Mathematical Logic

Thomas Walker

Autumn 2024

Contents

1 Propositional Logic

1.1 Propositional Formulas

A proposition is a true or false statement. Propositional logic is a language to deal with propositions.

Definition 1.1.1. *Propositional logic is a language that consists of*

1. *a non-empty set P of propositions called atomic propositions,*
2. *connectives $\wedge, \vee, \rightarrow, \leftrightarrow, \neg, \perp$ and*
3. *auxiliary symbols.*

The connectives operate on one or two propositions, say p and q, and so their function can be defined entirely by enumerating the possibilities in truth tables.

p	q	$p \wedge q$	$p \vee q$	$p \rightarrow q$	$p \leftrightarrow q$	$\neg p$
True	True	True	True	True	True	False
True	False	False	True	False	False	
False	True	False	True	True	False	
False	False	False	False	True	True	True

Table 1: Propositional Formulas

The connective $\perp$ represents a logical contradiction.

Definition 1.1.2. *For a set of atomic propositions, let $\mathcal{W}(P)$ to be the smallest set with the following properties.*

1. *$P \subseteq \mathcal{W}(P)$.*
2. *If $p, q \in \mathcal{W}(P)$, then $p \,\square\, q \in \mathcal{W}(P)$ for $\square \in \{\wedge, \vee, \rightarrow, \leftrightarrow\}$.*
3. *$p \in \mathcal{W}(P) \rightarrow (\neg p) \in \mathcal{W}(P)$.*
4. *$\perp \in \mathcal{W}(P)$.*

Therefore, $\mathcal{W}(P)$ represents the set of propositional formulas that can be constructed using the propositional logic defined by P. To construct proofs using propositional logic we use the method of induction. Suppose A is a property of elements in $\mathcal{W}(P)$. Then we can prove that $A(\phi)$ holds for all $\phi \in \mathcal{W}(P)$ by showing the following.

1. If $A(p)$ holds for all $p \in P$.
2. If $(A(\phi) \wedge A(\psi)) \rightarrow A(\phi \,\square\, \psi)$ for all $\square \in \{\wedge, \vee, \rightarrow, \leftrightarrow\}$.
3. If $A(\phi) \rightarrow A(\neg\phi)$.
4. $A(\perp)$.

Definition 1.1.3. *A sequence $\phi_1, \ldots, \phi_n \in \mathcal{W}(P)$ is a formation sequence of ϕ if the following statements are satisfied.*

1. *$\phi_n = \phi$.*
2. *For all i, either*
   (a) *ϕ_i is atomic,*
   (b) *$\phi_i = \phi_j \,\square\, \phi_k$ for $j, k < i$, or*

(c) $\phi_i = \neg\phi_j$ for $j < i$.

Theorem 1.1.4. *Given a set of atoms P, then $\mathcal{W}(P)$ is the set of expressions that have a formation sequence.*

1.2 Valuation Maps

From now on the truth value of $p \in P$ will be either 1 (True) or 0 (False).

Definition 1.2.1. *A map $v : \mathcal{W}(P) \to \{0,1\}$ is a valuation map if for all $\phi, \psi \in \mathcal{W}(P)$ the following statements hold.*

1. $v(\phi \wedge \psi) = \min(v(\phi), v(\psi))$.
2. $v(\phi \vee \psi) = \max(v(\phi), v(\psi))$.
3. $v(\phi \to \psi) = 0$ *if and only if* $v(\phi) = 1$ *and* $v(\psi)) = 0$.
4. $v(\phi \leftrightarrow \psi) = 1$ *if and only if* $v(\phi) = v(\psi)$.
5. $v(\neg\phi) = 1 - v(\phi)$.
6. $v(\bot) = 0$.

Lemma 1.2.2. *Suppose that v and v' are two valuation maps such that $v(p) = v'(p)$ for all $p \in P$. Then, $v(\phi) = v'(\phi)$ for all $\phi \in \mathcal{W}(P)$.*

Corollary 1.2.3. *If $v' : P \cup \{\bot\} \to \{0,1\}$ is such that $v'(\bot) = 0$, then there is a unique valuation map such that $v(p) = v'(p)$ for all $p \in P$.*

Definition 1.2.4. *Let P be a set of atoms, and let $\phi \in \mathcal{W}(P)$.*

1. *ϕ is a tautology, $\vDash \phi$, if $v(\phi) = 1$ for all valuations maps of $\mathcal{W}(P)$.*
2. *ϕ is a semantic consequence for $\Gamma \subseteq \mathcal{W}(P)$, $\Gamma \vDash \phi$, if for every valuation map v such that $v(\psi) = 1$ for all $\psi \in \Gamma$, we have that $v(\phi) = 1$.*

1.3 Substitution

Consider a finite set of atoms $P = \{p_1, \ldots, p_n\}$, and $\phi \in \mathcal{W}(P)$. We can define a substitution map as follows.

1. For $\phi \in P$ we let $\phi\lfloor\psi/p_i\rfloor = \begin{cases} \phi & \phi \neq p_i \\ \psi & \psi = p_i. \end{cases}$
2. For $\phi = \phi_1 \square \phi_2$ we let $\phi\lfloor\psi/p_i\rfloor = \phi_1\lfloor\psi/p_i \square \phi_2\lfloor\psi/p_i\rfloor$.
3. For $\phi = \neg\phi_1$ we let $\phi\lfloor\psi/p_i\rfloor = \neg\phi_1\lfloor\psi/p_i\rfloor$.

Theorem 1.3.1. *If $\vDash (\phi_1 \leftrightarrow \phi_2)$ then $\vDash (\psi\lfloor\phi_1/p\rfloor \leftrightarrow \psi\lfloor\phi_2/p\rfloor)$.*

The substitution map $\phi\lfloor\psi/p_i\rfloor$ replaces all instances of p_i in ϕ with ψ. Theorem 1.3.1 says that substituting atoms of the same truth value into the same proposition should change the truth value of the proposition in the same way.

Corollary 1.3.2. *For every $\phi \in \mathcal{W}(P)$ there is a $\psi \in \mathcal{W}(P)$ such that ψ only has connectives $\{\wedge, \vee, \neg\}$ such that $\vDash (\phi \leftrightarrow \psi)$.*

Let $\phi(p_1, \ldots, p_n) \in \mathcal{W}(P)$, where $n \leq |P|$. Then the truth function obtained from ϕ, $F_\phi : \{0,1\}^n \to \{0,1\}$ is such that for $\bar{x} = (x_1, \ldots, x_n)$ the value of $F_\phi(\bar{x})$ is the truth value of ϕ when $p_i = x_i$ for $i = 1, \ldots, n$. More generally, a truth function will refer to any $F : \{0,1\}^n \to \{0,1\}$.

Definition 1.3.3. *Formulas φ and ψ in n-variables are logically equivalent if $F_\varphi = F_\psi$. Equivalent, we say that φ and ψ are logically equivalent if $\vDash (\varphi \leftrightarrow \psi)$.*

Definition 1.3.4. *A set A of connectives is adequate if for every $1 \leq n \leq |P|$, and for every truth function, F, on n-variables there exists $\phi \in \mathcal{W}(P)$ that is only constructed using connectives from A and variables $p_1, \ldots, p_n$ such that $F = F_\phi$.*

Theorem 1.3.5. *The set $\{\wedge, \vee, \neg\}$ is adequate.*

Corollary 1.3.6. *The following sets are adequate.*

- $\{\neg, \vee\}$.
- $\{\neg, \wedge\}$.
- $\{\neg, \rightarrow\}$.

