Mathematical Logic

Eduardo Freire

May 2021

1 Propositional Logic

Definition 1.1. *Let $Vars_P := \{P_n : n \in \mathbf{N}\}$ be the set of the symbols $P_1, P_2, \ldots,$ each called a propositional variable. We define the **language of propositional logic** as $\mathcal{L}_P := Vars_P \cup \{\rightarrow, \neg\}$.*

Definition 1.2. *Let ϕ be an $\mathcal{L}_P$ string. We say that ϕ is a **propositional formula** (also called **p-formula**) if and only if*

1. *ϕ is a propositional variable, or*
2. *$\phi :\equiv (\alpha \rightarrow \beta)$ where α and β are propositional formulas, or*
3. *$\phi :\equiv (\neg\alpha)$ and α is a propositional formula.*

Definition 1.3. *An **assignment function** is any function with domain $Vars_P$ and codomain $\{T, F\}$. Given an assignment function s, we define the function $\bar{s}$ whose domain is the set of all p-formulas and codomain is $\{T, F\}$ as follows:*

$$\bar{s}(\phi) := \begin{cases} s(\phi) & \phi \in Vars_P, \\ F & \phi :\equiv (\neg\alpha) \text{ and } \bar{s}(\alpha) = T, \\ F & \phi :\equiv (\alpha \rightarrow \beta) \text{ and } \bar{s}(\alpha) = T \text{ and } \bar{s}(\beta) = F, \\ T & \text{otherwise.} \end{cases}$$

*Also, if Σ is a set of p-formulas, we say that s **satisfies** Σ if and only if $\bar{s}(\sigma) = T$ for every $\sigma \in \Sigma$. Otherwise, we say that s **does not satisfy** Σ. If there is some assignment function s' that satisfies Σ, we say that Σ is **satisfiable**.*

Definition 1.4. *Let ϕ be a p-formula. If $\bar{s}(\phi) = T$ for every assignment function s, we say that ϕ is a **tautology**. On the other hand, if $\bar{s}(\phi) = F$ for every assignment function s, we call ϕ a **contradiction**. In particular, we define $\top$ as the tautology $(P_1 \rightarrow (P_1 \rightarrow P_1))$ and $\bot$ as the contradiction $\neg\top$, i.e $\neg(P_1 \rightarrow (P_1 \rightarrow P_1))$.*

Definition 1.5. *Let Λ be a set of p-formulas such that for every p-formula ϕ, $\phi \in \Lambda$ if and only if*

1. *$\phi :\equiv (A \rightarrow (B \rightarrow A))$, or*

2. $\phi :\equiv ((A \to (B \to C)) \to ((A \to B) \to (A \to C)))$, *or*
3. $\phi :\equiv ((\neg B \to \neg A) \to (A \to B))$

where A, B, C are p-formulas. We call Λ the set of ***logical axioms****.*

Lemma 1.1. *Every $\lambda \in \Lambda$ is a tautology.*

Proof. This is trivial to check case by case, using the definition of assignment functions for p-formulas. □

Lemma 1.2. *Let α and β be p-formulas and s be an assignment function such that $\bar{s}(\alpha) = T$ and $\bar{s}(\alpha \to \beta) = T$. Then $\bar{s}(\beta) = T$.*

Proof. Assume for contradiction that $\bar{s}(\beta) = F$. Since $\bar{s}(\alpha) = T$ by assumption, it follows from the definition of $\bar{s}$ that $\bar{s}(\alpha \to \beta) = F$, which contradicts our assumption that $\bar{s}(\alpha \to \beta) = T$. Thus $\bar{s}(\beta) = T$. □

Definition 1.6. *Let Σ be a set of p-formulas and ϕ be a p-formula. We say that $\Sigma \models \phi$ if and only if every assignment function that satisfies Σ assigns ϕ to T.*

Definition 1.7. *Let Σ be a set of p-formulas and ϕ be a p-formula. We say that a finite sequence $D = (\phi_1, \phi_2, \ldots, \phi_n)$ of p-formulas whose last entry is ϕ is a* ***deduction from*** *Σ* ***of*** *ϕ if and only if for each $1 \leq i \leq n$,*

1. $\phi_i \in \Lambda \cup \Sigma$, *or*
2. *There exists $j, k < i$ such that $\phi_j :\equiv (\phi_k \to \phi_i)$.*

In this case, we write $\Sigma \vdash \phi$, read as Σ proves ϕ. If Γ is a set of p-formulas such that $\Sigma \vdash \gamma$ for every $\gamma \in \Gamma$, we write $\Sigma \vdash \Gamma$.

The following lemma has an easy proof and will be used implicitly several times.

Lemma 1.3. *Let Σ, Γ be sets of p-formulas and α, β, ϕ be p-formulas. It follows that:*

1. *If $\Sigma \vdash (\alpha \to \beta)$ and $\Sigma \vdash \alpha$, then $\Sigma \vdash \beta$,*
2. *If $\Gamma \vdash \phi$ and $\Gamma \subseteq \Sigma$, then $\Sigma \vdash \phi$,*
3. *If $\Gamma \vdash \phi$ and $\Sigma \vdash \Gamma$, then $\Sigma \vdash \phi$.*

Theorem 1.1 (Soundness Theorem)**.** *Let Σ be a set of p-formulas, ϕ be a p-formula. Then $\Sigma \vdash \phi$ implies $\Sigma \models \phi$.*

Proof. Assume that $\Sigma \vdash \phi$. We let s be an arbitrary assignment function that satisfies Σ and induct on the shortest length of deduction of ϕ. If there is a deduction of ϕ with length 1, then either $\phi \in \Lambda$ or $\phi \in \Sigma$. In the first case, ϕ is a tautology by Lemma 1.1, so $\bar{s}(\phi) = T$. The other case follows from our assumption that s satisfies Σ. Now assume inductively that if ψ is a p-formula provable from Σ such that its shortest length of deduction is less than or equal to n then $\bar{s}(\psi) = T$.

Assume that the shortest length of deduction of ϕ is $n+1$. $\phi \notin \Sigma$ and $\phi \notin \Lambda$, since its shortest length of deduction would be 1 in that case. Thus, we have ϕ_j and ϕ_k in the deduction of ϕ such that $\phi_j :\equiv \phi_k \to \phi$. By the inductive hypothesis, $\bar{s}(\phi_j) = \bar{s}(\phi_k) = T$, so it follows from Lemma 1.2 that $\bar{s}(\phi) = T$. $\square$

Lemma 1.4. *For every p-formula ϕ, $\vdash (\phi \to \phi)$.*

Proof. Let ϕ be a p-formula. The following is a deduction of $(\phi \to \phi)$ from $\{\}$.

(1)	$(\phi \to ((P_1 \to \phi) \to \phi))$	Ax 1
(2)	$((\phi \to ((P_1 \to \phi) \to \phi)) \to ((\phi \to (P_1 \to \phi)) \to (\phi \to \phi)))$	Ax 2
(3)	$((\phi \to (P_1 \to \phi)) \to (\phi \to \phi))$	MP 1,2
(4)	$(\phi \to (P_1 \to \phi))$	Ax 1
(5)	$(\phi \to \phi)$	MP 3,4.

$\square$

Theorem 1.2 (Deduction Theorem)**.** *Let Σ be a set of p-formulas and θ, ϕ be p-formulas. Then, $\Sigma \vdash (\theta \to \phi) \iff \Sigma \cup \{\theta\} \vdash \phi$.*

Proof. For the forward direction, assume that $\Sigma \vdash (\theta \to \phi)$. We can use the same deduction from Σ of $(\theta \to \phi)$ to see that $\Sigma \cup \{\theta\} \vdash (\theta \to \phi)$. But clearly $\Sigma \cup \{\theta\} \vdash \theta$, so $\Sigma \cup \{\theta\} \vdash \phi$ by modus ponens.

