

Contents

Preface	1
Introduction	1
Propositional Logic	3
Formal Language	6
Induction & Recursion	15
Induction	15
Recursion	20
Proof Systems	21
Completeness Theorem	29
First-Order Logic	32
Formal Language	33
Proof System	46
Godel's Completeness Theorem	53
What's Next?	65
Higher-Order Logic	65
Godel's Incompleteness Theorem	66

Preface

This introduction to mathematical logic is based on many resources, most prominent of which are:

1. Math 125A by Berkley University's Prof. Antonio Montalban.
2. Gadi Aleksandrowicz's writings on mathematical logic, which can be found in his blog (writings are in Hebrew)
3. A select few lectures from Stanford's CS103 (lectures 9, 10 and 11 from this directory).

And the classical textbooks *A Mathematical Introduction to Logic* by Enderton and *Introduction to Mathematical Logic* by Mendelson.

Introduction

Very generally speaking, mathematics can be thought of as a collection of definitions of mathematical objects along with a sequence of propositions (sometimes referred to as lemmas or theorems or facts

or corollaries, depending on the importance and obviousness of the proposition in the eyes of the author) justified by arguments (proofs) which follow from the definitions, the suppositions of the proposition, previous results and acceptable reasoning. A simplified but representative example of math literature is presented below:

DEFINITION 0.1 *⟨vegan food⟩*

Vegan food is any food product which is in no part derived from animals.

DEFINITION 0.2 *⟨cow cheese⟩*

Cow cheese is a family of food products created from cow milk.

PROPOSITION 1

Cow cheese is not a vegan food.

Proof. Cow cheese is created from cow milk, which is derived from an animal, so by definition it is not a vegan food □

Most readers will find the proof satisfactory (ignoring the fact that “food”, “product”, “derived”, “animals” and “cow” have not been defined, which makes the definitions ambiguous, but this is a toy example so it is intended to be concise at the cost of reduced accuracy), but it is worth considering why we find these definitions, proposition and proof to be acceptable - we would like to investigate our method of proof and our method of reasoning. This is one of the main aims of mathematical logic - to study mathematical reasoning and axiomatize and formalize it, to provide a clear and well defined way to evaluate mathematical arguments.

Note that definitions and propositions (and proofs) are all really just sentences in some language (in this case, English), which has symbols (the alphabet) and grammar which give the expression meaning. Indeed, the expression *1\$@#2 as as in cheese* is simply nonsense in the sense that it is not a valid expression in the English language since it makes use of both symbols outside of the alphabet and makes no sense grammatically. However, spoken language can still be ambiguous, even when expressions are well defined. For example, in the following statement - *if a group multiplication operation commutes, it is abelian*, does *abelian* refer to *group* or to *operation*?

This ambiguity, along with the rather complicated set of rules that govern spoken languages, make such languages unfit for the study of mathematical reasoning. Instead, we translate arguments to a *formal language* where expressions are well-defined and the rules are clear and (ideally) agreeable enough as to not merit a proof. For instance, we can express the proof using formal language as:

PROPOSITION 2 (rephrased)

$$(2 \implies \neg 1)$$

Proof. $((2 \implies \text{cow milk}) \wedge (\text{cow milk} \implies \text{animal product})) \wedge (\text{animal product} \implies \neg 1)$ \square

(Where \wedge is the logical and connective which will be properly defined at a later point)

This formulation of the proof, although still lacking (in the sense that it still needs to be shown that $\text{cow milk} \implies \text{animal product}$), is nicer in the sense that the actual logical argument is clear and easy to verify, and the rigid structure and limited vocabulary of the formal language forces us to fully express our argument, as omissions would be painfully obvious. It also exposes the fact that the proof is via hypothetical syllogism, i.e. we claim that $((A \implies B) \wedge (B \implies C)) \vdash (A \implies C)$, where \vdash is syntactical implication and will be defined at a later point.

Another useful property of this representation is that it is verifiable by a computer - while it is hard for a computer to reason about the validity of an argument written mostly in plain English, it definitely can verify arguments expressed in formal language, which forms the basis for the field of formal verification.

Once we have a proper mathematical definition for an argument, definition and proof, we can treat them as mathematical objects and study them as such, and then produce theorems on mathematical logic and on formal systems (for example, we can discuss what can and cannot be stated in a given logic system, and what can or cannot be proven), most famous of which are Godel's incompleteness theorems. For this reason, mathematical logic is also called *metamathematics*.

In our current discussion, we will briefly introduce and discuss propositional (sentential or zeroth-order) logic, first-order logic and second-order logic systems.

Propositional Logic

Let's get a few basic definitions out of the way first:

DEFINITION 0.1 (boolean domain)

The boolean domain, denoted by \mathbb{B} , is a set of exactly two elements, one of which is interpreted as true and the other as false (or untrue). It is usually denoted by $\{0, 1\}$ or $\{F, T\}$, where $0, F \mapsto \text{false}$ and $1, T \mapsto \text{true}$.

DEFINITION 0.2 ⟨classical logic⟩

Classical logic deals with statements which are either true or false. Classical logic is also called bivalent logic, since there are only two possible values for statements.

This means that in classical logic, “half truths” and similar statements don’t exist.

DEFINITION 0.3 ⟨truth value⟩

The truth value of an object is its relation to truth. In classical logic, the truth value is one of the two elements of the boolean domain \mathbb{B} .

From here on, we will assume classical logic, so we shall always identify *truth values* with *boolean domain*.

DEFINITION 0.4 ⟨proposition⟩

A proposition is a statement that can be assigned a truth value.

EXAMPLE 0.1 ⟨propositional statements⟩

“It will rain tomorrow” is a proposition, since it can be assigned a truth value.

“Will it rain tomorrow?” cannot be assigned a truth value, so it is not a proposition.

“Please grab my coat” is also not a proposition, since it too cannot be assigned a truth value.

DEFINITION 0.5 ⟨valuation⟩

A valuation is an assignment of a truth value to a proposition.

DEFINITION 0.6 ⟨truth function⟩

A truth function is a mapping $f : \mathbb{B}^n \mapsto \mathbb{B}$ for some $n \in \mathbb{N}$, i.e. it maps truth values to a unique truth value. (in non-classical logic, truth functions may have different domains and co-domains)

Since the domain of a truth function is finite (by a combinatorial argument there are exactly $|\mathbb{B}|^n = 2^n$ possible combination), it is possible to enumerate all possible inputs of a truth function and construct a list of all possible outputs, which provides a mapping for each possible input value of the function. It is common to organize this information in a table, which motivates the following definition:

DEFINITION 0.7 ⟨truth table⟩

A truth table of a truth function is a mathematical table of all valid input values and their outcome.

EXAMPLE 0.2 ⟨Logical AND⟩

A logical AND is a binary operator $\wedge : \mathbb{B}^2 \rightarrow \mathbb{B}$, $(A, B) \mapsto A \wedge B$ whose output is true if and only if both its inputs are true, equivalently its truth table is given by

A	B	$A \wedge B$
T	T	T
T	F	F
F	F	F
F	T	F

When dealing with a binary operator, we often write its truth table such that the output is placed in the middle column and represented only by the operator (in this case, \wedge), so it is more common to see the truth table of \wedge written as

A	\wedge	B
T	T	T
T	F	F
F	F	F
F	F	T

DEFINITION 0.8 ⟨propositional logic⟩

Propositional logic (PL) is the study of propositions and their relations under truth functions. Classical propositional logic (CPL) is propositional logic set in classical logic.

Formal Language

Propositions and arguments, which we study in propositional logic, are expressed in a language. In our examples for propositional statements, we used the English language to express our propositions, so it is only natural that we will need to choose a language for the logic system we wish to study, and as we have mentioned earlier, that language will need to be unambiguous, among other desirable properties. The languages we use in math are called *formal languages*. Since languages are the subject of study in the field of linguistics, it is natural to borrow terms from that field when defining a formal language.

Recall from linguistics that broadly speaking, a language consists of 3 parts:

1. A set of symbols - an alphabet
2. A set of syntactic rules - a syntax - which defines rules for valid structuring of sentences
3. Semantics, which deals with the meaning of words

When constructing sentences, we make use of connectives (“go home *and* eat lunch”) and linking verbs (“I *am* hungry”), both of which link linguistic units together (“go home” + “eat lunch” in the former and “I” + “hungry” in the latter, note that they serve different purposes), and punctuation marks. In propositional logic, we do not assign any meaning to verbs, but we are interested in connectives since they can combine separate propositions into a larger proposition (“It is raining *and* I’m cold *and* don’t have my coat”).