Definition 1.3.7. *We say that $\phi \in \mathcal{W}(P)$ is in disjunctive normal form if it is a disjunction ($\vee$'s) of conjunctions ($\wedge$'s) of literals (statements of negations of atoms).*

Corollary 1.3.8. *Suppose $\phi \in \mathcal{W}(P)$, then there is a $\psi \in \mathcal{W}(P)$ in disjunctive normal form such that $\vDash (\phi \leftrightarrow \psi)$.*

1.4 The Deductive Approach

To start making deductions we need to introduce some meaning to our logical statements. We can do this by considering a $\Sigma \subseteq \mathcal{W}(P)$ and derive the semantic consequences, $\Sigma \vDash \varphi$, using a valuation map. That is, we say that $\Sigma \nvDash \varphi$ if and only if there is a valuation map v such that $v(\psi) = 1$ for all $\psi \in \Sigma$ and $v(\varphi) = 0.\Sigma$ can be thought of as a set of assumptions, or axioms from which we can derive logical statements. We proceed agnostically to the nature of Σ and focus on the process of deduction. The idea is to start from some premises, φ or $\varphi \rightarrow \psi$, and then use the deduction rules to make conclusions,

$$\frac{\varphi \quad \varphi \rightarrow \psi}{\psi}.$$

The set $\{\rightarrow, \bot\}$ is adequate, so the superset $\{\rightarrow, \wedge, \bot\}$ is also adequate. In the following, we will use this superset as our set of connectives to construct our set of rules.

- Introduction Rules.
    - $\wedge$ introduction, is defined for premises φ and ψ as

$$\frac{\varphi \quad \psi}{\varphi \wedge \psi}(\wedge I).$$

– $\to$ introduction, is defined for premise φ and finitely many manipulations as

$$\begin{array}{c} \not\varphi \\ \vdots \\ \dfrac{\psi}{\varphi \to \psi}(\to I). \end{array}$$

- Elimination Rules.
  - $\wedge$ elimination, is defined for premise $\varphi \wedge \psi$ as

$$\frac{\varphi \wedge \psi}{\varphi}(\wedge E).$$

  - $\to$ elimination, is defined for premises φ and ψ as

$$\frac{\varphi \quad \varphi \to \psi}{\psi}(\to E).$$

- Other Rules.
  - Principle of explosion, is defined for premise $\perp$ as

$$\frac{\perp}{\varphi}(\perp).$$

  - Reductio ad absurdum, is defined for premise $\neg\varphi$ and finitely many manipulations as

$$\begin{array}{c} \not\neg \not\varphi \\ \vdots \\ \dfrac{\perp}{\varphi}\ (\text{RAA}). \end{array}$$

Definition 1.4.1. *Let $\Sigma \subseteq \mathcal{W}(P)$, then $\Sigma \vdash \varphi$ if there is $\varphi_1, \ldots, \varphi_n = \varphi$ such that*

1. *$\varphi_i \in \Sigma$, or*
2. *φ_i is obtained from a deduction rule.*

1.5 Soundness and Completeness Theorems

Theorem 1.5.1 (Soundness)**.** *Let $\Gamma \subseteq \mathcal{W}(P)$ and $\varphi \in \mathcal{W}(P)$, then $\Gamma \vdash \varphi$ implies that $\Gamma \vDash \varphi$.*

Soundness means that if a statement can be deduced from a set of assumptions, then it is the semantic consequence of those assumptions.

Theorem 1.5.2 (Completeness)**.** *Let $\Gamma \subseteq \mathcal{W}(P)$ and $\varphi \in \mathcal{W}(P)$, then $\Gamma \vDash \varphi$ if and only if $\Gamma \vdash \varphi$.*

Completeness means that in addition to being sound, if a statement is a semantic consequence of a set of assumptions, then it can be deduced from those assumptions.

1.6 Consistent and Maximally Consistent Sets of Formulae

Definition 1.6.1. *We say that* $\Gamma \subseteq \mathcal{W}(P)$ *is consistent if* $\Gamma \nvdash \bot$ *and inconsistent if* $\Gamma \vdash \bot$*.*

Lemma 1.6.2. *The following are equivalent.*

1. Γ *is consistent.*
2. *For no* $\phi \in \mathcal{W}(P)$ *does both* $\Gamma \vdash \phi$ *and* $\Gamma \vdash \neg\phi$ *hold.*
3. $\Gamma \nvdash \phi$ *for at least one* $\phi \in \mathcal{W}(P)$*.*

Lemma 1.6.3. *Suppose* v *is a valuation function and* $\Gamma \subseteq \mathcal{W}(P)$ *such that* $v(\psi) = 1$ *for all* $\psi \in \Gamma$*. Then* Γ *is consistent.*

Lemma 1.6.4. *Suppose that* Γ *is a consistent set of formulae.*

1. *If* $\Gamma \cup \{\neg\varphi\} \vdash \bot$ *then* $\Gamma \vdash \varphi$*.*
2. *If* $\Gamma \cup \{\varphi\} \vdash \bot$ *then* $\Gamma \vdash \neg\varphi$*.*

Definition 1.6.5. Γ *is called maximally consistent when*

- Γ *is consistent and*
- *if* $\Gamma' \supseteq \Gamma$ *such that* Γ' *is consistent, then* $\Gamma' = \Gamma$*.*

Lemma 1.6.6. *Let* Γ *be maximally consistent.*

1. *For all formulae* ϕ *either* $\phi \in \Gamma$ *or* $\neg\phi \in \Gamma$*.*
2. *For all formulae* ϕ, ψ *we have that* $(\phi \to \psi) \in \Gamma$ *if and only if* $\phi \in \Gamma$ *implies* $\psi \in \Gamma$*.*

Corollary 1.6.7. *If* Γ *is maximally consistent, then*

1. $\phi \in \Gamma$ *if and only if* $\neg\phi \notin \Gamma$*, and*
2. $\neg\phi \in \Gamma$ *if and only if* $\phi \notin \Gamma$*.*

Lemma 1.6.8 (Lindenbaum)**.** *For each* $\Gamma \subseteq \mathcal{W}(P)$ *that is consistent, there is a maximally consistent* $\Gamma^* : \Gamma \subseteq \Gamma^* \subset \mathcal{W}(P)$*.*

Lemma 1.6.9. *Suppose that* Γ *is consistent. Then there is a valuation function* v *such that* $v(\theta) = 1$ *for all* $\theta \in \Gamma$*.*

Corollary 1.6.10. *If* $\Gamma \nvdash \phi$ *then there is a valuation function* v *such that* $v(\theta) = 1$ *for all* $\theta \in \Gamma$ *but* $v(\phi) = 0$*.*

2 Predicate Logic

2.1 First Order Structures, Languages and Formulas

Definition 2.1.1. *Suppose A is a set and $n \in \mathbb{Z}^+$.*

- *A n-ary relation on A is a subset $R \subseteq A^n$.*
- *A n-ary function of A is a well-defined map $f : A^n \to A$.*

Definition 2.1.2. *A first-order structure $\mathcal{A}$ has the following components.*

1. *A non-empty set A called the domain of $\mathcal{A}$,*
2. *A set of relations on A, given by*
$$\left\{R_i \subseteq A^{n_i} : i \in l, n_i \in \mathbb{Z}^+\right\}.$$
3. *A set of functions on A, given by*
$$\left\{f_j : A^{n_j} \to A : j \in J, n_j \in \mathbb{Z}^+\right\}.$$
4. *A set of constants, given by*
$$\{c_k \in A : k \in K\}.$$

A first-order structure can be summarised as $\mathcal{A} = \langle A, (R_i : i \in l), (f_j : j \in J), (c_k : k \in K)\rangle$, with $(n_i : i \in l), (n_j : j \in J), K$ being known as the signature of $\mathcal{A}$.

Definition 2.1.3. *A first order language L with signature $(n_i : i \in l), (n_j : j \in J), K$ has the following components.*

1. *A countable set of variables,*
$$\mathcal{V} = \{v_1, v_2, \dots\}.$$
2. *A set of connectives,*
$$\{\wedge, \vee, \neg, \to, \leftrightarrow, \bot, \forall, \exists\}.$$
3. *Auxiliary symbols.*
4. *A set of relations,*
$$\mathcal{R} = \{R_i : i \in I\}.$$
5. *A set of functions,*
$$\mathcal{F} = \{f_j : j \in J\}.$$
6. *A set of constants,*
$$\mathcal{C} = \{c_k : k \in K\}.$$
7. *An equality symbol, $\doteq$.*

For propositional logic.

- The set of variables was given by the set of atoms, P.
- The set of connectives were given by $\{\wedge, \vee, \to, \leftrightarrow, \neg, \bot\}$.
- The auxiliary symbols constituted $\{, \}$.