For the converse direction, we will assume that $\Sigma \cup \{\theta\} \vdash \phi$ and induct on the shortest length of deduction of ϕ. For the base case, assume first that $\phi \in \Lambda \cup \Sigma$. Then, $\Sigma \vdash \phi$ and $\phi \to (\theta \to \phi)$ is a logical axiom so Σ also proves it. By modus ponens, $\Sigma \vdash (\theta \to \phi)$. The last subcase of the base case is $\phi :\equiv \theta$, but we already know that $\Sigma \vdash (\theta \to \theta)$, by Lemma 1.4.

Next, assume the inductive hypothesis and let the shortest length of deduction of ϕ be $n+1$. Then, we must have ψ and $(\psi \to \phi)$ in the deduction of ϕ from $\Sigma \cup \{\theta\}$. By the inductive hypothesis (IH), $\Sigma \vdash (\theta \to (\psi \to \phi))$ an $\Sigma \vdash (\theta \to \psi)$. Then,

(1)	$\Sigma \vdash ((\theta \to (\psi \to \phi)) \to ((\theta \to \psi) \to (\theta \to \phi)))$	Ax 2
(2)	$\Sigma \vdash ((\theta \to \psi) \to (\theta \to \phi))$	MP 1,IH
(3)	$\Sigma \vdash (\theta \to \phi)$	MP 2,IH

$\square$

Lemma 1.5. *Let ψ, ϕ be p-formulas. Then $\psi, \neg\psi \vdash \phi$.*

Proof.

(1) $\neg\psi \to (\neg\phi \to \neg\psi)$ Ax 1

(2) $\neg\psi$

(3) $(\neg\phi \to \neg\psi)$ MP 1,2

(4) $(\neg\phi \to \neg\psi) \to (\psi \to \phi)$ Ax 3

(5) $(\psi \to \phi)$ MP 3,4

(6) ψ

(7) ϕ MP 5,6.

□

Definition 1.8. *A set of p-formulas Σ is inconsistent if and only if there is a p-formula ϕ such that $\Sigma \vdash \phi$ and $\Sigma \vdash \neg\phi$. Σ is consistent if and only if it is not inconsistent.*

Lemma 1.6. *Let Σ be a set of p-formulas. The following statements are equivalent:*

1. *Σ is consistent.*
2. *There is a p-formula ψ such that $\Sigma \nvdash \psi$.*
3. *There is no p-formula ψ such that $\Sigma \vdash \neg(\psi \to \psi)$.*
4. *$\Sigma \nvdash \bot$.*

Proof. For the equivalence between (1) and (2), we show instead that Σ is inconsistent if and only if Σ proves every p-formula. For the forward direction, assume that Σ is inconsistent. Then there is some formula ϕ such that $\Sigma \vdash \phi$ and $\Sigma \vdash \neg\phi$. From the deductions of each of these, we can use Lemma 1.5 to produce a deduction of any formula ψ.

For the converse direction, assume that Σ proves every p-formula. Then $\Sigma \vdash P_1$ and $\Sigma \vdash \neg P_1$, so it is inconsistent.

For the equivalence between (2) and (3), assume first that there is a p-formula ψ such that $\Sigma \vdash \neg(\psi \to \psi)$. By Lemma 1.4, $\Sigma \vdash (\psi \to \psi)$. Thus, it follows from Lemma 1.5 that Σ proves every formula, thus showing that (2) is not the case. The other direction is trivial.

(4) $\implies$ (2) is trivial, and (3) $\implies$ (4) also follows easily. □

Lemma 1.7. *Let Σ be a set of p-formulas. If ϕ is a p-formula such that $\Sigma \nvdash \phi$, then $\Sigma \cup \{\neg\phi\}$ is consistent.*

Proof. We prove by contrapositive, so assume that $\Sigma \cup \neg\phi$ is inconsistent. By Lemma 1.6, $\Sigma \cup \neg\phi \vdash \bot$, and the Deduction Theorem guarantees that $\Sigma \vdash (\neg\phi \to \bot)$. Then,

(1) $\Sigma \vdash (\neg\phi \to \bot)$	Deduction Theorem
(2) $\Sigma \vdash (\neg\phi \to \bot) \to (\top \to \phi)$	Ax 3
(3) $\Sigma \vdash \top \to \phi$	MP 1,2
(4) $\Sigma \vdash \top$	Ax 1
(5) $\Sigma \vdash \phi$	MP 4,5.

$\square$

Lemma 1.8. *The following statements are equivalent:*

1. *For every set of p-formulas Γ and every p-formula ϕ, if $\Gamma \models \phi$ then $\Gamma \vdash \phi$.*
2. *Every consistent set of p-formulas is satisfiable.*

Proof. For the forward direction, assume the contrapositive of (1) and let Δ be a consistent set of p-formulas. By Lemma 1.6, $\Delta \nvdash \bot$. By assumption, $\Delta \not\models \bot$. If there was no assignment s that satisfied Δ, then $\Delta \models \bot$ would be vacuously true, so Δ must be satisfiable.

For the converse direction, assume (2) and let Γ and ϕ be such that $\Gamma \nvdash \phi$. By Lemma 1.7, $\Gamma \cup \{\neg\phi\}$ is consistent, so it is satisfied by some assignment s. Thus, $s(\neg\phi) = T$, so $s(\phi) = F$. Since s satisfies Γ but $s(\phi) = F$, it follows that $\Gamma \not\models \phi$, as wanted. $\square$

Definition 1.9. *Let Σ be a set of p-formulas. We say that Σ is complete if and only if Σ is consistent and for every p-formula ϕ, exactly one of $\phi, \neg\phi$ is in Σ.*

Lemma 1.9. *Let Σ be a complete set of p-formulas. Then, $\Sigma \vdash \phi \iff \phi \in \Sigma$ for all p-formulas ϕ.*

Proof. For the forward direction assume that $\Sigma \vdash \phi$. If $\neg\phi \in \Sigma$ then clearly $\Sigma \vdash \neg\phi$, so Σ is inconsistent, contradicting the assumption that Σ is complete. Thus $\neg\phi \notin \Sigma$, therefore $\phi \in \Sigma$. The converse direction is trivial. $\square$

Definition 1.10. *Let Σ be a set of p-formulas. We say that Σ is maximally consistent if and only if*

1. *Σ is consistent, and*
2. *For every consistent Σ', if $\Sigma \subseteq \Sigma'$ then $\Sigma' = \Sigma$.*

Lemma 1.10. *Definitions 1.9 and 1.10 are equivalent.*

Proof. Let Σ be a set of p-formulas. For the forward direction, assume that Σ is complete and that Σ' is consistent with $\Sigma \subseteq \Sigma'$. Assume for contradiction that there is some $\psi \in \Sigma'$ such that $\psi \notin \Sigma$. Since Σ is complete we can apply Lemma 1.9 to see that, $\neg\psi \in \Sigma$, so it follows by assumption that $\neg\psi \in \Sigma'$ thus Σ' is inconsistent. This contradiction means that $\Sigma' \subseteq \Sigma$, so $\Sigma' = \Sigma$.