DEFINITION 0.9 **⟨alphabet⟩**

An alphabet Σ is a set of symbols. A member of Σ is also called a letter.

DEFINITION 0.10 **⟨string⟩**

A string s over an alphabet Σ is an ordered n -tuple of n letters, i.e. $s \in \Sigma^n$. n is the length of the string. Instead of Σ^n we often write Σ^* to denote that we are interested in strings of various lengths. Strings may be expressed using triangle brackets and commas, e.g. $s = \langle a, b, c, d, e \rangle$ or brackets and commas, e.g. $s = abcde$, if the meaning is unambiguous.

DEFINITION 0.11 **⟨formal language⟩**

A formal language L over an alphabet Σ is a subset of Σ^* , i.e. a set of strings which are called words or expressions or formulas, and are said to be well-formed if they are indeed in L .

Note that the definition of a formal language does not specify how one may form well-formed expressions given an alphabet Σ , although when defining a concrete formal language, such syntactic rules (also called *formation rules*) are usually presented.

EXAMPLE 0.3 ⟨formal language⟩

Given an alphabet $\Sigma = \{1, 2, 3, 4, 5, 6\}$, we define a language L such that L is the set of all words which contain only an odd number of occurrences of odd numbers, where 0 is considered an even amount. Note that this is a formation rule - it is easy to come up with well-formed expressions of an arbitrary length, for example with $n = 5$ we have 13566, and it is easy to test whether a string is well-formed in L by counting the number of odd numbers, for example clearly 13424242244 is not in well-formed since it contains exactly 2 odd degree numbers.

Now we can define the language of (classical) propositional logic:

DEFINITION 0.12 ⟨Formal language of propositional logic, CPL⟩

The alphabet of the language is given by:

1. A set of primitive symbols called variables, which are usually represented via letters from the English alphabet with an optional index. Variables are assigned truth-values via an unary truth-function, sometimes called a truth-assignment function.
2. A set of operator symbols called connectives which correspond to truth-functions.

While the set of connectives varies, it is common to use the following set:

1. \neg called logical negation (NOT) which is an unary truth-function $\mathbb{B} \rightarrow \mathbb{B}$ such that $\neg(T) = F$, $\neg(F) = T$.
2. \wedge called logical conjunction (logical AND), which was defined in 0.2 Logical AND.
3. \vee called logical disjunction (logical OR), which is a function $\vee : \mathbb{B}^2 \rightarrow \mathbb{B}$ s.t. $(A \vee B) = F$ if and only if $A = F$ and $B = F$ (otherwise $(A \vee B) = T$) (note that this can also be expressed via a truth table).
4. \implies called logical implication, which is a function $\implies : \mathbb{B}^2 \rightarrow \mathbb{B}$ s.t. $(A \implies B) = F$ if and only if $A = T$ and $B = F$. Note that if both A, B are false then $A \implies B$ is assigned the truth value T , which is counter-intuitive at first.

Parenthesis $()$ are also used for grouping compound propositions and are called grouping symbols, for example $(A \wedge B) \wedge (\neg C)$. When the formula is unambiguous, for example in the case of $A \vee \neg B$, parenthesis may be omitted.

We usually refer to expressions/words/formulas by the term formula and denote formulas with ϕ and φ . The language \mathcal{CPL} consists of all well-formed formulas (defined below) over the alphabet described above.

DEFINITION 0.13 *⟨well-formed formula, WFF⟩*

A formula ϕ is well-formed if and only if:

1. ϕ is simply a variable.
2. ϕ is of the form $\neg\varphi$ or $(\varphi \wedge \psi)$ or $(\varphi \vee \psi)$ or $(\varphi \implies \psi)$, where φ, ψ are well-defined formulas, called sub-formulas of ϕ .

Condition 1 motivates calling variables atomic formulas, since they are the building block of (well-defined) formulas. Well-formed formulas are abbreviated as WFF.

It is common to refer to WFFs of the form A and $\neg A$ where A is a proposition as literals for brevity.

Due to the recursive nature of the definition for WFFs, it is natural to represent WFFs using a hierarchical structure - an abstract syntax tree.

DEFINITION 0.14 *⟨abstract syntax tree, AST⟩*

An abstract syntax tree (AST) of a WFF is a tree (as defined in graph theory) whose leaves are variables and all other vertices are connectives applied on the children which are their sub-formulas.

Later we will present the unique readability theorem, whose result should be intuitively understood by anyone who has worked with ASTs in the past - given a WFF, there exists a single way (up to an isomorphism, i.e. up to a relabeling or reordering of the leaves) to read the formula, where *reading* here means to analyze the formula according to 0.13 well-formed formula, WFF, which is equivalent to forming an AST as per the above definition.

DEFINITION 0.15 *⟨logical equivalence⟩*

It is common to include the logical equivalence operator, denoted by \iff , as a connective. It is a binary operator whose outcomes is T if and only if both operands are T . As we will see later, it is equivalent to $(A \implies B) \wedge (B \implies A)$. The definition of a WFF can be extended to include this connective as well in condition 2.

EXAMPLE 0.4 *⟨classifying formulas in CPL⟩*

1. $\neg(\neg(\neg a))$ where a is a variable is well-formed: First we observe that a is an atomic formula by condition 1, so by condition 2, $\neg a$ is well-formed, so we can apply condition 2 again with $\varphi = \neg a$ and arrive at $\neg(\neg a)$ being well-formed. Applying condition 2 one last time, we get that $\neg(\neg(\neg a))$ is well-formed.
2. $(A \wedge B) \wedge \wedge C$ with A, B, C as variables is not well formed, since we have a well-formed formula $\varphi = A \wedge B$ (by conditions 1 on A, B and then condition 2 on $A \wedge B$), but then we have an expression of the form $\varphi \wedge \wedge C$ which is not a simple variable so it doesn't satisfy condition 1, but also it is not of any of the forms listed in condition 2, so it is ill-formed.
3. $A @ B$ where A, B are variables is ill-formed since we have not defined $@$ to be any connective or explicitly defined it in terms of a well-defined formula.

DEFINITION 0.16 *⟨truth assignment/evaluation of WFFs⟩*

Given a truth-assignment function $f : V \rightarrow \mathbb{B}$, where V is the set of simple variables in the language, and a WFF ϕ , the assignment of truth value to ϕ using f (extended to the domain of all WFFs in the language) is performed by following the recursive definition of a WFF:

1. If ϕ is a simple variable, then $\phi \in V$ and $f(\phi)$ is already given by the truth-assignment function.
2. There exists 4 other cases (5 if we include \iff) as specified by condition 2 in 0.13 well-formed formula, WFF (in the four cases below, φ, ψ are WFFs)
 1. $\phi = \neg\varphi$, then $f(\phi) = \neg f(\varphi)$ (using the truth-table of \neg to assign the final value)
 2. $\phi = \varphi \wedge \psi$, then $f(\phi) = f(\varphi) \wedge f(\psi)$
 3. $\phi = \varphi \vee \psi$, then $f(\phi) = f(\varphi) \vee f(\psi)$
 4. $\phi = \varphi \implies \psi$, then $f(\phi) = f(\varphi) \implies f(\psi)$

In other words, we extend the truth-assignment function f to a new truth-assignment function $\bar{f} : E \rightarrow \mathbb{B}$ where E is the set of expression in the formal language, such that:

1. $\forall x \in V, \bar{f}(x) = f(x)$
2. $\bar{f}(\mathcal{F}(x, \dots)) = \mathcal{F}(\bar{f}(x), \bar{f}(x), \dots)$ (shorthand for 2)

EXAMPLE 0.5 *⟨truth assignment⟩*

Given atomic formulas (i.e. variables) A, B, C such that $A \mapsto T, B \mapsto F, C \mapsto F$, we will evaluate the following WFF:

$$\begin{aligned}(A \wedge B) \vee (C \vee \neg C) &= f(A \wedge B) \vee f(C \vee \neg C) = \\ &= (f(A) \wedge f(B)) \vee (f(C) \vee (\neg f(C))) = \\ &= (T \wedge F) \vee (F \vee T) = F \vee T = T\end{aligned}$$

DEFINITION 0.17 *⟨satisfiability⟩*

A WFF ϕ is satisfiable if it is true (i.e. evaluates to true) under some assignment to its variables (i.e. if $\bar{f}(\phi) = T$, where \bar{f} is the extension of the truth-assignment function on the variables to the set of WFFs, as defined in 0.16 truth assignment/evaluation of WFFs).