Let $E = P \cup \{\wedge, \vee, \to, \leftrightarrow, \neg, \bot\} \cup \{,\}$ and consider the set $S_f(E)$ of all finite sequences of elements of E. Note that $E^* = \bigcup_{n \in \mathbb{Z}_{\geq 0}} E^n = S_f(E)$ and $\mathcal{W}(P) \subsetneq S_f(E)$.

Definition 2.1.4. *The set of L-terms,* $\mathrm{Term}(L)$*, is the smallest subset of $S_f(L)$ with the following properties.*

1. $\mathcal{V} \subseteq \mathrm{Term}(L)$.
2. $\mathcal{C} \subseteq \mathrm{Term}(L)$.
3. *For a n_f-array function $f \in \mathcal{F}$ and $t_1, \ldots, t_{n_f} \in \mathrm{Term}(L)$, then $f(t_1, \ldots, t_n) \in \mathrm{Term}(L)$.*

Definition 2.1.5. *The set of L-formulas,* $\mathrm{Form}(L)$*, is the smallest subset of $S_f(L)$ with the following properties.*

1. $\bot \in \mathrm{Form}(L)$.
2. $t_1, \ldots, t_{n_R} \in \mathrm{Term}(L)$ *implies that* $R(t_1, \ldots, t_{n_R}) \in \mathrm{Form}(L)$.
3. $t_1, t_2 \in \mathrm{Term}(L)$ *implies that* $t_1 \doteq t_2 \in \mathrm{Form}(L)$.
4. $\phi, \psi \in \mathrm{Form}(L)$ *implies that* $\phi \,\Box\, \psi \in \mathrm{Form}(L)$ *for any connective* $\Box$.
5. $\phi \in \mathrm{Form}(L)$ *implies that* $\neg\phi \in \mathrm{Form}(L)$.
6. $\phi \in \mathrm{Form}(L)$ *implies that* $\forall v \phi \in \mathrm{Form}(L)$ *and* $\exists v \phi \in \mathrm{Form}(L)$.

The connectives $\exists$ and $\vee$ from predicate logic can be interpreted in propositional logic.

- $(\exists x)\phi$ means $(\neg(\forall x)(\neg\phi))$.
- $\phi \vee \psi$ means $((\neg\phi) \to \psi)$.

2.2 Interpretations of Languages and Assignments

Definition 2.2.1. *We say that $\phi \in \mathrm{Form}(L)$ is atomic if it is of the form $R_i(t_1, \ldots, t_{n_i})$ for $R_i \in \mathcal{R}$ and $t_1, \ldots, t_{n_i} \in \mathrm{Term}(L)$.*

Definition 2.2.2. *Let L be a first-order language with*

- $\mathcal{R} = \{R_i : i \in l\}$,
- $\mathcal{F} = \{f_j : j \in J\}$, *and*
- $\mathcal{C} = \{c_k : k \in K\}$.

Then an L-structure, $\mathcal{A}$, is a first order structure with signature, $\mathcal{R}, \mathcal{F}, K$. To specify a language we write

$$L = \mathcal{R} \cup \mathcal{F} \cup \mathcal{C} \cup \{\text{logical symbols and variables}\}.$$

If $\mathcal{A}$ is a corresponding L structure,

- the interpretation of $R_i \in \mathcal{R}$ in $\mathcal{A}$ is denoted $R_i^{\mathcal{A}}$,
- the interpretation of $f_j \in \mathcal{F}$ in $\mathcal{A}$ is denoted $f_j^{\mathcal{A}}$, and
- the interpretation of $c_k \in \mathcal{C}$ in $\mathcal{A}$ is denoted $c_k^{\mathcal{A}}$.

Definition 2.2.3. *Let L be a first-order language and $\mathcal{A}$ a corresponding L-structure. An $\mathcal{A}$-assignment is a function $\beta : V \to A$ and is extended to $\mathrm{Term}(L)$ so that the following hold.*

1. $t^{\mathcal{A}}[\beta] = \beta(v_i)$ *for* $t = v_i$,
2. $t^{\mathcal{A}}[\beta] = c_k^{\mathcal{A}}$ *if* $t = c_k$,
3. $t^{\mathcal{A}}[\beta] = f_j^{\mathcal{A}}\left(t_1^{\mathcal{A}}[\beta], \ldots, t_{m_i}^{\mathcal{A}}[\beta]\right)$ *if* $t = f_j\left(t_1, \ldots, t_{m_i}\right)$.

For β a $\mathcal{A}$-assignment $a \in A$ and $v \in V$ we can define a substitution map $\beta(a/v) : V \to A$ defined by

$$\beta(a/v)(v_i) = \begin{cases} \beta(v_i) & v_i \neq v \\ a & v_i = v. \end{cases}$$

Definition 2.2.4. *Let L be a language and $\mathcal{A}$ a corresponding L-structure. For an assignment function $\beta : \mathrm{Term}(L) \to \mathcal{A}$ we define a valuation map $v^{\beta} : \mathrm{Form}(L) \to \{0, 1\}$ inductively in the following way.*

1. $v^{\beta}(\bot) = 0$.
2. *For* $t_1, t_2 \in \mathrm{Term}(L)$ *such that* $t_1 \doteq t_2 \in \mathrm{Form}(L)$ *we have that*

$$v^{\beta}\left(t_1 \doteq t_2\right) = \begin{cases} 1 & t_1^{\mathcal{A}}[\beta] = t_2^{\mathcal{A}}[\beta] \\ 0 & \textit{otherwise.} \end{cases}$$

3. *For* $t_1, \ldots, t_{n_i} \in \mathrm{Term}(L)$ *and* $R_i(t_1, \ldots, t_{n_i})$ *we have that*

$$v^{\beta}\left(R_i(t_1, \ldots, t_{n_i})\right) = \begin{cases} 1 & \left(t_1^{\mathcal{A}}[\beta], \ldots, t_{n_i}^{\mathcal{A}}[\beta]\right) \in R_i^{\mathcal{A}} \\ 0 & \textit{otherwise.} \end{cases}$$

4. *For* $\phi, \psi \in \mathrm{Form}(L)$, *then* $v^{\beta}(\phi \,\square\, \psi)$ *is as defined in the case of propositional logic.*
5. *For* $\phi \in \mathrm{Form}(L)$, *we have that* $v^{\beta}(\neg\phi) = 1 - v^{\beta}(\phi)$.
6. *For* $\phi \in \mathrm{Form}(L)$ *we have that* $v^{\beta}\left(\forall w \phi\right) = \min\left\{v^{\beta(a/w)}(\phi) : a \in A\right\}$.
7. *For* $\phi \in \mathrm{Form}(L)$ *we have that* $v^{\beta}\left(\exists w \phi\right) = \max\left\{v^{\beta(a/w)}(\phi) : a \in A\right\}$.

Definition 2.2.5. *Let $v^{\beta} : \mathrm{Form}(L) \to \{0, 1\}$ be a valuation map as defined above.*

- $\mathcal{A} \vDash \phi[\beta]$ *if* $v^{\beta}(\phi) = 1$.
- $\mathcal{A} \vDash \phi$, *that is ϕ is satisfied by $\mathcal{A}$, if for all $\mathcal{A}$-assignments β we have that $\mathcal{A} \vDash \phi[\beta]$.*
- $\vDash \phi$, *that is ϕ is logically valid, if for all L-structures $\mathcal{A}$ we have that $\mathcal{A} \vDash \phi$.*

2.3 Variable Binding and Sub-formals

Definition 2.3.1. *Suppose ϕ and ψ are L-formulas.*

1. *If $\forall x_i \phi$ is a sub-formula of ψ. Then ϕ is the scope of the quantifier $\forall x_i$ in ψ.*
2. *A variable x_i in ψ is bounded if it is in the scope of a quantifier $\forall x_i$ in ψ.*
3. *If a variable is not in the scope of any quantifier, then the variable has free occurrence and is called a*

free variable.