For the converse direction, assume that Σ is maximally consistent and let ϕ be a formula such that $\Sigma \nvdash \phi$. By Lemma 1.7, $\Sigma \cup \{\neg\phi\}$ is consistent. Since $\Sigma \cup \{\neg\phi\} \subseteq \Sigma$, it follows that $\Sigma \cup \{\neg\phi\} = \Sigma$, so $\neg\phi \in \Sigma$, therefore $\Sigma \vdash \neg\phi$, as wanted. Also, since Σ is consistent, it can only prove at most one of ϕ and $\neg\phi$ for any given ϕ. ☐

Lemma 1.11. *Let Σ be a complete set of p-formulas. If s is an assignment function such that for every propositional variable p,*

$$s(p) := \begin{cases} T & p \in \Sigma \\ F & \neg p \in \Sigma, \end{cases}$$

then s is the unique assignment that satisfies Σ.

Proof. Let s be as described in the Lemma. Notice that s is well-defined, since Lemma 1.9 guarantees that for every propositional variable p either $p \in \Sigma$ or $\neg p \in \Sigma$, but not both. To see that s satisfies Σ, we show that $s(\sigma) = T \iff \sigma \in \Sigma$ by induction on the complexity of σ.

The base case is that σ is a propositional variable, but then $s(\sigma) = T \iff \sigma \in \Sigma$ follows trivially. Assume the expected induction hypothesis. If $\sigma :\equiv \neg\alpha$, then $s(\sigma) = T \iff s(\alpha) = F \iff \neg\alpha \in \Sigma \iff \sigma \in \Sigma$. The other case is $\sigma :\equiv (\alpha \to \beta)$. For the forward direction, assume that $s(\alpha \to \beta) = T$, and notice that $s(\alpha \to \beta) = T \iff s(\alpha) = F$ or $s(\beta) = T$. If $s(\alpha) = F$, then $\neg\alpha \in \Sigma$, by the inductive hypothesis. By Lemma 1.5, $\Sigma, \alpha \vdash \beta$, so the Deduction Theorem gives that $\Sigma \vdash (\alpha \to \beta)$, thus $\sigma \in \Sigma$. Next, assume that $s(\beta) = T$. Then, $\Sigma \vdash \beta$, so $\Sigma \vdash (\beta \to (\alpha \to \beta))$, thus $\Sigma \vdash (\alpha \to \beta)$.

For the converse direction, assume that $(\alpha \to \beta \in \Sigma)$. If $\neg\alpha \in \Sigma$ then $s(\alpha) = F$, so $s(\alpha \to \beta) = T$. The last case is $\alpha \in \Sigma$. Applying modus ponens, $\Sigma \vdash \beta$, so $\beta \in \Sigma$ and $s(\beta) = T$ by the inductive hypothesis, so $s(\alpha \to \beta) = T$. It follows by induction that s satisfies Σ.

Now assume that s' is another assignment that satisfies Σ and let p be an arbitrary propositional variable. If $p \in \Sigma$ then $s'(p) = T$, but also $s(p) = T$. If $p \notin \Sigma$ then $\neg p \in \Sigma$ so $s'(\neg p) = T$ and $s'(p) = F$, and we also have $s(p) = F$. Since s and s' agree on every propositional variable, they must be the same function, so that s is unique. ☐

Theorem 1.3 (Completeness Theorem)**.** *Let Σ be a set of p-formulas and ϕ be a p-formula. Then, $\Sigma \models \phi \implies \Sigma \vdash \phi$.*

Proof. If we can show that any consistent set of p-formulas is satisfiable the result follows by Lemma 1.8, so let Δ be one such set. Since $\mathcal{L}_P$ only has countably many symbols and every $\mathcal{L}_P$ string is finite, there are only countably many p-formulas. Thus, we can fix a list of the p-formulas as follows:

$$\phi_0, \phi_1, \phi_2, \ldots$$

This can be done so that every p-formula occurs in the list exactly once.

Let $\Sigma_0 := \Delta$ and define Σ_{n+1} recursively as

$$\Sigma_{n+1} := \begin{cases} \Sigma_n \cup \{\phi_n\} & \Sigma_n \vdash \phi_n \\ \Sigma_n \cup \{\neg\phi_n\} & \Sigma_n \nvdash \phi_n \end{cases}$$

We argue by induction that each Σ_n is consistent. The base case follows from the assumption that Δ is consistent, so assume that Σ_n is consistent. If $\Sigma_n \nvdash \phi_n$, $\Sigma_{n+1} = \Sigma_n \cup \{\neg\phi_n\}$ is consistent by Lemma 1.7. The other case is $\Sigma \vdash \phi_n$, but then $\Sigma_{n+1} = \Sigma_n \cup \{\phi_n\}$ is clearly consistent.

Define $\Sigma := \bigcup_{n\in\mathbf{N}} \Sigma_n$. Clearly $\Sigma_0 = \Delta \subseteq \Sigma$. Assume for contradiction that Σ is inconsistent and fix some deduction D of $\perp$ from Σ. Since D is finite, there are only finitely many assumptions used (i.e elements of Σ) used in D, so that there is some $N \in \mathbf{N}$ such that Σ_N includes all of those assumptions. Thus, $\Sigma_N \vdash \perp$. But we have already shown that Σ_N must be consistent, so we have our contradiction.

Also, given any p-formula ψ, there is some natural n such that $\phi_n :\equiv \psi$, so one of ψ or $\neg\psi$ are in Σ. Since Σ is consistent, it cannot be the case that both $\psi, \neg\psi \in \Sigma$, so Σ is complete. By Lemma 1.11, there is an assignment s that satisfies Σ. Since $\Delta \subseteq \Sigma$, s also satisfies Δ, thus Δ is satisfiable and we are done. □

Definition 1.11. *A set Γ of p-formulas is finitely satisfiable if and only if all of its finite subsets are satisfiable.*

Theorem 1.4 (Compactness Theorem)**.** *A set Γ of p-formulas is satisfiable if and only if it is finitely satisfiable.*

Proof. The forward direction is trivial, so we focus on the converse. Assume that Γ is not satisfiable. It follows vacuously that $\Gamma \models \perp$, so $\Gamma \vdash \perp$ by the Completeness Theorem. Since every proof is finite, there must be some $\Gamma_0 \subseteq \Gamma$ such that $\Gamma_0 \vdash \perp$. By the Soundness Theorem, $\Gamma_0 \models \perp$, therefore it is not satisfiable. □

2 First-order Logic

Definition 2.1. *We say that Σ is **an $\mathcal{L}$-theory** if and only if Σ is a set of $\mathcal{L}$-sentences for some language $\mathcal{L}$.*

Lemma 2.1. *Let s be a variable assignment function, u, t be $\mathcal{L}$-terms and x be variable. Then, $\bar{s}[x|\bar{s}(t)](u) = \bar{s}(u^x_t)$.*

Proof. We induct on the complexity of u. First, assume that u is a constant symbol. Then, $u^x_t = u$, so $\bar{s}(u^x_t) = u^{\mathfrak{A}} = \bar{s}[x|\bar{s}(t)](u)$. The case where u is a variable other than x is trivial, so assume $u :\equiv x$. Then, $u^x_t = t$, thus $\bar{s}(u^x_t) = \bar{s}(t)$. By the definition of $s[x|\bar{s}(t)]$, $s[x|\bar{s}(t)](u) = s[x|\bar{s}(t)](x) = \bar{s}(t)$, as we wanted to show.