The problem of determining if a WFF in propositional logic is satisfiable is decidable and known as the boolean satisfiability problem, or SAT.

DEFINITION 0.18 *⟨tautological (semantical) implication⟩*

Given a set ϕ of WFFs and a singular WFF ψ , we say that ϕ tautologically implies ψ if every truth-assignment function f under which all formulas in ϕ are satisfied (i.e. are assigned a true value) also satisfies ψ . We denote this relation as $\phi \models \psi$. Note that this symbol is not part of the formal language used to define the WFFs, but a part of the metalanguage used to discuss the language. We will not insist on ϕ being a proper set and will sometimes write $\phi \models \psi$ even if ϕ is a single proposition if ψ is semantically implied by ϕ (for example, $A \models \neg\neg A$).

DEFINITION 0.19 *⟨tautological equivalence⟩*

Given WFFs φ, ψ we say that they are tautologically equivalent if $\{\varphi\} \models \psi$ and $\{\psi\} \models \varphi$, and denote this relation as $\varphi \equiv \psi$.

DEFINITION 0.20 *⟨tautology⟩*

A WFF φ is a tautology if it is always true regardless of the truth assignment function. A tautology is represented via $\models \varphi$ (note the absence of a LHS).

EXAMPLE 0.6 *tautological implications*

1. Given $\phi = \{A, B\}$ (where A, B are propositions not necessarily atomic) and truth-assignment such that $A \mapsto T, B \mapsto T$, then $\phi \models A \vee B$.
2. Given $\varphi = \neg(A \wedge B)$ and $\psi = \neg A \vee \neg B$, by writing the truth table:

A	B	$A \wedge B$	$\neg(A \wedge B)$	$\neg A$	$\neg B$	$\neg A \vee \neg B$
T	T	T	F	F	F	F
T	F	F	T	F	T	T
F	F	F	T	T	T	T
F	T	F	T	T	F	T

φ and ψ are assigned T for exactly the same input values, so $\varphi \equiv \psi$.

3. Similarly, $A \iff B$ and $(A \implies B) \wedge (B \implies A)$ are tautologically equivalent, as shown by the following truth table

A	B	$A \implies B$	$B \implies A$	$(A \implies B) \wedge (B \implies A)$	$A \iff B$
T	T	T	T	T	T
T	F	F	T	F	F
F	F	T	T	T	T
F	T	T	F	F	F

4. The WFF $A \vee \neg A$ is a tautology, since its truth table for all possible truth assignments of A is always true.

DEFINITION 0.21 *model*

Suppose ϕ is a set of WFFs, then we say that \mathcal{M} is a model of ϕ if \mathcal{M} satisfies ϕ . In practice, for our purposes, a model is synonymous with a set of assignments to all variables of ϕ such that all WFFs in ϕ are satisfied. We denote this by $\models_{\mathcal{M}} \phi$.

By a simple process of comparing truth tables, we can arrive at the following set of useful tautological equivalents:

THEOREM 1 (propositional calculus)

Given propositional variables A, B, C and the extended set of connectives $\neg, \vee, \wedge, \implies, \iff$, the following are tautologically equivalent:

1. $(A \wedge B) \wedge C \equiv A \wedge (B \wedge C)$
2. $(A \vee B) \vee C \equiv A \vee (B \vee C)$
3. $((A \iff B) \iff C) \equiv (A \iff (B \iff C))$
4. $A \wedge (B \vee C) \equiv (A \wedge B) \vee (A \wedge C)$
5. $A \vee (B \wedge C) \equiv (A \vee B) \wedge (A \vee C)$
6. $\neg\neg A \equiv A$
7. $\neg(A \implies B) \equiv A \wedge \neg B$
8. $\neg(A \wedge B) \equiv \neg A \vee \neg B$
9. $\neg(A \vee B) \equiv \neg A \wedge \neg B$
10. $((A \wedge B) \implies C) \equiv (A \implies (B \implies C))$
11. $(A \implies B) \equiv \neg A \vee B$

Rules 1 – 3 state that the operators \wedge, \vee, \iff are associative, rules 4 – 5 state that \vee, \wedge are distributive with respect to each other, rules 6 – 9 deal with negation of propositions. Rules 8, 9 specifically are called De Morgan's laws. The last two rules 10, 11 deal with replacing the connectives used to construct a formula.

Proof. Each equivalence is easy to verify by comparing the truth-tables as we did in 0.21 model. For brevity it is left as an exercise for the reader to verify □

COROLLARY 1.1

Given a WFF ϕ , we can rewrite ϕ such that it only uses the connectives \vee, \neg .

Proof. We consider the extended set of connectives $\neg, \vee, \wedge, \implies, \iff$: these are the only connectives allowed in ϕ between any two sub-formulas. Recall from 0.21 model that $A \iff B \equiv (A \implies B) \wedge (B \implies A)$, thus every expression with \iff can be rewritten using the connectives \wedge, \implies , applying this to ϕ constructs a new formula ϕ' such that $\phi \equiv \phi'$ and ϕ' uses only $\neg, \vee, \wedge, \implies$. By Theorem 1 propositional calculus, we can rewrite each \implies expression using \neg, \vee (since by rule 11 $(A \implies B) \equiv \neg A \vee B$), so we can rewrite ϕ' as ϕ'' such that $\phi' \equiv \phi''$ and ϕ'' makes use of only \neg, \vee, \wedge . Finally, by De Morgan's laws and by rule 6 we can rewrite every sub-formula of the form $A \wedge B$ as $\neg(\neg A \vee \neg B)$, so finally we have $\phi'' \equiv \phi'''$ where ϕ''' is expressed only using \vee, \neg , and overall we have $\phi \equiv \phi'''$ □

COROLLARY 1.2

The language \mathcal{CPL} restricted to the connectives \vee, \neg and the language \mathcal{CPL} with connectives $\neg, \vee, \wedge, \implies, \iff$ can express exactly the set of formulas.

Proof. Adding connectives cannot reduce the amount of representable formulas since the smaller set of connectives is contained within the larger set, and by the previous corollary \vee, \neg can represent all formulas using the larger set of connectives, so the two are equivalent \square

This has powerful implications in proofs - since the two languages are equivalent, it suffices to prove results for only \vee, \neg , and the truth of these results for the larger set of connectives immediately follows from the equivalence. Note that a similar argument can be presented to show that \neg, \wedge and \neg, \implies can also be used to express all WFFs in propositional logic.

Tautology gives us a formal method for reasoning about some forms of proof:

EXAMPLE 1.1 *⟨proof by contradiction⟩*

In a proof by contradiction, instead of proving a proposition P , we prove that $\neg P \implies F$ (where F stands for the false value), and conclude that P must be true, which suggests $(\neg P \implies F) \equiv P$. Indeed, by rules 6, 11 of Theorem 1 propositional calculus we know that $\neg P \implies F$ is tautologically equivalent to $P \vee F$, which is equivalent to P , so this argument is valid. (we will define arguments and validity when introducing proof systems, for now we use the intuitive definitions).

EXAMPLE 1.2 *⟨proof by contrapositive⟩*

In a proof by the contrapositive, we prove $A \implies B$ by showing that $\neg B \implies \neg A$. Let us verify that the two are equivalent by comparing truth tables:

A	B	$A \implies B$	$\neg B \implies \neg A$
T	T	T	T
T	F	F	F
F	F	T	T
F	T	T	T

So an argument by the contrapositive is valid.

EXAMPLE 1.3 *⟨vacuous true⟩*

Consider the following proposition: “all living dinosaurs are vegetarian”. We say that this proposition is vacuously true, because there are no living dinosaurs. Expressed in the formal language of propositional language, this is equivalent to the statement $F \implies A$, where A is the proposition and F is a false value, which is T by the truth table of logical implication.