Definition 2.3.2. *A formula $\phi \in \mathrm{Form}(L)$ is called a sentence if it has no free variables.*

Lemma 2.3.3. *Suppose that $\mathcal{A}$ is an L-structure and let $\phi, \psi \in \mathrm{Form}(L)$ be sentences. Then*

1. *$\mathcal{A} \vDash \phi \wedge \psi$ if and only if $\mathcal{A} \vDash \phi$ and $\mathcal{A} \vDash \psi$.*
2. *$\mathcal{A} \vDash \phi \vee \psi$ if and only if $\mathcal{A} \vDash \phi$ or $\mathcal{A} \vDash \psi$.*
3. *$\mathcal{A} \vDash \neg\phi$ if and only if $\mathcal{A} \nvDash \phi$.*
4. *$\mathcal{A} \vDash (\phi \to \psi)$ if and only if $\mathcal{A} \vDash \phi$ implies $\mathcal{A} \vDash \psi$.*

Now suppose $\phi \in \mathrm{Form}(L)$ only has the free variable $v \in V$, then

1. *$\mathcal{A} \vDash (\forall v\phi)$ if and only if for all $a \in A$ we have that $\mathcal{A} \vDash \phi[a]$.*
2. *$\mathcal{A} \vDash (\exists v\phi)$ if and only if there exists $a \in A$ such that $\mathcal{A} \vDash \phi[a]$.*

Lemma 2.3.4. *Suppose $\gamma, \beta : V \to A$ are $\mathcal{A}$-assignments which agree on $v_1, \ldots, v_n$. Then for $t(v_1, \ldots, v_n) \in \mathrm{Term}(L)$ we have that*

$$t^{\mathcal{A}}[\gamma] = t^{\mathcal{A}}[\beta].$$

Lemma 2.3.5. *Suppose β and γ are two $\mathcal{A}$-assignments which agree in $v_1, \ldots, v_n$. For $\phi(v_1, \ldots, v_n) \in \mathrm{Form}(L)$ we have that*

$$v^{\beta}(\phi) = v^{\gamma}(\phi).$$

Note that we say $A \vDash \phi[a_1, \ldots, a_n]$ when $\mathcal{A} \vDash \phi[\beta]$ and $\beta(v_i) = a_i$ for all $1 \le i \le n$.

Corollary 2.3.6. *Suppose ϕ is a sentence, then $\mathcal{A} \vDash \phi$ or $\mathcal{A} \vDash \neg\phi$.*

Definition 2.3.7. *For an L-structure $\mathcal{A}$ we denote the set*

$$\mathrm{Con}(\mathcal{A}) = \{\phi \in \mathrm{Form}(L) : \phi \text{ a sentence such that } \mathcal{A} \vDash \phi\}.$$

2.4 Substitution

For a non-empty set P, let $\theta \in \mathcal{W}(P)$ be a tautology be such that $p_1, \ldots, p_n$ appear in θ. Let $\phi_1, \ldots, \phi_n \in \mathrm{Form}(L)$, and let ψ be the formula obtained by substituting the p_i in θ with ϕ_i. It can be shown that $\psi \in \mathrm{Form}(L)$ is a logically valid formula. For β a $\mathcal{A}$-assignment we consider $\bar{x} := \big(v^{\beta}(\phi_1), \ldots, v^{\beta}(\phi_n)\big)$ as an element of $\{0,1\}^n$ so that $v^{\beta}(\psi) = F_{\theta}(\bar{x})$.

Definition 2.4.1. *Suppose $t, s \in \mathrm{Term}(L)$ and $v \in V$. Then we define the operator, $s\lfloor t/v \rfloor$, on $\mathrm{Term}(L)$ as follows.*

- *If $s = v_i$, then*

$$s\lfloor t/v \rfloor = \begin{cases} t & v_i = v \\ v_i & v_i \neq v. \end{cases}$$

- *If* $s = c$, *then*

$$s\lfloor t/v \rfloor = c.$$

- *If* $s = f(t_1, \ldots, t_{m_j})$, *then*

$$s\lfloor t/v \rfloor = f\left(t_1\lfloor t/v \rfloor, \ldots, t_{m_j}\lfloor t/v \rfloor\right).$$

Which extends to formulas $\phi \in \mathrm{Form}(L)$ *as follows.*

- $\perp \lfloor t/v \rfloor = \perp$.
- $R\,(t_1, \ldots, t_{n_i})(\lfloor t/v \rfloor) = R\,(t_1\lfloor t/v \rfloor, \ldots, t_{n_i}\lfloor t/v \rfloor)$.
- $(t_1 \doteq t_2)\lfloor t/v \rfloor = (t_1\lfloor t/v \rfloor \doteq t_2\lfloor t/v \rfloor)$.
- $(\phi \,\square\, \psi)\lfloor t/v \rfloor = \phi\lfloor t/v \rfloor \,\square\, \psi\lfloor t/v \rfloor$, *as in propositional logic.*
- $(\forall v_i \phi(v_i))\lfloor t/v \rfloor = \begin{cases} \forall v_i(\phi\lfloor t/v \rfloor) & v \neq v_i \\ \forall v_i \phi(v_i) & v = v_i. \end{cases}$
- $(\exists v_i \phi(v_i))\lfloor t/v \rfloor = \begin{cases} \exists v_i(\phi\lfloor t/v \rfloor) & v \neq v_i \\ \exists v_i \phi(v_i) & v = v_i. \end{cases}$

2.5 Variable Scoping

Definition 2.5.1. *Let* $\phi \in \mathrm{Form}(L)$, $t \in \mathrm{Term}(L)$ *and* $v_i \in V$ *a variable in* ϕ. *Then* t *is free from* v_i *in* ϕ *if for every* $v_j \in t$, v_i *is not in the scope* $\forall v_j$ *or* $\exists v_j$ *in* ϕ.

Lemma 2.5.2. *Suppose* $\phi \in \mathrm{Form}(L)$ *is a free variable in* ϕ *and let* t *be an* L*-term that is free from* v_i *in* ϕ. *Then*

$$\vDash (\forall v_i \; \phi(v_i)) \to \phi(t).$$

Definition 2.5.3. *Suppose* $\Gamma \subseteq \mathrm{Form}(L)$, $\mathcal{A}$ *is an* L*-structure and* β *is an assignment. Then* $\mathcal{A} \vDash \Gamma\lfloor\beta\rfloor$ *if* $\mathcal{A} \vDash \psi\lfloor\beta\rfloor$ *for all* $\psi \in \Gamma$. *Moreover,* $\Gamma \vDash \phi$ *if for all* L*-structures* $\mathcal{A}$ *and* $\mathcal{A}$*-assignments* β, *we have that* $\mathcal{A} \vDash \Gamma\lfloor\beta\rfloor$ *implies* $\mathcal{A} \vDash \phi\lfloor\beta\rfloor$.

Lemma 2.5.4. *Suppose* $\phi(v_1, \ldots, v_n) \in \mathrm{Form}(L)$. *Then* $\mathcal{A} \vDash \phi(v_1, \ldots, v_n)$ *if and only if* $\forall(v_1 \ldots v_n)\phi(v_1, \ldots, v_n)$ *where* $\forall(v_1 \ldots v_n) = \forall v_1 \ldots \forall v_n$.

Lemma 2.5.5. *Let* $\phi \in \mathrm{Form}(L)$.

1. $\vDash \neg\forall x \; \phi \leftrightarrow \exists x \; \neg\phi$.
2. $\vDash \neg\exists x \; \phi \leftrightarrow \forall x \; \neg\phi$.
3. $\vDash \forall x \; \phi \leftrightarrow \; \neg(\exists x \; \neg\phi)$.
4. $\vDash \exists x \; \phi \leftrightarrow \; \neg(\forall x \; \neg\phi)$.

Corollary 2.5.6. *Let* $\phi \in \mathrm{Form}(L)$, *then there exists* $\psi \in \mathrm{Form}(L)$ *such that* $\vDash \phi \leftrightarrow \psi$, *where* ψ *only has the connectives* $\{\wedge, \to, \perp, \forall\}$.

Lemma 2.5.7. *Suppose $\phi, \theta \in \mathrm{Form}(L)$ and $v \in V$ is not a free variable in θ. Then $\vDash \forall v\ (\theta \to \phi(v))$ implies that $\vDash \theta$ which implies that $\forall v\ \phi(v)$.*

2.6 Natural Deduction

The deduction rules we defined for propositional logic are similarly defined for predicate logic. To these, we define additional rules.