Next, assume that $u :\equiv ft_1 \dots t_n$, where f is a n-ary function symbol and $t_1, \dots t_n$ are terms. Then, $u_t^x = f(t_1)_t^x \dots (t_n)_t^x$. Applying the inductive hypothesis, $\bar{s}(u) = f^{\mathfrak{A}}(\bar{s}((t_1)_t^x), \dots, \bar{s}((t_n)_t^x)) = f^{\mathfrak{A}}(\bar{s}[x|\bar{s}(t)](t_1), \dots, \bar{s}[x|\bar{s}(t)](t_n)) = \bar{s}[x|\bar{s}(t)](u)$. $\square$

Lemma 2.2. *Let $\mathfrak{A}$ be an $\mathcal{L}$-structure, u be an $\mathcal{L}$-formula, x be a variable, t be an $\mathcal{L}$-term, and x be substitutable for t in u. Then, if s is a variable assignment function, $\mathfrak{A} \models u[s(x|\bar{s}(t))] \iff \mathfrak{A} \models u_t^x[s]$.*

Proof. We will use induction on the structure of u. First, assume $u :\equiv t_1 = t_2$, where t_1, t_2 are $\mathcal{L}$-terms. Then, $\mathfrak{A} \models t_1 = t_2[s(x|\bar{s}(t))] \iff \bar{s}[x|\bar{s}(t)](t_1) = \bar{s}[x|\bar{s}(t)](t_2)$. By Lemma 2.1, $\bar{s}[x|\bar{s}(t)](t_1) = \bar{s}[x|\bar{s}(t)](t_2) \iff \bar{s}((t_1)_t^x) = \bar{s}((t_2)_t^x)$. Since $(t_1)_t^x = (t_2)_t^x :\equiv u_t^x$, $\mathfrak{A} \models t_1 = t_2[s(x|\bar{s}(t))] \iff \mathfrak{A} \models u_t^x[s]$.

Now, assume $u :\equiv Rt_1 \dots t_n$, where R is an n-ary relation symbol. Then, $\mathfrak{A} \models u[s(x|\bar{s}(t))] \iff (\bar{s}[x|\bar{s}(t)](t_1), \dots, \bar{s}[x|\bar{s}(t)](t_n)) \in R^{\mathfrak{A}}$. By Lemma 2.1, $(\bar{s}[x|\bar{s}(t)](t_1), \dots, \bar{s}[x|\bar{s}(t)](t_n)) = (\bar{s}((t_1)_t^x), \dots, \bar{s}((t_n)_t^x))$. But $\mathfrak{A} \models u_t^x[s] \iff (s((t_1)_t^x), \dots, s((t_n)_t^x)) \in R^{\mathfrak{A}}$, so the result follows.

Next, assume that $u :\equiv (\neg\alpha)$. Applying the inductive hypothesis, $\mathfrak{A} \models (\neg\alpha)_t^x[s] \iff \mathfrak{A} \not\models \alpha_t^x[s] \iff \mathfrak{A} \not\models \alpha[s(x|\bar{s}(t))] \iff \mathfrak{A} \models u[s(x|\bar{s}(t))]$. The case where $u :\equiv (\alpha \vee \beta)$ is similar.

The last case is $u :\equiv (\forall y\phi)$. First, assume that $y :\equiv x$. Then, $u_t^x :\equiv u$, and s and $s[x|\bar{s}(t)]$ agree on all the free variables of u (x is not free in u), so $\mathfrak{A} \models u_t^x[s] \iff \mathfrak{A} \models u[s] \iff \mathfrak{A} \models u[s(x|\bar{s}(t))]$. Next, assume that y is not the same variable as x. If x is not free in ϕ, then $u_t^x :\equiv u$ and the argument is very similar to the previous. If x is free in ϕ, then y does not occur in t, since x is substitutable for t in ϕ. Then, $\mathfrak{A} \models (\forall y\phi_t^x)[s] \iff \mathfrak{A} \models \phi_t^x[s(y|a)]$ for every $a \in A$, where A is the universe of $\mathfrak{A}$. By the inductive hypothesis, $\mathfrak{A} \models \phi_t^x[s(y|a)] \iff \mathfrak{A} \models \phi[s(y|a)(x|\bar{s}[y|a](t))]$. Since y does not occur in t, it follows that $\bar{s}[y|a](t) = \bar{s}(t)$. It is also not difficult to verify that $s[y|a][x|\bar{s}(t)] = s[x|\bar{s}(t)][y|a]$, so $\mathfrak{A} \models \phi[s(y|a)(x|\bar{s}[y|a](t))] \iff \mathfrak{A} \models \phi[s(x|\bar{s}(t))(y|a)]$. But $\mathfrak{A} \models (\forall y\phi)[s(x|\bar{s}(t))] \iff \mathfrak{A} \models \phi[s(x|\bar{s}(t))(y|a)]$, for every $a \in A$, so the argument is complete. $\square$

Theorem 2.1. *Let Σ be a set of $\mathcal{L}$-formulas and ϕ be an $\mathcal{L}$-formula and θ be a sentence. Then, $\Sigma \vdash (\theta \to \phi)$ if and only if $\Sigma \cup \theta \vdash \phi$.*

Proof. For the forward direction, assume $\Sigma \vdash (\theta \to \phi)$. Then, $\Sigma \cup \theta \vdash (\theta \to \phi)$, but $\Sigma \cup \theta \vdash \theta$, so we can apply a rule of inference of type PC to conclude that $\Sigma \cup \theta \vdash \phi$.

For the converse direction, assume $\Sigma \cup \theta \vdash \phi$. We will use induction on the shortest length of deduction of ϕ from $\Sigma \cup \theta$. For the base case, ϕ is either an axiom or ϕ_P is a tautology, therefore $\Sigma \vdash (\theta \to \phi)$ trivially.

Assume inductively that if the shortest length of deduction of η from $\Sigma \cup \theta$ is less than or equal to n then $\Sigma \vdash (\theta \to \eta)$ and assume that the shortest length of deduction of ϕ is $n + 1$. ϕ cannot be an axiom or θ, as if that were the case its shortest length of deduction from $\Sigma \cup \theta$ would be 1. So there is some rule of inference (Γ, ϕ), where $\Gamma = \{\gamma_1, \dots, \gamma_k\}$ with $k < n + 1$. If (Γ, ϕ) is of type

PC, then, by the inductive hypothesis, $\Sigma \vdash (\theta \to (\gamma_1 \wedge \cdots \wedge \gamma_k))$. But, since ϕ is a propositional consequence of Γ, it follows that $\Sigma \vdash ((\gamma_1 \wedge \cdots \wedge \gamma_k) \to \phi)$. Applying a rule of inference of type PC, we can conclude that $\Sigma \vdash (\theta \to \phi)$.

Next, assume that (Γ, ϕ) is of type QR, $\Gamma = \{\psi \to \alpha\}$ and $\phi :\equiv \psi \to (\forall x \alpha)$, where the variable x is not free in ψ. Since $\Sigma \cup \theta \vdash (\psi \to \alpha)$, the inductive hypothesis guarantees that $\Sigma \vdash (\theta \to (\psi \to \alpha))$, therefore $\Sigma \vdash ((\theta \wedge \psi) \to \alpha)$. Since θ is a sentence, x is not free in $\theta \wedge \psi$, so we can apply a rule of inference of the same kind to infer that $\Sigma \vdash ((\theta \wedge \psi) \to \forall x \alpha)$ thus $\Sigma \vdash (\theta \to \phi)$. The remaining case is similar. $\square$

Corollary 2.1.1. *Let Σ be a set of $\mathcal{L}$-formulas and η be a sentence. Define $\bot :\equiv ((\forall x)x = x \wedge (\forall x)\neg x = x)$. Then, $\Sigma \vdash \eta$ if and only if $(\Sigma \cup \neg\eta) \vdash \bot$.*

Proof. Assume $\Sigma \vdash \eta$. Clearly, $\Sigma \cup \neg\eta \vdash \eta$ and $\Sigma \cup \neg\eta \vdash \neg\eta$, thus $\Sigma \cup \neg\eta \vdash (\eta \wedge \neg\eta)$. Since every formula is a propositional consequence of $(\eta \wedge \neg\eta)$, $\Sigma \cup \neg\eta \vdash \bot$.