Recall that in classical logic, every proposition is assigned a truth value of true or false depending on the truth-assignment of its variables. Suppose that the set of variables is *finite*, then every propositional statement is identified with the truth table of a truth function $\mathbb{B}^k \rightarrow \mathbb{B}$, where k is the cardinality of the variables and \mathbb{B}^k is the n -tuple of the truth values assigned to the variables. Every proposition can be identified with such truth function. This begs the question - can the language \mathcal{CPL} with all the aforementioned connectives represent all possible boolean functions for a finite k ?

DEFINITION 0.22 *⟨functional completeness⟩*

A set of logical connectives is functionally complete if the corresponding truth functions of the connectives can be composited to construct the set of all truth functions $\mathbb{B}^k \rightarrow \mathbb{B}$, where $k \in \mathbb{N}$ and $k \geq 1$.

The question is then, are $\neg, \wedge, \vee, \implies$ functionally complete, which is answered in the following theorem.

THEOREM 2

The connectives $\neg, \wedge, \vee, \implies$ are functionally complete.

Proof. Let $k \in \mathbb{N}, k \geq 1$, and let $f : \mathbb{B}^k \rightarrow \mathbb{B}$ be a truth function, then we can write out the truth table for f by considering all 2^k possible value combinations (since k is finite, 2^k is countable, so this is a valid argument). Consider each combination for which the outcome of f is T and denote their amount by $n \in \mathbb{N}$ and consider the set of combinations $\{\alpha_1, \alpha_2, \dots, \alpha_n\}$ - these combinations are a combination of truth values of the variables, so each can be identified with a truth function $f_{\alpha_i} : \mathbb{B}^k \rightarrow \mathbb{B}, x \mapsto L_{\alpha_i}(x_1) \wedge L_{\alpha_i}(x_2) \wedge \dots \wedge L_{\alpha_i}(x_k)$, where $L_{\alpha_i}(x_j)$ denotes the literal of x_j (i.e. x_j or $\neg x_j$) in combination α_i . Now construct a new function $g : \mathbb{B}^k \rightarrow \mathbb{B}$ such that $g(x) = f_{\alpha_1}(x) \vee f_{\alpha_2}(x) \vee \dots \vee f_{\alpha_n}(x)$. Notice that by definition of logical disjunction, $g(x)$ is T only if any of $f_{\alpha_i}(x)$ is true, and each $f_{\alpha_i}(x)$ is true for a distinct combination for which $f(x)$ is a true, so in total we have that g and f have the same truth tables, but g uses only variables, \neg, \wedge, \vee in its explicit form, so it corresponds to a WFF using only \neg, \wedge, \vee , so the formula ϕ_g whose truth-function is g is tautologically equivalent to the proposition whose truth function is f □

An immediate corollary of the proof is that \neg, \vee are logically complete by Theorem 2 and 1.2. The form of g presented in the proof is also given a name

DEFINITION 0.23 \langle disjunctive normal form, DNF \rangle

A formula ϕ is considered to be in DNF if it is a disjunction (\vee) of finitely many disjunctions (\wedge) of literals ($A, \neg A$).

Theorem 2 provides a *semantic* means of converting a formula to DNF form. It is also possible to convert a formula to DNF form via *syntactic* means by using rules describes in Theorem 1 propositional calculus. By Theorem 2, the existence of a DNF for a given formula is guaranteed.

Note that the DNF is not unique, clearly $A \vee \neg A \vee B$ is in DNF but $A \vee \neg A \vee B \equiv B$ which is also in DNF.

DEFINITION 0.24 \langle conjunctive normal form, CNF \rangle

A formula ϕ is considered to be in CNF if it is a conjunction (\wedge) of finitely many disjunctions (\vee) of literals ($A, \neg A$).

THEOREM 3

Every WFF ϕ can be written in CNF.

Proof. Let ϕ be a WFF, then so is $\neg\phi$ by definition. By the same syntactic algorithm described in Theorem 2 we can convert $\neg\phi$ to DNF, denote this formula ψ . Since $\psi \equiv \neg\phi$, $\neg\psi \equiv \phi$ (by rule 6 in Theorem 1 propositional calculus). $\neg\psi$ is of the form $\neg(A_1 \vee A_2 \vee \dots \vee A_n)$, so by De Morgan laws, we have $\neg\psi = \neg A_1 \wedge \neg A_2 \wedge \dots \wedge \neg A_n$, but each term A_i is a conjunction of variable literals $A_i = L_{A_i}(x_1) \wedge L_{A_i}(x_2) \wedge \dots \wedge L_{A_i}(x_n)$, so again by De Morgan we get $\neg A_i = \neg L_{A_i}(x_1) \vee \dots \vee \neg L_{A_i}(x_n)$, so in total we have $\neg\psi$ is in CNF, but $\neg\psi \equiv \phi$, so we have written ϕ in CNF \square

Induction & Recursion

Induction

WFFs and truth assignment to WFFs are defined recursively, and oftentimes proofs are conducted inductively. Going forward, we will meet more recursive definitions and inductive proofs, so it is useful to have more formal definitions and theorems that deal with induction and recursion. We follow Enderton's discussion of the topic from *Mathematical Introduction to Logic*.

First, let's recall the general structure of an inductive proof of some property P for all natural numbers \mathbb{N} greater or equal to an initial natural number n_0 :

1. Basis: We show n_0 satisfies P . (we say that $P(n_0)$ is true, this is a notation we will formalize when discussing first-order logic)
2. Hypothesis: We suppose that some $n \in \mathbb{N}$ such that $n \geq n_0$ satisfies P .
3. Step: We show that from our hypothesis, we can infer that $n + 1$ satisfies P , thus we conclude P is satisfied by all $n \in \mathbb{N}$ s.t. $n \geq n_0$.

Informally, why does this proof hold? First, we observe that the property holds for n_0 , and then we show that if it holds for some n then it holds for $n + 1$ which is the successor of n (we will discuss the formal construction of the natural numbers at a different time), but then in particular it holds for $n = n_0$, so we can then take $n = n_0 + 1$ (for which the hypothesis is also true due to the inductive step), and eventually we can generate the entire set $S = \mathbb{N} \setminus \{n \in \mathbb{N} | n < n_0\}$ only by using n_0, S where S is the successor function $A : \mathbb{N} \rightarrow \mathbb{N}, n \mapsto n + 1$, and we know that the statement will hold for all successors since it held for the base case and for every successor. To encapsulate this relation, we say that A is (n_0, S) -inductive, which means that if $P(n_0)$ is true and $P(n) \implies P(S_n)$ is true, then $P(a)$ is true $\forall s \in A$.

Let's formalize the definition of an inductive set:

DEFINITION 0.25 *(inductive set)*

Given a set U (sometimes called a universe), a subset $B \subseteq U$ (sometimes called a basis), and a set of operators \mathcal{F} on U , we say that a subset $S \subseteq U$ is (B, \mathcal{F}) -inductive if:

1. S is closed under \mathcal{F} , i.e. $\forall \{ \in \mathcal{F}, \{ (s^*) \in S, \text{ where } s^* \text{ is shorthand for the } n\text{-tuple of elements of } S \text{ such that it is in the domain of } \mathcal{F}. \}$
2. $B \subseteq S$.

When B, \mathcal{F} are unambiguous, we may simply refer to S as an inductive set.

Readers may already be familiar with a different definition for an inductive set:

DEFINITION 0.26 *(inductive set)*

A set $S \subseteq \mathbb{R}$ is inductive if:

1. $1 \in S$
2. $\forall s \in S, s + 1 \in S$, i.e. S is closed under succession

Note that this is simply a special case of a (B, \mathcal{F}) inductive set with $B = \{1\}$ and $\mathcal{F} = \{x \mapsto x + 1\}$.

PROPOSITION 3

U is (B, \mathcal{F}) -inductive. (U, B, \mathcal{F} are defined in 0.25 inductive set).

Proof. \mathcal{F} is a set of operators on U so clearly U is closed under \mathcal{F} , and $B \subseteq U$ by definition, so U is (B, \mathcal{F}) -inductive. \square

DEFINITION 0.27 $\langle (B, \mathcal{F})$ generated set (top-down) \rangle

Let C^* be the intersection of all (B, \mathcal{F}) -inductive sets of U , then C^* is called the set generated by \mathcal{F} from B .