- $\forall$ introduction, under the deduction $\begin{matrix} D \\ \phi(x) \end{matrix}$ where x is not free for any uncanceled hypothesis of D, we have

$$\begin{matrix} D \\ \phi(x). \end{matrix}$$

- $\forall$ elimination, under the deduction $\forall v\ \phi(v)$, for $t \in \mathrm{Term}(L)$ free from v in ϕ we have

$$\frac{\forall\ \phi(v)}{\phi(t)} (\forall\ E).$$

Lemma 2.6.1. *Let $\Gamma \subseteq \mathrm{Form}(L)$, then if v is not a free variable in Γ then $\Gamma \vdash \phi(v)$ implies that $\Gamma \vdash \forall v\ \phi(v)$.*

2.7 Completeness and the Model Existence Lemma

Lemma 2.7.1. *If Γ is a set of L-sentences which is consistent, that is $\Gamma \nvdash \bot$, then there is an L-structure, which we call a model, where $\mathcal{A} \vDash L$.*

Definition 2.7.2. *Let L be a language. A theory is a collection of L-sentences, $T \subseteq \mathrm{Form}(L)$, such that $T \vdash \phi$ implies that $\phi \in T$.*

- *For a theory T, we call $\Gamma \subseteq \mathrm{Form}(L)$ a set of axioms of T if*

$$T = \{\phi \in \mathrm{Form}(L) : \text{A sentence such that } \Gamma \vdash \phi\}.$$

- *A theory T is a Henkin theory if for each sentence of the form $\exists x\ \phi(x)$ there is a constant c such that $(\exists x\ \phi(x) \to \phi(c)) \in T$. Such a c is called a Henkin witness.*
- *A extension of a theory T for a language L is any theory T' for a language L' such that $T \subseteq T'$.*

Definition 2.7.3. *Let T be an L-theory. For each $\theta = \exists v\ \phi(v)$ in L, add a distinct constant c_θ for each θ. The resulting language is denoted L^* and we define T^* to be the theory of axioms*

$$T \cup \{\exists v\ \phi(v) \to \phi(c_\theta) : \theta = \exists v\ \phi(v) \text{ a sentence}\}.$$

We call T^* an extensions by constants and denote $L^* = L \cup C$ for

$$C = \{c_\theta : \theta = \exists v\ \phi(v) \text{ is a sentence}\}.$$

Lemma 2.7.4. *Let T be a theory and T^* the extension by constants. Then $T = \mathrm{Form}(L) \cap T^*$.*

Lemma 2.7.5. *Let Γ be a set of L-sentences. If $\Gamma \cup \{\exists v\, \phi(v) \to \phi(c)\} \vdash \psi$ where c is a constant not in Γ and ψ, then $\Gamma \vdash \psi$.*

Corollary 2.7.6. *If T is consistent then T^* is also consistent.*

For a theory T, define the Henkin extension, T_ω, using the following inductive process. Let $T_0 = T$, $T_{n+1} = (T_n)^*$ and $T_\omega = \bigcup_{n \in \mathbb{Z}_{\geq 0}}$.

Lemma 2.7.7. *We have T_ω is a Henkin theory.*

Lemma 2.7.8. *For a theory T we have that $T = T_\omega \cap \mathrm{Form}(L)$.*

Corollary 2.7.9. *When T is consistent, there is a consistent Henkin extension T_ω of T.*

Lemma 2.7.10 (Lindenbaum)**.** *Every consistent theory is contained in a maximally consistent theory within the same language.*

Lemma 2.7.11. *Any extension of a Henkin theory in the same language is a Henkin theory.*

2.8 Soundness of Equality Deduction Rules

Lemma 2.8.1. *Suppose that β is an $\mathcal{A}$-assignment for an L-structure $\mathcal{A}$. Then*

$$v^{\beta}(\phi\lfloor t/v \rfloor) = v^{\beta\lfloor t^{\mathcal{A}}\lfloor\beta\rfloor/v\rfloor}(\phi(v)).$$

Lemma 2.8.2. *Given β an $\mathcal{A}$-assignment and $s, t \in \mathrm{Term}(L)$ we have*

$$(t\lfloor t/v \rfloor)^{\mathcal{A}}\lfloor\beta\rfloor = t^{\mathcal{A}}\left\lfloor \beta\left(s^{\mathcal{A}}\lfloor\beta\rfloor/v\right)\right\rfloor.$$

2.9 Conclusions from the Completeness Theorem

Theorem 2.9.1 (Compactness)**.** *Suppose that Σ is a set of L-sentences. Then Σ has a model if and only if every finite subset of Σ has a model.*

Theorem 2.9.2 (Downward Lowenheim-Skolem)**.** *Suppose L is a countable first order language and that $\mathcal{A}$ is an L-structure. Let*

$$T = \mathrm{Th}(\mathcal{A}) = \{\theta : \mathcal{A} \vDash \theta, \theta \in \mathrm{Form}(L) \text{ a sentence}\}.$$

Then there is a countable model $\mathcal{B}$ such that $\mathcal{B} \vDash T$.

Using the Lindstrom theorem one can show that predicate logic is the strongest compact and Lowenheim-Skolem logic.

2.10 Decidability and the Entscheidungsproblem

It has been shown that there is no algorithm to show whether any given first-order formula is valid. The proof of this statement utilizes Gödel's incompleteness theorem.

Definition 2.10.1. *A set of natural numbers S is recursively enumerative (RE) if there is a (not necessarily terminating) algorithm that generates all the elements of S.*

Theorem 2.10.2 (Gödel's Incompleteness Theorem)**.** *No consistent set of axioms that is RE is capable of generating all valid sentences of $\langle \mathbb{N}, +, \cdot, 0 \rangle = \mathcal{N}$.*

That is, no such algorithm can determine whether a formula $\mathcal{N}$ is valid, as this would mean the set of valid sentences of $\mathcal{N}$ is RE.
The incompleteness theorem refers to the failure of a logical system to determine the validity of every sentence.

2.11 Dense Linear Orders

Let L be a language containing a single binary relation, $\leq$.

Definition 2.11.1. *An L-structure, $\mathcal{A}$, is a linear order if it satisfies the following.*

ϕ-1. $\forall x_1 \forall x_2 (x_1 \leq x_2) \wedge (x_2 \leq x_1) \to (x_1 \doteq x_2)$.

ϕ-2. $\forall x_1 x_2 x_3 (x_1 \leq x_2) \wedge (x_2 \leq x_3) \to (x_1 \leq x_3)$.

ϕ-3. $\forall x_1 x_2 (x_1 \leq x_2) \vee (x_2 \leq x_1)$.

A linear order is dense if it also satisfies the following.

ϕ-4. $\forall x_1 \forall z_2 \exists x_3 (x_1 \leq x_2) \to (x_1 \leq x_3) \wedge (x_3 < x_2)$, *where* $x_1 < x_2 \doteq (x_1 \leq x_2) \wedge \neg (x_1 \doteq x_2)$.

A dense linear order has no endpoints if it also satisfies the following.

ϕ-5. $\forall x_1 \exists x_2 x_1 < x_2$.

ϕ-6. $\forall x_1 \exists x_2 x_2 < x_1$.

An isomorphism between L-structures $\mathcal{A}$ and $\mathcal{B}$ is a bijection $F : A \to B$ such that the following hold.

- For every constant $c \in L$, we have $F\left(c^{\mathcal{A}}\right) = c^{\mathcal{B}}$.
- For every m-ary function $f \in L$, we have that $F\left(f^{\mathcal{A}}(a_1, \dots, a_m)\right) = f^{\mathcal{B}}\left(F(a_1), \dots, F(a_m)\right)$.
- For every k-ary relation $R \in L$ we have $R^{\mathcal{A}}(a_1, \dots, a_k)$ if and only if $R^{\mathcal{B}}\left(F(a_1), \dots, F(a_k)\right)$.