For the converse direction, assume $\Sigma \cup \neg\eta \vdash \bot$. By the Deduction Theorem, $\Sigma \vdash (\neg\eta \to \bot)$. Then, $\Sigma \vdash (\neg\bot \to \eta)$. Since $(\neg\bot)_P$ is a tautology, $\Sigma \vdash \eta$. $\square$

Lemma 2.3. *Let Σ be a set of $\mathcal{L}$-formulas and ϕ be an $\mathcal{L}$-formula. Then, $\Sigma \vdash \phi$ if and only if $\Sigma \vdash (\forall x \phi)$.*

Proof. For the forward direction, assume that $\Sigma \vdash \phi$. Define $\top := (\forall x)(x = x) \vee \neg(\forall x)(x = x)$ and notice that $\top$ is a sentence where $\top_P$ is a tautology. Then, $\Sigma \vdash (\top \to \phi)$. Since x is not free in $\top$, it follows from a rule of inference of type QR that $\Sigma \vdash (\top \to (\forall x \phi))$. But $\top_P$ is a tautology, so $\Sigma \vdash \top$, therefore $\Sigma \vdash (\forall x \phi)$.

The converse direction follows straightforwardly from an application of the axiom Q1. $\square$

Lemma 2.4. *Let Σ be a set of $\mathcal{L}$-formulas and let Σ' be a set formed by adding or removing a universal quantifier from one of the formulas in Σ. Then, if ϕ is an $\mathcal{L}$-formula, $\Sigma \vdash \phi$ if and only if $\Sigma' \vdash \phi$.*

Proof. For the forward direction assume that $\Sigma \vdash \phi$. Let $\top$ be an abbreviation for $(\forall x)(x = x) \vee \neg(\forall x)(x = x)$ and notice that $\top$ is a sentence, and that $\top_P$ is a tautology. Then, since $\Sigma \vdash \phi$, $\Sigma \vdash (\top \to \phi)$. By the first rule of inference of type QR, it follows that $\Sigma \vdash (\top \to (\forall x)(\phi))$ but sigma clearly proves $\top$, thus $\Sigma \vdash (\forall x)(\phi)$.

For the converse direction, assume that $\Sigma \vdash (\forall x)(\phi)$. By the Quantifier Axiom of type $Q1$, $\Sigma \vdash (\forall x \phi) \to \phi^x_x$, since x is always subsitutable for x in ϕ. It follows easily from a rule of inference of type PC that $\Sigma \vdash \phi$. $\square$

Lemma 2.5. *Let Σ be a consistent set of $\mathcal{L}$-sentences. Define $\mathcal{L}' := \mathcal{L} \cup \{c^m_n : m, n \in \mathbf{N}\}$, where each c^n_m is a constant symbol, called a Henkin constant. Then, Σ is also consistent as a set of $\mathcal{L}'$ sentences.*

Lemma 2.6. *Let Σ be a consistent set of $\mathcal{L}'$-sentences, where $\mathcal{L}'$ is the extension by constants of $\mathcal{L}$.*

Theorem 2.2 (Completeness Theorem)**.** *Let $\mathcal{L}$ be a first order language. If Σ is a set of $\mathcal{L}$-formulas and ϕ is an $\mathcal{L}$-formula, then $\Sigma \models \phi \implies \Sigma \vdash \phi$.*

Proof. It follows from Lemma 2.3 that $\Sigma \vdash \phi$ if and only if there is a deduction from Σ of the universal closure of ϕ, so we can assume that ϕ is an $\mathcal{L}$-sentence. Also, by Lemma 2.4, Σ can be assumed to be a set of $\mathcal{L}$-sentences.

Furthermore, we argue that it is sufficient to prove that every consistent set of $\mathcal{L}$-sentences has a model. To see that, let ϕ be an $\mathcal{L}$-sentence and Σ be a set of $\mathcal{L}$-sentences such that $\Sigma \models \phi$. If Σ is inconsistent, then there is a deduction of any $\mathcal{L}$-formula from Σ, including ϕ, so we may assume that Σ is consistent. Assume for contradiction that $\Sigma \nvdash \phi$. By Corollary 2.1.1, this means that $\Sigma \cup \neg\phi$ is consistent. By our assumption, this means that $\Sigma \cup \neg\phi$ has a model $\mathfrak{A}$. By the definition of model, we must have $\mathfrak{A} \models \neg\phi$, so $\mathfrak{A} \not\models \phi$. But $\Sigma \models \phi$, and $\mathfrak{A}$ is clearly a model of Σ, therefore $\mathfrak{A} \models \phi$. This is a contradiction, so it must be the case that $\Sigma \vdash \phi$.

Now, assume that Σ is a consistent set of $\mathcal{L}$-sentences. We have to show that Σ has a model, and to do that we will explicitly construct one.

Define $\mathcal{L}_0 := \mathcal{L}$. Now assume inductively that $\mathcal{L}_k$ is defined for some $k \in \mathbf{N}$. We define $\mathcal{L}_{k+1}$ as $\mathcal{L}_k \cup \{c_1^{k+1}, c_2^{k+1}, \dots\}$ where c_m^n is a constant symbol for every $m, n \in \mathbf{N}$. Finally, define the extended language $\mathcal{L}'$ by

$$\mathcal{L}' = \bigcup \{\mathcal{L}_k : k \in \mathbf{N}\}.$$

Next, let $\Sigma_0 := \Sigma$ and assume inductively that Σ_k is defined for some $k \in \mathbf{N}$. Since $\mathcal{L}'$ is a countable union of countable sets, it is countable, so we can list its sentences of the form $(\exists y \theta)$ like so:

$$(\exists y \theta_1), (\exists y \theta_2), (\exists y \theta_3), \dots,$$

and this can be done in such a way that every sentence of the desired form occurs in the list exactly once. Now, use this list to construct the following set:

$$H_k := \{(\exists y \theta_n) \to \theta^y_{c_n^k} : n \in \mathbf{N}\}.$$

The H_k's are collectively called the Henkin Axioms.

We define $\Sigma_{k+1} := \Sigma_k \cup H_k$. Also, define

$$\widetilde{\Sigma} := \bigcup \{\Sigma_k : k \in \mathbf{N}\}.$$

It is easy to see that each Σ_k is a set of sentences, and so is $\widetilde{\Sigma}$. By Lemma (), Σ_k is consistent for every $k \in \mathbf{N}$. Assume for contradiction that $\widetilde{\Sigma}$ is inconsistent and let D be the shortest deduction of $\perp$ from $\widetilde{\Sigma}$. Since every deduction is finite, there is some $n \in \mathbf{N}$ that is big enough such that D is also a deduction of $\perp$ from Σ_n, which contradicts the fact that Σ_n is consistent. Thus, $\widetilde{\Sigma}$ is consistent.