PROPOSITION 4

C^* as defined in 0.27 (B, \mathcal{F}) generated set (top-down) is the smallest inductive set of U .

Proof. First we show that C^* is inductive. The second condition is obviously met since $B \subseteq S$ for all inductive sets S so it is contained in their union. Now, the image of C^* under some $\{ \in \mathcal{F}$ is the image of an inductive set S restricted to C^* , so it is also in S since S is inductive, but this applies for all inductive sets S , so the image is also in their intersection, so it is also in C^* , so C^* is also closed under \mathcal{F} . In conclusion, C^* is inductive. Now, suppose by contradiction it is not minimal, then $\exists A \subseteq U$ such that A is inductive and $|A| < |C^*|$, but then C^* has some member not in A , but C^* is the intersection of all inductive sets, so $C^* \subseteq A$ for all inductive sets A , a contradiction. So C^* is minimal. \square

The reason this definition is called a *top-down* definition is because we start with some inductive set U at the top, and “go down” to smaller inductive sets, eventually reaching the bottom with C^* .

Let’s consider a different, more constructive (bottom-up) definition, where we incrementally build an inductive set:

DEFINITION 0.28 \langle construction sequence of a generated set \rangle

Given a universe U , a basis $B \subseteq U$, and a set of operators \mathcal{F} on U , we define a (B, \mathcal{F}) -construction sequence to be a sequence $C : \mathbb{N} \mapsto \text{Pow}(U)$ (where $\text{Pow}(U)$ is the powerset of U , i.e. the set of all subsets of U), such that $C_1 = B$ and $C_{i+1} = C_i \cup \mathcal{F}(C_i)$, where $\mathcal{F}(C_i)$ means “the image of C_i under all operators in \mathcal{F} , such that the number of inputs for each operator is matched”.

EXAMPLE 3.1

Let $U = \mathbb{R}$, $B = \{1\}$ and $\mathcal{F} = \{x \mapsto x+1\}$, then the (B, \mathcal{F}) -construction sequence is given by:

$$\begin{aligned} C_1 &= \{1\} \\ C_2 &= C_1 \cup \{\mathcal{F}(C_1)\} = \{1\} \cup \{1+1\} = \{1, 2\} \\ C_3 &= C_2 \cup \{\mathcal{F}(C_2)\} = \{1, 2\} \cup \{3\} = \{1, 2, 3\} \\ C_n &= \{1, 2, 3, 4, \dots, n\} \end{aligned}$$

Meaning that the constructive sequence is exactly the same of all natural numbers starting from 1 up to and including n .

PROPOSITION 5

Given a (B, \mathcal{F}) -constructive sequence $C_1, C_2, C_3, \dots, C_n$, then $C_1 \subseteq C_2 \subseteq C_3 \dots \subseteq C_n$.

Proof. By definition $C_n \subseteq C_{n+1}$

□

DEFINITION 0.29 $\langle (B, \mathcal{F})$ generated set (bottom-up) \rangle

Given a (B, \mathcal{F}) -constructive sequence C_n in some universe U , we say that $C_* = \bigcup_{n \in \mathbb{N}} C_n$ is the set generated by \mathcal{F} from B .

EXAMPLE 3.2 \langle bottom-up generated sets \rangle

1. Applying 0.29 (B, \mathcal{F}) generated set (bottom-up) to 3.1, we get a bottom-up construction of the set \mathbb{N} .

2. Consider \mathcal{CPL} with connectives \neg, \vee and a finite set of variables X (recall that this language is functionally complete), and let \overline{X} denote the universe of formulas in \mathcal{CPL} with these symbols, then the set of WFFs formed from this set of symbols can be constructed using a bottom-up approach: choose $B = X$ and $\mathcal{F} = \{\neg, \vee\}$ whose members are $\neg : \overline{X} \rightarrow \overline{X}, x \mapsto \neg x$ and $\vee : \overline{X} \times \overline{X} \rightarrow \overline{X}, (x, y) \mapsto x \vee y$, and the set of WFFs is given by C_* .

THEOREM 4

0.27 (B, \mathcal{F}) generated set (top-down) and 0.29 (B, \mathcal{F}) generated set (bottom-up) are equivalent, i.e. $C_* = C^*$

Proof. To prove the theorem, we show that $C_* \subseteq C^*$ and $C^* \subseteq C_*$.

First we argue that $C_* \subseteq C^*$ by induction on n : For $n = 1$, we have $C_1 = B$ and clearly $B \subseteq C^*$ by definition of an inductive set, now suppose $C_n \subseteq C^*$ for some n and consider C_{n+1} , whose elements are either from C_n so they are also in C^* or from \mathcal{F} operated on elements from C_n , but since C^* is closed under \mathcal{F} they are also members of C^* , so $C_{n+1} \subseteq C^*$, so $\forall n \in \mathbb{N}, C_n \subseteq C^*$ by (informal) induction, so by definition of C_* and by 5 we have $C_* \subseteq C^*$.

To show that $C^* \subseteq C_*$, it suffices to show that C_* is inductive, since C^* is defined as the intersection of all inductive sets so it is a subset of all inductive sets. First, $B = C_1$ so $B \subseteq C_*$, now suppose C_* is not closed under \mathcal{F} , i.e. $\exists c \in C_*^\omega$ and $\exists \{ \in \mathcal{F}$ such that $\{c\} \notin C_*$, but since C_* is a union of all C_n then $\exists k \in \mathbb{N}$ s.t. $c \in C_k^\omega$, but then $f(c) \in C_{k+1}$ by definition, and $C_{k+1} \subseteq C_*$, a contradiction, so C_* is closed under \mathcal{F} , so it is inductive.

Overall, we have $C_* \subseteq C^*$ and $C^* \subseteq C_*$, so $C^* = C_*$ □

Since $C_* = C^*$, we henceforth refer to this set as C (unless otherwise stated), and ignore the method of construction unless it is relevant.

THEOREM 5 (structural induction principle)

Let C be a (B, \mathcal{F}) generated set in some universe U , and let P be a property $P : U \rightarrow \mathbb{B}$ such that

1. $\forall b \in B, P(b) = T$, i.e. P is satisfied by all elements of B
2. $\forall \{ \in \mathcal{F}$, and let C^ω be a cartesian product of C such that each element has exactly the same number of elements as the arity of $\{$, that is to say $f(c)$ is defined $\forall c \in C^\omega$, then if all elements of c satisfy the property P , so does $f(c)$.

Then P is satisfied by all elements of C .

Proof. Since C is generated in U , then $C \subseteq U$. Consider the set P_U defined via a membership test as $P_U = \{u \in U | P(u) = T\}$, we shall show that it is (B, \mathcal{F}) inductive:

1. By assumption 1 of the theorem, $B \subseteq P_U$
2. By assumption 2 of the theorem, if c is a set of elements that satisfy P and matching the arity of some $\{ \in \mathcal{F}$, then $f(c)$ satisfies P , so P_U is closed under \mathcal{F} .

In conclusion, P_U is (B, \mathcal{F}) inductive, but since C is the set generated by \mathcal{F} from B then by 0.27 (B, F) generated set (top-down) $C \subseteq P_U$, so all elements of C satisfy P □

Note: for simplicity, we will drop the $^\omega$ notation and the emphasis on matching the number of elements to the arity of the operator when these details can be trivially filled in by the reader in favor of conciseness. Instead, we will simply write $f(\bar{c})$ to denote that \bar{c} is a cartesian product of elements in c such that it is in the domain of f .

Recursion

DEFINITION 0.30 (freely generated inductive set)

Let S be a (B, \mathcal{F}) generated set in some universe U . We say that S is freely generated if:

1. $\forall \{, \} \in \mathcal{F}, f[\overline{S}] \cap g[\overline{S}] = \emptyset$ (i.e. the range of all operators in \mathcal{F} are disjoint).
2. $\forall f \in \mathcal{F}, f[\overline{S}] \cap B = \emptyset$, i.e. the basis B and the range of $\{$ are disjoint.

In other words, all operators in \mathcal{F} are injective and their range and B are disjoint.