Lemma 2.11.2 (Los-Vaught Test)**.** *For every sentence θ either $\Delta \vdash \theta$ or $\Delta \vdash \neg\theta$.*

Theorem 2.11.3 (Cantor)**.** *If $\mathcal{A}$ and $\mathcal{B}$ are two countable dense linear orders without endpoints, then $\mathcal{A}$ and $\mathcal{B}$ are isomorphic.*

Corollary 2.11.4. *If $\mathcal{A}$ and $\mathcal{B}$ are isomorphic, then for all L-sentences θ we have that $\mathcal{A} \vDash \theta$ if and only if $\mathcal{B} \vDash \theta$.*

3 Set Theory

Set theory is a first-order theory.

3.1 Basic Set Theory

Definition 3.1.1. *Two sets are equal if and only if* $\forall x (x \in A) \leftrightarrow (x \in B)$.

Definition 3.1.2. *If* A *is a set, then the power set,* $\mathcal{P}(A)$ *is such that* $\forall x (x \subseteq A \to x \in \mathcal{P}(A))$*, where* $x \subseteq A$ *if and only if* $\forall y \in x \to \forall y \in A$.

Definition 3.1.3. *The ordered pair* (x, y) *is defined to be the set* $\{\{x\}, \{x, y\}\}$.

- *For sets* A *and* B*, their product is*

$$A \times B = \{(a, b) : a \in A, b \in B\}.$$

- *The* n*-fold product of a set* A *is defined inductively as*

$$A^0 = \{\emptyset\}, \quad A^1 = A, \quad A^2 = A^1 \times A^1, \; \dots, A^n = A^{n-1} \times A, \; \dots$$

- *The set of finite sequences of elements of* A *is* $\bigcup_{n \in \mathbb{N}} A^n$.

Definition 3.1.4. *A function* $f : A \to B$ *is a subset* $f \subseteq A \times B$ *which has the property that*

$$\forall x \in A \; \exists! b \in B \; (a, b) \in f.$$

where $\exists! c \in C \; \phi(c)$ *if and only if* $\exists c \in C \;\; (\phi(c) \wedge (\forall d \in C \phi(d) \to (d \doteq c)))$*, in words this means that there exists a unique* $c \in C$ *such that* $\phi(c)$ *holds.*

- $A = \mathrm{dom}(f)$ *is the domain of* f.
- $B = \mathrm{ran}(f)$ *is the range of* f.
- $B^A \subseteq \mathcal{P}(A \times B)$ *is the set of functions* $f : A \to B$.
- *For* $X \subseteq A$*, we let* $f(X) = \{f(x) : x \in X\}$.

3.2 Cardinality

Definition 3.2.1. *We say two sets* A *and* B *are equinumerous,* $A \approx B$*, if there is a bijection* $f : A \to B$.

- *A set is finite if it is equinumerous with some* $n \in \mathbb{N}$.
- *A set* A *is countably infinite if* $A \approx \mathbb{N}$.
- *A set is countable if it is finite or countably infinite.*

Proposition 3.2.2.

1. *Every subset of a countable set is countable.*

2. *A set A is countable if and only if there is an injection $f : A \to \mathbb{N}$.*

3. *If A and B are countable then so is $A \times B$.*

4. *If the axiom of choice* is assumed, then for countable sets $A_0, A_1, \ldots$ we have that $\bigcup_{n \in \mathbb{N}} A_n$ is countable.*

Theorem 3.2.3 (Cantor)**.** *If X is a set, then there is no surjective function $f : X \to \mathcal{P}(X)$. Moreover, this implies that $X \not\approx \mathcal{P}(X)$.*

Definition 3.2.4. *For sets A and B we write $|A| \leq |B|$ when there exists an injective function $f : A \to B$.*

If $|A| \leq |B|$ then A is equinumerous with a subset of B.

Theorem 3.2.5 (Cantor-Schroder-Bernstein)**.** *Suppose that A, B are set and $f : A \to B$ and $g : B \to A$ are injective functions. Then $A \approx B$.*

3.3 Axioms of Set Theory

The Zermelo-Frankel Axioms facilitate the construction of sets using a first-order language.

ZF1. Extensionality - $\forall x \forall y \ ((x \doteq y) \leftrightarrow \forall z \left((z \in x) \leftrightarrow (z \in y)\right))$. In other words, two sets are equal if and only if they have the same elements.

ZF2. Empty set - $\exists x \forall y \ (y \notin x)$. In other words, there exists a set with no elements. The unique set with this property is denoted $\emptyset$.

ZF3. Pairing - $\forall x \forall y \exists z \ (\forall w \ (w \in z \leftrightarrow ((w \in x) \vee (w \in y))))$. In other words, we can combine the elements of two sets to form a new set.

ZF4. Union - $\forall A \exists B \forall x \ ((x \in B) \leftrightarrow \exists z ((z \in A) \wedge (x \in z)))$. In other words, we can write any set as the pairing of sets.

ZF5. Power Set - $\forall A \exists B \forall z \ ((z \in B) \leftrightarrow (z \subseteq A))$. In other words, for any set, we can form a set containing its subsets.

ZF6. Axiom Scheme of Specification - For $P(x, y_1, \ldots, y_r) \in \mathrm{Form}(L)$ we have that $\forall A \forall (y_1, \ldots, y_r) \exists B \forall x \ ((x \in B) \leftrightarrow ((x \in A) \wedge P(x, y_1, \ldots, y_r)))$. In other words, for a formula and a set, we can construct the subset of elements of the set for which the formula holds.

Using these axioms we can construct the following commonly used mathematical notions.

1. Intersection - $\cap C = \{x \in A : \forall z ((z \in C) \to (x \in Z))\}$.
2. Cartesian Product of Non-empty Set A and B - $A \times B = \{w \in \mathcal{P}(\mathcal{P}(A \cup B)) : \phi(w)\}$, where

$$\phi(w) = \exists x \exists y \forall z \left((x \in A) \wedge (Y \in B)\right) \wedge \left(w = \{\{x\}, \{x, y\}\}\right).$$

Definition 3.3.1. *For a set a, the successor of a is the set $a^\dagger = a \cup \{a\}$. A set, A, is inductive if*

$$(\emptyset \in A) \wedge \left(\forall x \left((x \in A) \to \left(x^\dagger \in A\right)\right)\right).$$

ZF7. Infinity - $\exists A \ (\emptyset \in A) \wedge \left(\forall x \left((x \in A) \to \left(x^\dagger \in A\right)\right)\right)$. In other words, there exists a set for whose elements when paired with itself as a singleton set is still in the set.

Definition 3.3.2. *Let A be an inductive set, and let $\phi(A) = (\emptyset \in A) \wedge \left(\forall x \left((x \in A) \rightarrow \left(x^\dagger \in A\right)\right)\right)$. Then*

$$\mathbb{N} = \{x \in A : \phi(B) \rightarrow (x \in B)\}.$$

Remark 3.3.3. *The above definition is agnostic to the choice of set A, and can be thought of as the intersection of all inductive sets. We will also denote this set by ω.*

Theorem 3.3.4. *Let $\mathbb{N}$ be defined as above.*

1. *$\mathbb{N}$ is an inductive set, with the property that $\mathbb{N} \subset B$ for any inductive set B.*
2. *Let $P(x)$ be a formula, then $\forall k((k \in \mathbb{N}) \rightarrow P(k))$ if $P(x)$ satisfies the following.*
   (a) $P(\emptyset)$ holds, and
   (b) $\forall k \; \left((k \in \mathbb{N}) \rightarrow \left(P(k) \rightarrow P(k^\dagger)\right)\right)$.