Since $\mathcal{L}'$ is a countable union of countable sets, it is countable, so we can list its sentences like so:

$$\theta_1, \theta_2, \theta_3, \dots.$$

Furthermore, we can make it so that every sentence of $\mathcal{L}'$ occurs in the list exactly once. For the next step, let $\Sigma^0 := \widetilde{\Sigma}$ and assume inductively that Σ^k is defined for some $k \in \mathbf{N}$. Define

$$\Sigma^{k+1} := \begin{cases} \Sigma^k \cup \{\theta_k\} & \Sigma^k \cup \{\theta_k\} \text{ is consistent,} \\ \Sigma^k \cup \{\neg\theta_k\} & \text{otherwise.} \end{cases}$$

Finally, define

$$\Sigma' = \bigcup \{\Sigma^k : k \in \mathbf{N}\}.$$

Now we show that each Σ^k is consistent. We have already shown that Σ^0 is consistent, so assume inductively that Σ^k is consistent. If $\Sigma^k \cup \{\theta_k\}$ is consistent, then the claim is trivial, so assume that this is not the case, so that $\Sigma^k \cup \theta_k \vdash \bot$. By Corollary 2.1.1, $\Sigma^k \vdash \neg\theta_k$. Assume for contradiction that $\Sigma^k \cup \neg\theta_k \vdash \bot$. Using the same corollary, we can conclude that $\Sigma^k \vdash \theta_k$, thus $\Sigma^k \vdash (\theta_k \wedge \neg\theta_k)$, which means Σ^k is inconsistent, and that is a contradiction. Thus, for every $k \in \mathbf{N}$, Σ^k is consistent, and we can apply the same argument we used to show that $\widetilde{\Sigma}$ is consistent to show that Σ' is consistent.

Next, we show that if η is a sentence, then $\eta \in \Sigma' \iff \Sigma' \vdash \eta$. The forward direction is trivial, so assume that $\Sigma' \vdash \eta$. Referring back to our list of $\mathcal{L}'$ sentences, there is some $k \in \mathbf{N}$ such that $\eta :\equiv \theta_k$. Let D be the shortest deduction of η from Σ'. Since D is finite, there is some $n \in \mathbf{N}$ such that D is also a deduction of η from Σ^n and $\theta_k \in \Sigma^n$. But $\theta_k \in \Sigma^n$ implies $\eta \in \Sigma'$. We say that Σ' is a maximally consistent set of sentences, since it is consistent and every $\mathcal{L}'$ sentence or its negation is in Σ'. It also follows that $\eta \in \Sigma' \iff \Sigma \models \eta$. Again, we only need to focus on the converse direction, so assume that $\Sigma' \models \eta$. Since η is a sentence either η or $\neg\eta$ is an element of Σ', so assume for contradiction that $\neg\eta \in \Sigma'$

Consider the set $S := \{t : t \text{ is a variable free term of } \mathcal{L}'\}$. Let $a, b \in S$ be arbitrary. We define the equivalence relation $\sim$ on S by

$$a \sim b \iff a = b \in \Sigma'.$$

We define A as the set of all equivalence classes of $\sim$ on S, and let A be the universe of the structure $\mathfrak{A}$ (this will be a model of Σ'). In other words, $A = \{[x] : x \in S\}$. For each constant symbol $c \in \mathcal{L}'$, we define $c^{\mathfrak{A}}$ as $[c]$.

Let f be a n-ary function symbol from $\mathcal{L}'$. To define $f^{\mathfrak{A}} : A^n \to A$, let $([t_1], [t_2], \ldots, [t_n]) \in A^n$ be arbitrary. Then, $f^{\mathfrak{A}}([t_1], [t_2], \ldots, [t_n]) := [ft_1t_2 \ldots t_n]$.

Let R be a n-ary relation symbol from $\mathcal{L}'$. To define $R^{\mathfrak{A}} \subseteq A^n$, let $([t_1], [t_2], \ldots, [t_n]) \in A^n$ be arbitrary. Then, $([t_1], [t_2], \ldots, [t_n]) \in R^{\mathfrak{A}} \iff Rt_1t_2 \ldots t_n \in \Sigma'$.

It is not clear at first that $f^{\mathfrak{A}}$ is well-defined. To see that it is, let $[t_1] = [t_2]$. We need to show that $f^{\mathfrak{A}}([t_1]) = f^{\mathfrak{A}}([t_2])$ (we are assuming that f is unary, but the argument is very similar for other arities). By the definition of $\sim$, $t_1 = t_2 \in \Sigma'$, so $\Sigma' \vdash t_1 = t_2$. Then, it follows easily from the logical axioms that $\Sigma' \vdash ft_1 = ft_2$, thus $ft_1 = ft_2 \in \Sigma'$ so $ft_1 \sim ft_2$. Thus, $[ft_1] = [ft_2]$, which means that $f^{\mathfrak{A}}([t_1]) = f^{\mathfrak{A}}([t_2])$, as we wanted to show. A similar argument shows that relations are also well-defined.

Let σ be a $\mathcal{L}'$-sentence. We will show that $\sigma \in \Sigma' \iff \mathfrak{A} \models \sigma$, which implies $\mathfrak{A} \models \Sigma'$. Let s be an arbitrary variable assignment function. Throughout the argument, we will use the fact that if t is a variable free term, then $\bar{s}(t) = [t]$, which very easy to verify. We will use induction on the structure of σ to show that $\mathfrak{A} \models \sigma[s] \iff \sigma \in \Sigma'$. First consider the case where $\sigma :\equiv Rt_1 \dots t_n$, where R is a n-ary relation symbol. Then, $Rt_1 \dots t_n \in \Sigma' \iff ([t_1], \dots [t_n]) \in R^{\mathfrak{A}} \iff (\bar{s}(t_1), \dots, \bar{s}(t_n)) \in R^{\mathfrak{A}} \iff \mathfrak{A} \models \sigma[s]$. The case where $\sigma :\equiv t_1 = t_2$ is similar. Now, assume that $\sigma :\equiv \neg\alpha$ where α is a $\mathcal{L}'$-sentence. Then, $\mathfrak{A} \models \neg\alpha[s] \iff \mathfrak{A} \not\models \alpha[s] \iff \alpha \notin \Sigma' \iff \neg\alpha \in \Sigma'$. The case where $\sigma :\equiv (\alpha \vee \beta)$ is similar.

Next, assume $\sigma :\equiv (\forall y)(\psi)$. Since σ is a sentence, the only possible free variable of ψ is y. We will show each direction separately, so first assume that $\sigma \in \Sigma'$. We need to show that given some arbitrary $[t] \in A$, $\mathfrak{A} \models \psi[s(y|[t])]$. So let $[t] \in A$ be arbitrary, then, $(\forall y \psi) \to \psi^y_t$ is an axiom of type Q1. Thus, $\Sigma' \vdash ((\forall y \psi) \to \psi^y_t)$. But $\Sigma' \vdash (\forall y \psi)$, so $\Sigma' \vdash \psi^y_t$. By the inductive hypothesis, $\mathfrak{A} \models \psi^y_t[s]$, so we can apply Lemma 2.2 to conclude that $\mathfrak{A} \models \psi[s(y|[t])]$.

For the converse direction, assume that $\mathfrak{A} \models \sigma$. Since $\neg\psi$ is a sentence, there is some constant symbol c such that $(\exists y \neg\psi) \to \neg\psi^y_c$ is Henkin Axiom. Thus, $\Sigma' \vdash ((\exists y \neg\psi) \to \neg\psi^y_c)$, therefore $\Sigma' \vdash (\psi^y_c \to (\forall y \psi))$. Since c is a variable free term of $\mathcal{L}'$, $[c] \in A$, thus $\mathfrak{A} \models \psi[s(y|[c])]$ and we can apply Lemma 2.2 again to see that $\mathfrak{A} \models \psi^y_c[s]$. By the inductive Hypothesis, $\psi^y_c \in \Sigma'$, thus $\Sigma' \vdash \psi^y_c$. Applying a rule of inference of type PC, $\Sigma' \vdash (\forall y \psi)$.