EXAMPLE 5.1

The set of WFFs is freely generated from $(B, \{\neg, \wedge, \vee, \implies, \iff\})$ where B is the set of atomic formulas. Note that this is only true if we use parenthesis properly (i.e. wrap the output of every connective in parenthesis), for example $(A \wedge B) \implies C$ can only be generated by applying the implication operator on $((A \wedge B), C)$, however if parenthesis are missing and no operator precedence is defined then $A \wedge B \implies C$ may be read as either $(A \vee B) \implies C$ or $A \vee (B \implies C)$, so only unambiguous WFFs are freely generated, but since ambiguity is not a property we wish to have, this does not limit us.

Upon proving this example, one can prove the *unique readability theorem* for WFFs

THEOREM 6 (unique readability theorem)

Given a WFF ϕ , there exists a unique reading of the formula.

This is left as an exercise for the reader. This theorem is critical in that it states that our language is well-defined.

THEOREM 7 (recursion theorem)

Let C be freely generated from U via (B, \mathcal{F}) , and for simplicity suppose $\mathcal{F} = \{f, g\}$ where f is an unary operator on U and g is a binary operator on U (but the theorem can be naturally extended to any set of operators \mathcal{F} such that C is freely generated).

Consider a new set V with functions h, F, G such that $h : B \rightarrow V$, F is an unary operator on V and G is a binary operator on V (so they correspond to f, g defined on U).

There exists a unique function $\bar{h} : C \rightarrow V$ such that:

1. $\forall x \in B, \bar{h}(x) = h(x)$ (i.e. \bar{h} restricted to B is h)
2. $\forall x, y \in C, \bar{h}(f(x)) = F(\bar{h}(x)) \quad \wedge \quad \bar{h}(g(x, y)) = G(\bar{h}(x), \bar{h}(y))$

In other words, \bar{h} forms a homomorphism from C (with operators f, g) to V (with operators F, G).

Proof. First we show that \bar{h} is well-defined $\forall x \in C$. Since C is freely generated from B, \mathcal{F} then for every $x \in C$, either:

1. $x \in B$, in which case $\bar{h} : x \mapsto h(x)$ by 1
2. $x = f(\alpha)$ for some $\alpha \in C$, so $\bar{h}(x) = \bar{h}(f(\alpha)) = F(\bar{h}(\alpha))$ by 2
3. $x = g(\alpha, \beta)$ for some $(\alpha, \beta) \in C^2$, so $\bar{h}(x) = \bar{h}(g(\alpha, \beta)) = G(\bar{h}(\alpha), \bar{h}(\beta))$ by 2

Since x must meet only one of the three cases (since C is freely generated), then \bar{h} is well-defined.

For uniqueness, suppose $\exists k : C \rightarrow V, k \neq \bar{h}$ that meets both conditions, so $\exists x \in C$ such that $\bar{h}(x) \neq k(x)$. We have 3 cases:

1. $x \in B$, but then $k : x \mapsto k(x)$ which is the same as \bar{h} restricted to B , so in this case $\bar{h}(x) = k(x)$, so this case is impossible.
2. $x = f(\alpha)$ for some $\alpha \in C$, so $k(x) = F(k(\alpha))$, but since C is freely generated α must be either in B or a result of a previous application of f on some $\alpha_1 \in C$, which eventually ends in some $\alpha_0 \in B$ (since $f[C]$ can be generated bottom-up by repeatedly applying f in B and the results of previous applications), so we have $F(k(\alpha)) = (F \circ F \circ F \cdots \circ F \circ k)(\alpha_0)$ which is a member of B , so by 1 we have $k(\alpha_0) = \bar{h}(\alpha_0)$, so $F(k(\alpha)) = F(\bar{h}(\alpha))$, so this case is also impossible.
3. By a similar argument to the previous one, this case is also impossible.

So we have shown that $\nexists x \in C$ s.t. $\bar{h}(x) \neq k(x)$, a contradiction, so \bar{h} is unique □

EXAMPLE 7.1 <coloring analogy>

Suppose V is a color space, i.e. every element of V corresponds to a color, and suppose you wish to assign a color to all elements of C using a recursive definition, i.e. you explicitly provide the color for the basis B (this is the function h), and explain how to obtain the color of the outcome of f via a function F , and similarly of g via a function G . If C is freely generated, then by the recursion theorem such definition is well-defined, i.e. there won't be a case where for some $x \in C \setminus B$, both F and G will be able to designate (potentially contradictory) colors for x .

EXAMPLE 7.2 <truth-assignment to a WFF>

Since WFFs are freely generated, then by 0.16 truth assignment/evaluation of WFFs given a truth evaluation function f for the set of variables, the extension of f to the set of WFFs is unique and well-defined by Theorem 7 recursion theorem.

Proof Systems

As a motivating example for our discussion of formal proof systems, consider the topic of graph coloring, as defined in graph theory, and consider the following lemma:

LEMMA 7.1

If a simple graph $G = (V, E)$ has a proper coloring $f : V \rightarrow C$ where C is the set of colors and E is not the empty set (i.e. there is at least one edge in the graph), then not all vertices are of the same color.

Proof. Suppose $G = (V, E)$ is a simple graph with at least one edge, so $\exists e \in E$ such that $e = \{u, v\}$ for some $u, v \in V, u \neq v$ (otherwise the graph isn't simple). Now, suppose G has a proper coloring $f : V \rightarrow C$, then by definition $\forall v \in V, f(v) \notin f(\text{nbrs}(v))$, where $\text{nbrs}(v)$ is the set of neighbors of v and $f(\text{nbrs}(v))$ is the image of $\text{nbrs}(v)$ under f , and since $u \in \text{nbrs}(v)$ since $\{u, v\} \in e$, then $f(v) \neq f(u)$, which completes the proof \square

Let's attempt to express this statement in \mathcal{CPL} , making a few simplifications along the way. First, we explicitly state that G has finitely many vertices, so we can enumerate its vertices. Next, we suppose that G is 2-colorable, i.e. $|C| = 2$, and label its elements $C = \{R, B\}$ (for *red* and *blue*). These are the assumptions of the theorem. To express them in formal language, we can assign a simple variable X_i^j to each vertex v such that X_v^j is the proposition "vertex v has color j ", which evaluates to either true or false. Stating that G is 2-colorable is equivalent to stating that for each v , at least one of X_v^R, X_v^B is true, and for the coloring to be well-formed we also need to require that exactly one of X_v^R, X_v^B be true. Since V is finite, we can express these propositions using a notation similar to sigma notation for summation:

1. $\bigwedge_{v \in V} (X_v^R \vee X_v^B)$
2. $\bigwedge_{v \in V} \neg(X_v^R \wedge X_v^B)$

These can be merged into a single statement, but for simplicity we keep them separate. Also, for the coloring to be proper, we need to require that no two vertices incident with the same edge share the same color, which can be formalized as:

3. $\bigwedge_{\{u, v\} \in E} \neg(X_v^R \wedge X_u^R) \wedge \neg(X_v^B \wedge X_u^B)$

Note that the assumptions that G has at least one edge is implied by supposing that 3 is not empty.

Finally, the theorem states that from these assumptions, it follows that not all vertices are of the same color, which can be expressed as

1. $\neg(\bigwedge_{v \in V} X_v^R) \wedge \neg(\bigwedge_{v \in V} X_v^B)$

The theorem then can be expressed as follows:

$$\begin{aligned} & \{ \bigwedge_{v \in V} (X_v^R \vee X_v^B), \\ & \bigwedge_{v \in V} \neg(X_v^R \wedge X_v^B), \\ & \bigwedge_{\{u,v\} \in E} \neg(X_v^R \wedge X_u^R) \wedge \neg(X_v^B \wedge X_u^B) \} \end{aligned}$$

$$\models \neg(\bigwedge_{v \in V} X_v^R) \wedge \neg(\bigwedge_{v \in V} X_v^B)$$

While we know (or believe) this statement to be true since we have provided a non-formal proof for this theorem above, it is not trivial to verify the tautological implication using the tools we have so far: writing out the truth table for all possible truth values of the set of variables is daunting, and if $|V|$ is very large it would require structural induction as well since it would be unwieldy to check that many cases even for a computer. Instead of showing tautological implication via truth tables (which is a *sematical implication*), we can instead consider *syntactical implications*, in which we derive propositions based on previous propositions which are true using inference rules and axioms. Obviously for the propositions to be sound (we will define soundness in a bit), they also need to be tautologically implied from the set of previous propositions. This is the idea behind a formal proof system, which we will now formalize.