3.4 Linear Orderings

Definition 3.4.1. *A linear ordering $(A, \leq)$ is a well-ordering if every non-empty subset of A has a least element.*

Definition 3.4.2. *Suppose $\mathcal{A}_i = (A_i, \leq_i)$ for $i = 1, 2$ are linear orderings. Then $\mathcal{A}_1$ and $\mathcal{A}_2$ are isomorphic, $\mathcal{A}_1 \simeq \mathcal{A}_2$, if there is a bijection $\alpha : A_1 \rightarrow A_2$ such that $\forall a, b \in a \leq_1 b$ if and only if $\alpha(a) \leq_2 \alpha(b)$.*

- *For any map α, if the forward implication of the above holds then α is order preserving.*

Definition 3.4.3. *Let $\mathcal{A}_i$ for $i = 1, 2$ be linear orderings.*

- *The reverse-lexicographic product $\mathcal{A}_1 \times \mathcal{A}_2$ is the linear order $(A_1 \times A_2, \leq)$ defined by the property that $(a_1, a_2) \leq (a_1', a_2')$ if and only if $a_2 < a_2'$ or $a_2 \doteq a_2'$ and $a_1 \leq_1 a_2$.*
- *The disjoint union of A_1 and A_2 is defined as*

$$A_1 \sqcup A_2 = \{(a, i) \in (A_1 \cup A_2) \times \{0, 1\} : (a \in A_1) \leftrightarrow (i \doteq 0) \wedge (a \in A_2) \leftrightarrow (i \doteq 1)\}.$$

- *The sum $\mathcal{A}_1 + \mathcal{A}_2$ is the linear ordering $(A_1 \sqcup A_2, \leq)$ where $a_1 \leq a_2$ for all $a_1 \in A_1$ and $a_2 \in A_2$. All other orderings are as in $\mathcal{A}_1$ and $\mathcal{A}_2$.*

Lemma 3.4.4. *Let $\mathcal{A}_1$ and $\mathcal{A}_2$ be linear orders.*

1. *$\mathcal{A}_1 + \mathcal{A}_2$ and $\mathcal{A}_1 \times \mathcal{A}_2$ are linear orders.*
2. *If $\mathcal{A}_1$ and $\mathcal{A}_2$ are well ordered sets, then so are $\mathcal{A}_1 + \mathcal{A}_2$ and $\mathcal{A}_1 \times \mathcal{A}_2$.*

3.5 Ordinals

Definition 3.5.1. *Let X and α be sets.*

1. *X is transitive if every element of X is a subset of X.*
2. *α is ordinal if*

(a) α is transitive, and

(b) the relation $x < y$ if and only if $x \in y$ is a strict well-ordering.

Lemma 3.5.2. *If α is an ordering, then $\alpha^\dagger$ is also an ordering.*

Proposition 3.5.3. *Let ω be as defined previously.*

1. *If $n \in \omega$ then n is an ordinal.*
2. *ω is a transitive set.*

Proposition 3.5.4. *Suppose α is an ordinal.*

1. *$\alpha \notin \alpha$.*
2. *If $\beta \in \alpha$ then β is an ordinal.*
3. *If β is an ordinal such that $\beta \subsetneq \alpha$, then $\beta \in \alpha$.*
4. *$\alpha = \{\beta : \beta$ an ordinal such that $\beta \in \alpha\}$.*

Definition 3.5.5. *If α and β are ordinals then $\alpha < \beta$ means that $\alpha \in \beta$ and $\alpha \leq \beta$ means that $\alpha < \beta$ or $\alpha = \beta$.*

Theorem 3.5.6. *Suppose that α, β, γ are ordinals.*

1. *If $\alpha < \beta$ and $\beta < \gamma$ then $\alpha < \gamma$.*
2. *If $\alpha \leq \beta$ and $\beta \leq \alpha$ then $\alpha = \beta$.*
3. *We have the trichotomy $\alpha < \beta$, $\alpha = \beta$ and $\beta < \alpha$.*
4. *A collection of ordinals, X, is well ordered. Moreover, the least element is given by $\cap X$.*

Corollary 3.5.7.

1. *If X is a set of ordinals, then $\cup X$ is an ordinal.*
2. *ω is an ordinal.*

Theorem 3.5.8. *If $(A, \leq)$ is a well-ordered set, then there is a unique ordinal which is similar to $(A, \leq)$.*

Definition 3.5.9. *Suppose that $(A, \leq)$ is a well-ordered set. Then $X \subseteq (\subsetneq) A$ is an (proper) initial segment of A if whenever $y < x \in X$ then $y \in X$.*

Lemma 3.5.10. *Suppose that $(A, \leq)$ is a well-ordered set. If $X \subset A$ is a proper initial segment of A, then there is a $x \in A$ such that*

$$X = \{a \in A : a < x\} =: A[x].$$

Theorem 3.5.11. *Suppose that $(A, \leq)$ is a well-ordered set, $f : A \to A$ is order-preserving and $f[A]$ is an initial segment of A. Then $\forall x \in A\ f(x) = x$.*

Corollary 3.5.12. *For ordinals α and β, if $\alpha \neq \beta$ then $\alpha \not\approx \beta$.*

Definition 3.5.13. *Suppose that $F(x, y, z_1, \ldots, z_r) \in \mathrm{Form}(L)$ is such that for fixed sets $s_1, \ldots, s_r$ and b there is a unique set a such that $F(a, b, s_1, \ldots, s_r)$ holds.*

- *$b \mapsto a$ defines a function F called an operation on sets with parameter variables $z_1, \ldots, z_r$.*
- *The sets $s_1, \ldots, s_r$ are the parameters of the operation.*

ZF8. Replacement - Let $F(x, y, z_1, \ldots, z_r)$ be an operation on sets with parameters $s_1, \ldots, s_r$, and let B be a set. Then there is a set A such that

$$A = \{a : F(a, b, s_1, \ldots, s_r) \text{ holds for some } b \in B\}.$$

In other words, the image of an operation on a set is also a set.

3.6 Transfinite Induction

Theorem 3.6.1 (Transfinite Induction)**.** *Let $P(x)$ be a property of sets. Assume that*

$$\forall \alpha (\forall \beta ((\beta < \alpha) \to P(\beta))) \to P(\alpha)).$$

Then $P(\gamma)$ holds for all ordinals γ.

Theorem 3.6.2. *Suppose that α is an infinite ordinal, $\omega \leq \alpha$, then $\alpha \approx \alpha \times \alpha$.*

Corollary 3.6.3.

1. *If $(A, \leq)$ is an infinite well ordered set, then $|A| = |A| \times |A|$.*
2. *Assuming the axiom of choice*, then any set A can be well ordered so that if A is infinite we have that $|A| = |A| \times |A|$. This is called the fundamental theorem of cardinal arithmetic.*

3.7 Transfinite Recursion

Transfinite recursion lets us construct sets for ordinals that can be obtained from smaller ordinals via some operation.

- Let $G(\cdot)$ be a set constructed from an ordinal. Then there exists an operation F such that

$$G(\alpha) = F(\{G(\beta) : \beta < \alpha\}) = F(G \restriction \alpha).$$

Theorem 3.7.1 (Transfinite Recursion)**.** *Let F be an operation on sets, then there is an operation G such that for all ordinals α we have*

$$G(\alpha) = F(G \restriction \alpha).$$

Moreover, if G' also has this property, then for all ordinals α we have $G'(\alpha) = G(\alpha)$.

Remark 3.7.2. *Note that G and G' may still differ on non-ordinal sets.*

Lemma 3.7.3 (Lindenbaum)**.** *Suppose L is a first order language whose alphabet of symbols, P, is well ordered. Let Σ be a consistent set of L-sentences. Then there is a consistent set $\Sigma^* \supset \Sigma$ of L-sentences such that every L-sentence ψ is such that either $\psi \in \Sigma^*$ or $\neg\psi \in \Sigma^*$.*

3.8 The Axiom of Regularity

ZF9. Regularity - $\forall((x \neq \emptyset) \rightarrow (\exists a\ ((a \in x) \wedge (a \cap x = \emptyset))))$. In other words, there is no set b such that $b \in b$.

4 The Axiom of Choice

4.1 The Well-Ordering Principle

Definition 4.1.1. *The axiom of choice is applied to a set of non-empty sets, A, and says that there is a function $f : A \to \cup A$ such that $f(a) \in a$ for all $a \in A$.*

The Zermelo-Frankel axioms (ZF) with the axiom of choice is denoted ZFC.

