theorem)** For cardinals $\kappa$ and $\lambda$, if $\kappa \leq \lambda$ and $\lambda$ is infinite, then $\kappa + \lambda = \lambda$. Furthermore, if $\kappa \neq 0$, then $\kappa \cdot \lambda = \lambda$.

**Theorem 24** For an infinite set $A$, the set $\bigcup_n A^n$ of all finite sequences of $A$ has cardinality equal to cardinality of $A$.

Now we go back to the topic on size of models.

**Theorem 25 (Downward Löwenheim–Skolem Theorem)** (a) Let $\Gamma$ be a satisfiable set of formulas in a countable language. Then $\Gamma$ is satisfiable in some countable structure.
(b) Let \( \Gamma \) be a satisfiable set of formulas in a language of cardinality \( \kappa \). Then \( \Gamma \) is satisfiable in some structure of cardinality \( \leq \kappa \).

**Proof.** We only prove (b) because (a) is just a special case of (b). Fix \( \Gamma \), by Soundness Theorem, \( \Gamma \) is consistent. We follow the steps in the proof of Completeness Theorem to form to structure \( \mathfrak{A}/E \) in which \( \Gamma \) was satisfied. The universe of \( \mathfrak{A} \) was the set of all terms in the language obtained by adding \( \kappa \) new constants. So \( |\mathfrak{A}| \) contains at least \( \kappa \) terms. On the other hand, there are only \( \kappa \) expressions in the augmented language, so \( |\mathfrak{A}| \) could have no more than \( \kappa \) terms. Thus the cardinality of \( \mathfrak{A} \) (by which we mean the cardinality of \( |\mathfrak{A}| \)) was \( \kappa \). The universe of \( \mathfrak{A}/E \) consists of equivalence classes of members of \( \mathfrak{A} \), so it has cardinality \( \leq \kappa \). \( \square \)

We have defined the **theory** of \( \mathfrak{A} \) \( \text{Th} \ \mathfrak{A} \), for a structure \( \mathfrak{A} \), to be the set of all sentences true in \( \mathfrak{A} \). It is easy to see that if \( \mathfrak{B} \) is a model of \( \text{Th} \ \mathfrak{A} \), then \( \mathfrak{A} \equiv \mathfrak{B} \) (Why?).

Now suppose that we have an uncountable structure \( \mathfrak{A} \) for a countable language. By previous theorem (applied to \( \text{Th} \ \mathfrak{A} \)) there is a countable model \( \mathfrak{B} \) which is a model of \( \text{Th} \ \mathfrak{A} \). Hence \( \mathfrak{A} \equiv \mathfrak{B} \). Conversely, suppose that we start with a countable structure \( \mathfrak{B} \). Is there an uncountable \( \mathfrak{A} \) such that \( \mathfrak{A} \equiv \mathfrak{B} \)? If \( \mathfrak{B} \) is finite (and the language includes equality), then this is impossible (Why?). But if \( \mathfrak{B} \) is infinite, then there will be such an \( \mathfrak{A} \), as stated in the following theorem.

**Theorem 26 (Upward Löwenheim-Skolem Theorem)** Let \( \mathfrak{A} \) be a set of formulas in a language of cardinality \( \kappa \), and assume that \( \mathfrak{A} \) is satisfiable in some infinite structure. Then for every cardinal \( \lambda \geq \kappa \), there is a structure of cardinality \( \lambda \) in which \( \Gamma \) is satisfiable.

**Proof.** Let \( \mathfrak{A} \) be the infinite structure in which \( \Gamma \) is satisfiable. Expand the language by adding a set \( C \) of \( \lambda \) new constants. Let

\[
\Sigma = \{ c_1 \neq c_2 : c_1, c_2 \text{ distinct members of } C \}.
\]

Then every finite subset of \( \Sigma \cup \Gamma \) is satisfiable, a slight change of \( \mathfrak{A} \) will provide a model (How?). So by compactness \( \Sigma \cup \Gamma \) is satisfiable, and by the
downward Löwenheim-Skolem Theorem, it is satisfiable in a structure $\mathfrak{B}$ of cardinality $\leq \lambda$, because the expanded language has cardinality $\kappa + \lambda = \lambda$. But any model of $\Sigma$ clearly has cardinality $\geq \lambda$. So $\mathfrak{B}$ has cardinality $\lambda$, the restriction of $\mathfrak{B}$ to the original language is what we want. \hfill \Box

Informally, an algorithm $P$ is a finite sequence of concrete instructions, such that for any input $x$, if we follow the instructions in $P$, then after finitely many steps, $P$ gives us an output.

Fix a set $X$, a subset $A$ of $X$ is decidable if there is an algorithm $P$, such that for any $x \in X$, $P$ outputs 1 if $x \in A$ and $P$ outputs 0 otherwise. For example, the set of sentential tautologies is decidable (why?).

Next we define a theory to be a set of sentences closed under logical implication, that is for any sentence $\sigma$ of the language, if $T \models \sigma$ then $\sigma \in T$. For example, the set of all valid sentences of the language is a theory.

For any structure $\mathfrak{A}$, Th $\mathfrak{A}$ is a theory. More generally, for a class $K$ of structures, define the theory of $K$, Th $K$, to be the set of sentences true in every member of $K$. We can verify that Th $K$ is indeed a theory.

A theory $T$ is complete iff for every sentence $\sigma$, either $\sigma \in T$ or $\neg \sigma \in T$. For example, Th $\mathfrak{A}$ is always complete.

**Lemma 7** Th $K$ is complete iff any two members of $K$ are elementarily equivalent. And a theory $T$ is complete iff any two models of $T$ are elementarily equivalent.

For example, the theory of fields is not complete, since we have fields of different characteristics.

**Definition 9** A theory $T$ is axiomatizable iff there is a decidable set $\Sigma$ of sentences such that $T$ is the set of consequences of $\Sigma$. If $\Sigma$ is finite, then we say $T$ is finitely axiomatizable.

For example, the theory of fields of characteristic 0 is axiomatizable, but not finitely axiomatizable (Why?).

**Lemma 8** A complete axiomatizable theory is decidable.
Theorem 27 (Los-Vaught Test) Let $T$ be a theory in a countable language such that

1. $T$ is $\lambda$-categorical for some infinite cardinal $\lambda$, that is, all models of $T$ having cardinality $\lambda$ are isomorphic.

2. All models of $T$ are infinite.

Then $T$ is complete.

Let $\Delta$ be the set of axioms for dense linear orderings, and $T$ be the set of all consequences of $\Delta$.

Lemma 9 Any countable model of $\Delta$ is isomorphic to $(\mathbb{Q}, <_{\mathbb{Q}})$, in other words, $T$ is $\aleph_0$-categorical.

Thus we can apply Los-Vaught test to conclude that $T$ is complete. Hence any two models of $T$ are elementarily equivalent; in particular, 

$$(\mathbb{Q}, <_{\mathbb{Q}}) \equiv (\mathbb{R}, <_{\mathbb{R}}).$$

We can also conclude that $T$ is also decidable.

Lemma 10 (Steinitz Theorem) Two algebraically closed fields are isomorphic iff they have the same characteristic and the same transcendence degree.

From Steinitz Theorem, we know that two algebraically closed fields of characteristic 0 is $\lambda$-categorical for any uncountable cardinal $\lambda$. Apply Los-Vaught test, we have

Theorem 28 The theory of algebraically closed fields of characteristic 0 is complete and decidable.