DEFINITION 0.31 *⟨logical axiom⟩*

A logical axiom is a WFF that is a tautology and that is taken to be true by a proof system.

We refer to *logical axioms* by *axioms* when the distinction is unambiguous or unimportant.

DEFINITION 0.32 *⟨rule of inference⟩*

A rule of inference is a rule that, given a set of WFFs, analyzes their structure and, if they match the rule, derives a conclusion.

EXAMPLE 7.3 ⟨modus ponens⟩

Modus ponens (MP) is a rule of inference which states the following: given two WFFs of the form:

1. $A \implies B$
2. A

Such that A, B are sub-formulas and 1, 2 evaluate to true, then we can conclude that the WFF B is also true.

Note that this is almost exactly the same as the tautological implication $A \wedge (A \implies B) \models B$, which justifies the soundness of MP.

DEFINITION 0.33 ⟨formal proof system⟩

A formal proof system is a collection of logical axioms and inference rules used to derive WFFs. Formal proof systems are also called deduction systems.

DEFINITION 0.34 ⟨formal proof⟩

A formal proof in a formal proof system is a finite sequence of WFFs P_1, P_2, \dots, P_n such that each P_i is one of:

1. A logical axiom
2. An assumption
3. Follows from preceding WFFs in the sequence from a rule of inference

The last WFF in the sequence (P_n) is known as the conclusion of the proof. The length of the proof is the number of WFFs, n .

DEFINITION 0.35 ⟨syntactical implication⟩

Given a set of WFFs ϕ and a single WFF φ , we say that φ is syntactically implied by ϕ in some formal proof system if by supposing all WFFs of ϕ evaluate to true, then we can devise a formal proof for φ . We denote this relation by $\phi \vdash \varphi$.

Since axioms always evaluate to true, we write $\vdash A$ for any axiom A since any set of WFFs ϕ including \emptyset syntactically implies A .

EXAMPLE 7.4 ⟨formal proof using MP⟩

Let's use a formal proof system with no axioms and MP as the only rule of inference, then if ϕ is over set of WFFs and $\phi \vdash (A \implies B)$ for some A, B sub-formulas, then $\phi \cup \{A\} \vdash B$. The formal proof for this is as follows, starting from a set of assumptions $\phi \cup \{A\}$:

1. $A \implies B$ (since $\phi \vdash (A \implies B)$ and $\phi \subset \phi \cup \{A\}$)
2. A (assumption)
3. B (MP on 1, 2)

DEFINITION 0.36 ⟨soundness of a proof system⟩

A proof system is sound if for every set of assumptions ϕ and for every WFF ψ , then $\phi \vdash \psi \implies \phi \models \psi$, i.e. we can only prove tautologies (so we can't prove false propositions).

DEFINITION 0.37 ⟨(semantical) completeness of a proof system⟩

A proof system is complete or semantically complete (to distinguish from functional completeness of a language) if for every set of assumptions ϕ and for every WFF ψ , then $\phi \models \psi \implies \phi \vdash \psi$, i.e. we can prove all tautologies implied by ϕ .

Finally, we can present the formal definition for a propositional proof system:

DEFINITION 0.38 ⟨propositional proof system⟩

A propositional proof system (pps) is a polynomial time function P whose image is the set of propositional tautologies (denoted TAUT) such that it is:

1. Complete
2. Sound
3. Efficient

Where efficient is defined as running in polynomial time.

PROPOSITION 6

A sound formal proof system implies that all logical axioms must be tautologies

Proof. Let A be an axiom of a proof system, then $\vdash A$, and by completeness $\models A$, so A must be a tautology □

The last proposition motivates our definition of logical axioms as tautologies. Note that if we are being truly rigorous, if we simply take a WFF and declare it an axiom, it doesn't do us much good, since the

axiom will only apply for the variables in the WFF. For example, if we take $A \vdash (B \implies A)$ to be an axiom (verify that it is indeed a tautological implication as well!), then given two other sub-formulas φ, ψ we can't simply claim that by the axiom alone $\psi \vdash (\varphi \implies \psi)$, since the axiom deals with A, B , but clearly we would like axioms to apply for any sub-formula otherwise they are not very useful. This leaves us with two options:

1. Add a new inference rule - a rule of substitution - which loosely speaking states that we can replace the variables in a formula with other variables that satisfy the same sequence of formulas, so for example in this case we can apply a rule of substitution and replace A, B with ψ, φ respectively.
2. Define axiom *schemas* or *templates* in the metalanguage of the formal system, which uses *schematic variables* which can be substituted for any variable in the formal language if the WFF matches the schema, then each application of an axiom schema is an *instantiation* of the schema where the schematic variables are instantiated to be variables from the formal language. This is analogous to defining generic or templated functions in programming languages that support metaprogramming paradigms.

We will opt for the second option, which means we should provide a definition for an axiom schema:

DEFINITION 0.39 \langle axiom scehma \rangle

An axiom schema is a WFF in the metalanguage of a formal system, in which schematic (or meta) variables appear. Metavariables are replaced by variables of the system when a schema is instantiated.

We will now present a common propositional proof system, called Łukasiewicz's P_2 , which itself is a revision of Frege's system, both of which are examples of axiomatic systems known as *Hilbert systems*.

DEFINITION 0.40 \langle Łukasiewicz's P_2 \rangle

P_2 is a propositional proof system with MP as the single rule of inference, and the following set of axiom schemas:

- A1. $\varphi \implies (\psi \implies \varphi)$
- A2. $(\varphi \implies (\psi \implies \chi)) \implies ((\varphi \implies \psi) \implies (\varphi \implies \chi))$
- A3. $(\neg\psi \implies \neg\tau) \implies (\tau \implies \psi)$

One can verify that all instantiations of these schemas are tautologies.

EXAMPLE 7.5

In P_2 , given variables A, B , we will show that $\{A\} \vdash B \implies A$:

1. A (assumption)
2. $A \implies (B \implies A)$ (instantiation of axiom A1 with A, B)
3. $B \implies A$ (MP on 1, 2)

A quick word on the distinction between axiom schemas and inference rules: while both share the similarity that both should produce tautologies using syntactic rules (otherwise the system has no hope of being sound), and both provide templates that, when matched, produce new valid propositions, they are distinct in that the input of axiom schemas is WFFs which are taken as *variables* and the output is a WFF, while the input of inference rules is WFFs with a certain structure and the output is a WFF. Put more concisely, inference rules operate on sets of WFFs, while axiom schemas operate on variables.

THEOREM 8 \langle soundness of the proof system P_2 \rangle

The proof system P_2 is sound, i.e. $\phi \vdash \varphi \implies \phi \models \varphi$.

Proof. Suppose $\phi \vdash \varphi$, then there exists a finite proof for φ in ϕ . We prove the soundness of P_2 by induction on the length of the proof. For $n = 1$, φ is either in ϕ or a logical axiom, which is a tautology, so clearly $\phi \models \varphi$. For $n > 1$, φ must be inferred from previous propositions by an inference rule, and the only one in P_2 is MP, so the proof must show $\phi \vdash \{\psi \implies \varphi, \psi\}$ and the proof for this must be shorter than n , so by the induction hypothesis $\phi \models \{\psi \implies \varphi, \psi\}$, and now by the soundness of MP this means that $\phi \models \varphi$ \square

We will soon see that P_2 is *complete*, but first we will show more elementary results, which are *metatheorems* since the object they study is the formal system itself (as opposed to theorems which study the objects the system deals with). Note that when proving a metatheorem by induction, the induction is applied on the length of the proof.

THEOREM 9 \langle law of identity in P_2 \rangle

Every proposition implies itself:

$$\vdash p \implies p$$

Proof.