Definition 4.1.2. *Let X be a non-empty set of sets, let $A = \mathcal{P}(X) \setminus \{\emptyset\}$. By the axiom of choice, there is a function $f : A \to X$ such that $f(Y) = Y$ for all non-empty subsets of X. This function is called a choice function on X.*

Theorem 4.1.3 (Well-Ordering Principle)**.** *Suppose that X is a non-empty set and let $f : \mathcal{P}(X) \setminus \{\emptyset\} \to X$ be a choice function. Then there is a well-ordering $\leq$ of X.*

Lemma 4.1.4. *For any set A there exists an ordinal α such that $h : \alpha \to A$ cannot be injective.*

Corollary 4.1.5. *Assume ZF, then the axiom of choice is equivalent to the well-ordering principle, $\mathrm{ZF} \vdash (\mathrm{AC} \leftrightarrow \mathrm{WO})$.*

Corollary 4.1.6. *Assume ZFC.*

1. *For any set A, there is an ordinal, α, such that $\alpha \approx A$.*
2. *For any sets A or B, we have that $|A| \leq |B$ or $|B| \leq |A|$.*
3. *For any infinite set A, we have that $|A| = |A \times A|$.*

Lemma 4.1.7. *Assume ZFC. For sets $\emptyset \neq A$ and B we have $|A| \leq |B|$ if and only if there exists a surjective function $h : B \to A$.*

4.2 Cardinals and Cardinality

Throughout this section assume ZFC.

Definition 4.2.1. *An ordinal α is cardinal if it is not equinumerous with any $\beta < \alpha$.*

Lemma 4.2.2. *For any set A, there is a unique cardinal such that $\alpha \approx A$. We call this cardinal the cardinality of A and denote it be $\mathrm{card}(A)$ or $|A|$.*

Definition 4.2.3. *For disjoint sets A and B with $|A| = \kappa$ and $|B| = \lambda$ for cardinals κ and γ. Define*

- *$\kappa + \lambda = |A \cup B|$, and*
- *$\kappa \cdot \lambda = |A \times B|$.*

Theorem 4.2.4. *For cardinals κ and λ where $\kappa \leq \lambda$ and λ is infinite, we have that*

1. $\kappa + \lambda = \lambda$, *and*

2. $\kappa \cdot \lambda = \lambda$ *if* $\kappa \neq 0$.

Theorem 4.2.5. *For an infinite set A with cardinality γ, suppose each element of A is a set with cardinality $\leq \kappa$. Then $|\cup A| \leq \lambda \cdot x$.*

4.3 Zorn's Lemma

Definition 4.3.1. *A partially ordered set, or poset, $(A, \leq)$ has the following properties.*

1. $\forall xyz \in A\ (x \leq y \leq z \to x \leq z)$.

2. $\forall xyz \in A\ ((x \leq y) \wedge (y \leq x) \to (x \doteq y))$.

3. $\forall xyz \in A\ (x \leq x)$.

Definition 4.3.2. *For a poster $(A, \leq)$.*

- *A chain, $C \subseteq A$, has the property that*

$$(\forall x \in C)(\forall y \in C)((x \leq y) \vee (y \leq x)).$$

- *An upper bound, $a \in A$ of C has the property that*

$$\forall x \in C\ x \leq a.$$

Definition 4.3.3. *Zorn's Lemma, ZL, supposes we have a non-empty poset $(A, \leq)$ in which every chain in A has an upper bound in A and says that A has a maximal element.*

Theorem 4.3.4.

1. *Assume* ZFC, *then* ZL *holds.*

2. *Assume* ZF *and* ZL, *then* AC *holds.*

5 Applications

5.1 An Introduction to Computational Semantics

Mathematical logic has significant applications in the realm of theoretical computer science. The foundations of computation are described logically using semantics. Mathematical logic describes the syntax of how we construct formulas and expressions, whereas semantics contextualises this into a framework to capture different philosophies and properties. Three main types of semantics are used to describe computation.

1. Operational Semantics - Contextualises computations are executions running on abstract machines.
2. Denotational Semantics - Uses mathematical objects, and functions, to describe computation.
3. Axiomatic Semantics - Defines computation in terms of logical formulas that are satisfied during execution.

Here we will provide an introduction to operational semantics which describes how a program executes expressions on an abstract machine.

- Var is a set of program variables.
- Int is a set of constant integers.
- Exp is a domain of expressions. We specify an expression using a syntax defined by the grammar,

$$e ::= x \mid n \mid e_1 + e_2 \mid e_1 * e_2 \mid x := e_1; e_2.$$

Using this we formalize our abstract machine as a configuration, $\mathrm{Config} := \mathrm{Store} \times \mathrm{Exp}$ where $\mathrm{Store} = \mathrm{Var} \rightharpoonup \mathrm{Int}$.

- A small-step operational semantic $\rightarrow \subseteq \mathrm{Config} \times \mathrm{Config}$ describes how we transition between configurations. We can define inference rules for this semantic in the same as we did previously.

$$\frac{n = \sigma(x)}{\langle \sigma, x \rangle \to \langle \sigma, n \rangle}\mathrm{Var}$$
$$\frac{\langle \sigma, e_1 \rangle \to \langle \sigma', e_1' \rangle}{\langle \sigma, e_1 + e_2 \rangle \to \langle \sigma', e_1' + e_2 \rangle}\mathrm{LADD}$$
$$\frac{\langle \sigma, e_1 \rangle \to \langle \sigma', e_2' \rangle}{\langle \sigma, n + e_2 \rangle \to \langle \sigma', n + e_2' \rangle}\mathrm{RADD}$$
$$\frac{p = m + n}{\langle \sigma, n + m \rangle \to \langle \sigma, p \rangle}\mathrm{ADD}$$
$$\frac{\langle \sigma, e_1 \rangle \to \langle \sigma', e_1' \rangle}{\langle \sigma, e_1 * e_2 \rangle \to \langle \sigma', e_1' * e_2 \rangle}\mathrm{LMUL}$$
$$\frac{\langle \sigma, e_2 \rangle \to \langle \sigma', e_2' \rangle}{\langle \sigma, n * e_2 \rangle \to \langle \sigma', n * e_2 \rangle}\mathrm{RMUL}$$
$$\frac{p = m \times n}{\langle \sigma, m * n \rangle \to \langle \sigma, p \rangle}$$
$$\frac{\langle \sigma, e_1 \rangle \to \langle \sigma', e_1' \rangle}{\langle \sigma, x := e_1; e_2 \rangle \to \langle \sigma', x := e_1'; e_2 \rangle}\mathrm{ASSGN1}$$
$$\frac{\sigma' = \sigma[x \mapsto n]}{\langle \sigma, x := n; e_2 \rangle \to \langle \sigma', e_2 \rangle}\mathrm{ASSGN}$$

- Similarly, we can define large-step semantics, $\Downarrow \subseteq (\mathrm{Store} \times \mathrm{Exp}) \times (\mathrm{Store} \times \mathrm{Int})$, so that we evaluate expressions directly, rather than performing lots of small-step deductions. Again this is specified by a set of inference rules.

$$\frac{}{\langle \sigma, n\rangle \Downarrow \langle \sigma, n\rangle}\text{INT}$$

$$\frac{n = \sigma(x)}{\langle \sigma, x\rangle \Downarrow \langle \sigma, n\rangle}\text{VAR}$$

$$\frac{\langle \sigma, e_1\rangle \Downarrow \langle \sigma', n_1\rangle \quad \langle \sigma', e_2\rangle \Downarrow \langle \sigma'', n_2\rangle \quad n = n_1 + n_2}{\langle \sigma, e_1 + e_2\rangle \Downarrow \langle \sigma'', n\rangle}\text{ADD}$$

$$\frac{\langle \sigma, e_1\rangle \Downarrow \langle \sigma', n_1\rangle \quad \langle \sigma', e_2\rangle \Downarrow \langle \sigma'', n_2\rangle \quad n = n_1 \times n_2}{\langle \sigma, e_1 * e_2\rangle \Downarrow \langle \sigma'', n\rangle}\text{MUL}$$

$$\frac{\langle \sigma, e_1\rangle \Downarrow \langle \sigma', n_1\rangle \quad \langle \sigma'[x \mapsto n_1], e_2\rangle \Downarrow \langle \sigma'', n_2\rangle}{\langle \sigma, x := e_1; e_2\rangle \Downarrow \langle \sigma'', n_2\rangle}\text{ASSGN}$$

Using the framework we can then go on to investigate properties such as soundness, and completeness for computational programs. This has ramifications for the construction of programming languages and hence is a fundamental theory of computer science.