Thus, $\mathfrak{A} \models \Sigma'$. Since $\Sigma \subseteq \Sigma'$, it follows that $\mathfrak{A} \models \Sigma$ when Σ is viewed as a set of $\mathcal{L}'$ sentences. Consider the $\mathcal{L}$-structure $\mathfrak{A}\restriction_{\mathcal{L}}$, whose universe is the same of $\mathfrak{A}$ and where the function, relation and constant symbols of $\mathcal{L}$ have the same interpretation as before. We call $\mathfrak{A}\restriction_{\mathcal{L}}$ the restriction of $\mathfrak{A}$ to $\mathcal{L}$, or $\mathfrak{A}$ restricted to $\mathcal{L}$. Now, we show that $\mathfrak{A}\restriction_{\mathcal{L}} \models \Sigma$, when Σ is viewed as a set of $\mathcal{L}$-sentences, through induction on the structure of the sentences of Σ. To do that, we will show that $\mathfrak{A} \models \sigma \iff \mathfrak{A}\restriction_{\mathcal{L}} \models \sigma$.

First, assume that $\sigma :\equiv t_1 = t_2$, where t_1 and t_2 are variable free $\mathcal{L}$-terms. Then, $\mathfrak{A}\restriction_{\mathcal{L}} \models t_1 = t_2[s] \iff \bar{s}(t_1) = \bar{s}(t_2)$. It follows that $\mathfrak{A} \models \sigma[s] \iff \bar{s}(t_1) = \bar{s}(t_2) \iff \mathfrak{A}\restriction_{\mathcal{L}} \models \sigma[s]$. Now, assume $\sigma :\equiv Rt_1 \dots t_n$, where $t_1, \dots, t_n$ are variable free $\mathcal{L}$-terms and R is an n-ary relation symbol from $\mathcal{L}$. Then, $\mathfrak{A}\restriction_{\mathcal{L}} \models \sigma[s] \iff (\bar{s}(t_1), \dots, \bar{s}(t_n)) \in R^{\mathfrak{A}\restriction_{\mathcal{L}}} \iff (\bar{s}(t_1), \dots, \bar{s}(t_n)) \in R^{\mathfrak{A}} \iff \mathfrak{A} \models \sigma[s]$, since $R^{\mathfrak{A}\restriction_{\mathcal{L}}} = R^{\mathfrak{A}}$.

Next, assume $\sigma :\equiv (\neg\alpha)$. Then $\mathfrak{A}\restriction_{\mathcal{L}} \models \sigma \iff \mathfrak{A}\restriction_{\mathcal{L}} \not\models \alpha \iff \mathfrak{A} \not\models \alpha \iff \mathfrak{A} \models \sigma$. The case where $\sigma :\equiv (\alpha \vee \beta)$ is similar.

Now assume that $\sigma :\equiv (\forall y \psi)$. Then $\mathfrak{A}\restriction_{\mathcal{L}} \models \sigma[s] \iff \mathfrak{A}\restriction_{\mathcal{L}} \models \psi[s(y|a)]$ for every $a \in A$. So let $a \in A$ be arbitrary. By the inductive hypothesis, $\mathfrak{A}\restriction_{\mathcal{L}} \models \psi[s(y|a)] \iff \mathfrak{A} \models \psi[s(y|a)]$, and the induction is complete.

It follows that for every $\mathcal{L}$-sentence σ, $\sigma \in \Sigma' \iff \mathfrak{A}\restriction_{\mathcal{L}} \models \sigma$. Since $\Sigma \subseteq \Sigma'$, $\mathfrak{A}\restriction_{\mathcal{L}} \models \Sigma$. $\square$

Definition 2.2. *Let $\mathfrak{A}$ and $\mathfrak{B}$ be $\mathcal{L}$-structures. We say that $\mathfrak{A}$ is a substructure* *of $\mathfrak{B}$, and write $\mathfrak{A} \subseteq \mathfrak{B}$ if and only if*

1. *$A \subseteq B$,*
2. *For every constant symbol $c \in \mathcal{L}$, $c^{\mathfrak{A}} = c^{\mathfrak{B}}$,*
3. *For every n-ary function symbol f, $f^{\mathfrak{A}} = f^{\mathfrak{B}}\restriction_{\mathcal{A}\backslash}$, and for every n-ary relation symbol R, $R^{\mathfrak{A}} = R^{\mathfrak{B}} \cap A^n$.*

Definition 2.3. *Let $\mathfrak{A}$ and $\mathfrak{B}$ be $\mathcal{L}$-structures such that $\mathfrak{A} \subseteq \mathfrak{B}$. We say that $\mathfrak{A}$ is an elementary substructure of $\mathfrak{B}$ and write $\mathfrak{A} \prec \mathfrak{B}$ if and only if for every $\mathcal{L}$-formula ϕ and every assignment $s : Vars \to A$,*

$$\mathfrak{A} \models \phi[s] \iff \mathfrak{B} \models \phi[s].$$

Lemma 2.7. *Let $\mathfrak{A}$ and $\mathfrak{B}$ be $\mathcal{L}$-structures such that $\mathfrak{A} \subseteq \mathfrak{B}$. Assume that for every $s : Vars \to A$ and every $\mathcal{L}$-formula α such that $\mathfrak{B} \models \exists x\alpha[s]$ there is an $a \in A$ such that $\mathfrak{B} \models \alpha[s[x|a]]$. Then $\mathfrak{A} \prec \mathfrak{B}$.*

Proof. Assuming the hypothesis in the lemma, we show that given any assignment $s : Vars \to A$ and any $\mathcal{L}$-formula ϕ, $\mathfrak{A} \models \phi[s] \iff \mathfrak{B} \models \phi[s]$ by induction on the complexity of ϕ.

We focus only on the case where $\phi :\equiv \exists x\alpha$, as the other cases are straightforward. Assume first that $\mathfrak{A} \models \phi[s]$. This means that there is some $a \in A$ such that $\mathfrak{A} \models \alpha[s[x|a]]$. Since we have removed a quantifier, the inductive hypothesis guarantees that $\mathfrak{B} \models \alpha[s[x|a]]$. Since $A \subseteq B$, it follows that $a \in B$, so $\mathfrak{B} \models \exists x\alpha[s]$. Now assume that $\mathfrak{B} \models \exists x\alpha[s]$. By the hypothesis in the lemma, there is some $a \in A$ such that $\mathfrak{B} \models \alpha[s[x|a]]$. It follows similarly from the inductive hypothesis that $\mathfrak{A} \models \exists x\alpha[s]$. □

Theorem 2.3 (Downward Löwenheim–Skolem theorem)**.** *Let $\mathcal{L}$ be a countable language and $\mathfrak{B}$ be an $\mathcal{L}$-structure with infinite universe. Then $\mathfrak{B}$ has a countable elementary substructure.*

Proof. If B is countable then $\mathfrak{B}$ is its own countable elementary substructure, so assume that B is uncountable and fix some countable $A_0 \subseteq B$. Consider some arbitrary formula α and some eventually constant $s : Vars \to A_0$ such that $\mathfrak{B} \models \exists x\alpha[s]$. Then, there is some constant $a_{\alpha,s} \in B$ such that $\mathfrak{B} \models \alpha[s[x|a_{\alpha,s}]$. Since there are only countably many $\mathcal{L}$-formulas and only countably many eventually constant assignments into A_0, the set

$$A_1 = A_0 \cup \{a_{\alpha,s} : \mathfrak{B} \models \alpha[s[x|a_{\alpha,s}]]\}$$

is countable. We can construct A_n iteratively for all $n \in \mathbf{N}$, and take $A := \cup\{A_n : n \in \mathbf{N}\}$. Since A is a countable union of countable sets it is also countable. To show that the structure $\mathfrak{A}$ with universe A is a substructure of $\mathfrak{B}$, we need to show that A is closed under the functions of $\mathfrak{B}$.

Let f be a function symbol, a be an element of A and $b = f^{\mathfrak{B}}(a)$. We need to show that $b \in A$. Fix n large enough so that $a \in A_n$ and let ϕ be the formula $\exists y(y = f(x))$. Choose some eventually constant $s : Vars \to A_n$ such that $s(x) = a$. But then

$$\begin{aligned}\mathfrak{B} \models \phi[s] &\iff \mathfrak{B} \models y = f(x)[s[x|d]] \text{ for some } d \in B \\ &\iff d = f^{\mathfrak{B}}(a) = b \text{ for some } d \in B.\end{aligned}$$

So, since $b \in B$, $\mathfrak{B} \models \phi[s]$. By the construction of A_{n+1}, this means that there is some $a_{y=f(x),s}$ in B such such that $a_{y=f(x),s} \in A_{n+1}$. It is easy to see by the equivalencies above that $\mathfrak{B} \models y = f(x)[s[x|d]] \iff b = d$, which means $b = a_{y=f(x),s} \in A_{n+1}$, therefore $b \in A$.

Now we use Lemma 2.7 to show that $\mathfrak{A} \prec \mathfrak{B}$. So let $s : Vars \to A$ and α be such that $\mathfrak{B} \models \exists x\alpha[s]$. Then there is some eventually constant s' that agrees with s on the free variables of α so that $\mathfrak{B} \models \exists x\alpha[s] \iff \mathfrak{B} \models \exists x\alpha[s']$. Since s' is eventually constant, we can fix n large enough so that s' is also a function into A_n. Since $\mathfrak{B} \models \exists x\alpha[s']$, there is some $b \in B$ such that $\mathfrak{B} \models \alpha[s'[x|b]]$ and $b \in A_{n+1}$ by construction. But $s'[x|b]$ and s agree on the free variables of $\exists x\alpha$, so $\mathfrak{B} \models \alpha[s[x|b]]$ and we are done. $\square$

Definition 2.4. *Let $\mathcal{L}$ be a first order language and ϕ be an $\mathcal{L}$-formula where x is the only free variable in ϕ. Define $\mathcal{L}' := \mathcal{L} \cup \{c\}$ where c is a constant symbol. Then, given the $\mathcal{L}'$-formula ψ and a variable z, define $\psi^*(z)$ inductively as follows:*

1. *If ψ is an $\mathcal{L}$-formula, $\psi^*(z) :\equiv \psi$.*
2. *If $\psi :\equiv R(t_1)^z_c, \ldots, (t_n)^z_c$ where $t_1, \ldots, t_n$ are $\mathcal{L}$-terms and R is an n-ary relation symbol from $\mathcal{L}$ or equality (and $n = 2$), then $\psi^*(z) :\equiv (\exists z)(Rt_1, \ldots, t_n \wedge \phi^x_z)$.*
3. *If $\psi :\equiv (\neg\alpha)$ where α is an $\mathcal{L}'$-formula, then $\psi^*(z) :\equiv (\forall w)(\alpha^*(z))$.*
4. *If $\psi :\equiv (\alpha \vee \beta)$ where α and β are $\mathcal{L}'$-formulas, then $\psi^*(z) :\equiv (\alpha^*(z) \vee \beta^*(z))$.*
5. *If $\psi :\equiv (\forall w)(\alpha)$ were α is an $\mathcal{L}'$-formula and w is a variable, then*

$$\psi^*(z) :\equiv \begin{cases} (\forall w)(\alpha^*(z)) & \text{if } w \text{ is not } z \\ (\forall w)(\alpha^*(k)) & \text{otherwise,} \end{cases}$$

where k is a variable other than z.

Also, define ψ^* as $\psi^*(z)$.

Lemma 2.8. *Let ϕ be an $\mathcal{L}$-formula. Then $\vdash (\exists x\phi \leftrightarrow \neg\forall x\neg\phi)$*

Theorem 2.4. *Let $\mathcal{L}$, $\mathcal{L}'$, ϕ and x satisfy the conditions of Definition 2.4. Let T be a $\mathcal{L}$-theory such that $T \vdash (\exists x\phi)$ and define the $\mathcal{L}'$-theory $T' := T \cup \{\phi^x_c\}$. If ψ is an $\mathcal{L}'$-formula and χ is an $\mathcal{L}$-formula, the following hold:*

1. *$T \vdash \psi^* \iff T' \vdash \psi$,*
2. *$T \vdash \chi \iff T' \vdash \chi$.*

3 Computability Theory

Definition 3.1. *We define $\mathcal{O} : \emptyset \to \mathbf{N}$ as the function with no arguments that returns 0. $\mathcal{S} : \mathbf{N} \to \mathbf{N}$ is such that $\mathcal{S}(x) = x + 1$ for every $x \in \mathbf{N}$. For each $n \in \mathbf{N}$ we define the projection function $\mathcal{I}_i^n : \mathbf{N}^n \to \mathbf{N}$ for each $1 \leq i \leq n$ as $\mathcal{I}_i^n(x_1, x_2, \ldots, x_i, \ldots, x_n) = x_i$ for all $x_1, \ldots x_n \in \mathbf{N}$.*

The functions above are collectively called the initial functions.

Definition 3.2. *We define the set of computable functions as follows:*

1. *The initial functions are computable.*
2. *If h is a computable function of arity m (possibly 0) and $g_1, \ldots, g_m$ are functions of arity n, then $f(x_1, \ldots x_n) = h(g_1(\underset{\sim}{x}), \ldots, g_m(\underset{\sim}{x}))$.*
3. *If g is a computable function of arity n and h is a computable function of arity $n+2$, then the function f given by*

$$f(\underset{\sim}{x}, 0) = g(\underset{\sim}{x})$$
$$f(\underset{\sim}{x}, y+1) = h(\underset{\sim}{x}, y, f(\underset{\sim}{x}, y))$$

   is a computable function.

4. *If g is a computable function of arity $n+1$, then $f(\underset{\sim}{x}, y) = (\mu i \leq y)(g(\underset{\sim}{x}, i))$ is computable.*

4 Exercises

Exercise 7.3.8.

(a) The statement clearly holds for the initial functions, so assume inductively that $f(\underset{\sim}{x}) = h(g_1(\underset{\sim}{x}), \ldots, g_m(\underset{\sim}{x}))$ where g and h meet the inductive hypothesis. Then $f(\underset{\sim}{x}) \leq g_i(\underset{\sim}{x}) + K_h \leq x_j + K_g + K_h$. The result follows by setting $K := K_g + K_h$.

(b) The result follows easily if f is of rank 0, so assume that it is not. Then $f()$

Exercise 7.3.9.

(a) To show that $A(y, x)$ is a natural number we induct on y. The base case is straightforward, so assume that $A(y, x)$ is defined for all x. To show that $A(y+1, x)$ is defined for all x, we now induct on x. For the base case, $A(y+1, 0) = 2$ by definition, so assume that $A(y+1, x)$ is defined. Then $A(y+1, x+1) = A(y, A(y+1, x))$ by definition. But $A(y+1, x)$ is defined by the second inductive hypothesis therefore $A(y, A(y+1, x))$ is defined by the first inductive hypothesis.

It is easy to see that $A(1, x) = 2x + 2$ and $A(2, x) = 2^{x+2} - 2 > 2^x$ by induction.