1. $(p \implies ((p \implies p) \implies p)) \implies ((p \implies (p \implies p)) \implies (p \implies p))$ (A2 instantiated with $\varphi = p, \psi = (p \implies p), \chi = p$)
2. $p \implies ((p \implies p) \implies p)$ (A1 instantiated with $\varphi = p, \psi = (p \implies p)$)
3. $(p \implies (p \implies p)) \implies (p \implies p)$ (MP on 1, 2)
4. $p \implies (p \implies p)$ (A1 with $\varphi = p, \psi = p$)
5. $p \implies p$ (MP on 3, 4)

THEOREM 10 <deduction theorem in P_2 >

Given two formulas φ, ψ and a set of initial propositions ϕ which are true such that ψ is provable from $\phi \cup \{\varphi\}$, then $\varphi \implies \psi$ is provable from ϕ . Written formally, the theorem asserts the following:

$$(\phi \cup \{\varphi\} \vdash \psi) \implies \phi \vdash (\varphi \implies \psi)$$

Proof. We prove by induction on the length of the proof $\phi \cup \{\varphi\} \vdash \psi$.

For $n = 1$, then the proof is simply $\{\psi\}$, and by definition of a formal proof this means that either:

1. ψ is an assumption, i.e. $\psi \in \phi \cup \{\varphi\}$, so either:

1. $\psi \in \phi$
2. $\psi = \varphi$
2. ψ is an axiom.

In cases 1.1 and 2, then by axiom A1, $\psi \implies (\varphi \implies \psi)$ and by MP we have $(\varphi \implies \psi)$ (this is just 7.5). In case 1.2, by Theorem 9 law of identity in P_2 we have $(\varphi \implies \psi)$.

For some $n > 1$, the only other way to arrive at $\phi \cup \{\varphi\} \vdash \psi$ is through a law of inference (since all other valid forms of argument are only a single proposition and we have already covered those in the base case), and the only law of inference in P_2 is MP, so the proof must include the following two WFFs:

1. A
2. $A \implies \psi$

(then by MP we have ψ)

And A itself must be implied by some previous propositions in the proof, i.e. there exists a proof with length less than n that $\phi \cup \{\varphi\} \vdash A$, but then by the induction hypothesis we also have $\phi \vdash \varphi \implies A$ and also that $\phi \vdash \varphi \implies (A \implies \psi)$, so we present the following argument from initial propositions ϕ :

1. $\varphi \implies A$ (hypothesis, justified above)
2. $\varphi \implies (A \implies \psi)$ (hypothesis, justified above)
3. $(\varphi \implies (A \implies \psi)) \implies ((\varphi \implies A) \implies (\varphi \implies \psi))$ (A3)
4. $(\varphi \implies A) \implies (A \implies \psi)$ (MP on 2, 3)
5. $\varphi \implies \psi$ (MP on 1, 4)

Which completes the proof by induction □

Completeness Theorem

DEFINITION 0.41 (consistency of a formal proof system)

A formal proof system is consistent if for any proposition p , the formal proof system does not have both p and $\neg p$ as theorems (theorem in this sense means that a formal proof for the statement exists in the system).

THEOREM 11 (principle of explosion)

If a formal proof system is inconsistent, then any proposition ψ is provable in the system.

Proof. Suppose $\phi \vdash p$ and $\phi \vdash \neg p$ for some p (this is the definition of an inconsistent proof system), and let ψ be a proposition. We will provide a proof in P_2 :

1. $(\neg\psi \implies \neg p) \implies (p \implies \psi)$ (A3)
2. $\neg p \implies (\neg\psi \implies \neg p)$ (A1)
3. $\neg p$ (hypothesis)
4. $\neg\psi \implies \neg p$ (MP on 2, 3)
5. $p \implies \psi$ (MP on 1, 4)
6. p (hypothesis)
7. ψ (MP on 5, 6)

□

THEOREM 12 (proof by contradiction (reductio ad absurdum) in P_2)

Given a proposition p , if $\phi \cup \{\neg p\}$ is inconsistent, then $\phi \vdash p$.

Note that this is different from the example 1.1 proof by contradiction presented in the context of tautological implications, since we are dealing with syntactical implication in a formal proof system.

Proof. Pick some tautology ψ which is provable by ϕ (i.e. $\phi \vdash \psi$, there are always such tautologies), then by the principle of explosion $\phi \cup \{\neg p\} \vdash \neg\psi$, so by the deduction theorem $\phi \vdash \neg p \implies \neg\psi$, so we can write the following formal proof from ϕ :

1. $\neg p \implies \neg\psi$ (deduction theorem, justified above)
2. $(\neg p \implies \neg\psi) \implies (\psi \implies p)$ (A3)
3. $\psi \implies p$ (MP on 1, 2)
4. ψ (provable by ϕ)

5. p (MP on 3, 4)

□

We will soon show that consistency is equivalent to satisfiability in P_2 . To do this, we will need a new definition:

DEFINITION 0.42 *(maximally consistent set)*

Given a consistent set of WFFs ϕ , then ϕ is maximally consistent if for every WFF φ , either $\varphi \in \phi$ or $\neg\varphi \in \phi$, but not both, and if $\varphi \notin \phi$ then $\phi \cup \{\varphi\}$ is inconsistent.

LEMMA 12.1 *(maximization existence)*

If ϕ is consistent, then ϕ can be extended to a maximally consistent set ϕ^* .

Proof. Consider the domain of all WFFs, and consider some enumeration on them $P = \{p_1, p_2, p_3, \dots\}$ (for example, consider the bottom-up generation of the set of WFFs and take an arbitrary ordering of the set of variables B as the initial enumeration, then for every C_n in the construction sequence of the WFFs pick an arbitrary ordering of the WFFs in C_n but not in C_{n-1} and add them to the existing enumeration), and define a new sequence ϕ_n as follows:

$$\begin{aligned} \phi_0 &= \phi \\ \phi_{n+1} &= \begin{cases} \phi_n \cup \{p_{n+1}\} & \text{if } \phi_n \cup \{p_{n+1}\} \text{ is consistent} \\ \phi_n & \text{otherwise} \end{cases} \end{aligned}$$

Note that $\forall n \in \mathbb{N}$, ϕ_n is consistent by construction, as well as $\phi_0 \subseteq \phi_1 \subseteq \phi_2 \subseteq \dots \subseteq \phi_n$. Now define ϕ^* as the union of all ϕ_n , i.e. $\phi^* = \bigcup_{n \in \mathbb{N}} \phi_n$.

We shall prove that ϕ^* is maximally consistent:

1. consistency - suppose ϕ^* is inconsistent, then $\exists p \in P$ such that $\phi^* \vdash p$ and $\phi^* \vdash \neg p$, so there exists two finite proofs in ϕ^* for both p and $\neg p$, and since $\phi_0 \subseteq \phi_1 \subseteq \dots \subseteq \phi^*$, then $\exists N \in \mathbb{N}$ such that $\forall n \in \mathbb{N}, (n \geq N \implies ((\phi_n \vdash p) \wedge (\phi_n \vdash \neg p)))$, but then ϕ_n is inconsistent, a contradiction.
2. For the second condition ($\forall p \in P$ either $p \in \phi^*$ or $\neg p \in \phi^*$ and if $p \notin \phi^*$ then $\phi^* \cup \{p\}$ is inconsistent), by a similar argument we have that both $p, \neg p$ are the elements of the ordered set P matching some indices $i, j \in \mathbb{N}$, then ϕ_i, ϕ_j have considered $p, \neg p$ so if they are not in ϕ_n starting from those indices, then $\phi_n \cup \{p\}$ (or $\phi_n \cup \{\neg p\}$) is inconsistent which satisfies the second sub-condition, and clearly one of the two must be consistent which satisfies the first sub-condition.

□

COROLLARY 12.1

If ϕ is maximally consistent, then for every WFF p , either $\phi \vdash p$ or $\phi \vdash \neg p$.

Proof. Since ϕ is maximally consistent, either $p \in \phi$ or $\neg p \in \phi$. In the former, $\phi \vdash p$ trivially by the proof $\{p\}$ (which is valid since $p \in \phi$). In the latter, $\phi \vdash \neg p$ by the same argument \square

THEOREM 13 \langle consistence iff satisfiable \rangle

P_2 with ϕ is consistent, i.e. $\nexists p$ such that $\phi \vdash p$ as well as $\phi \vdash \neg p$, if and only if ϕ is satisfiable.

Proof. \Leftarrow We will prove the contrapositive: suppose ϕ is inconsistent, then $\phi \vdash p$ as well as $\phi \vdash \neg p$. Now, suppose by contradiction that ϕ is satisfiable, i.e. it has some model \mathcal{M} for which (by the soundness of P_2) if $\forall \varphi$ WFF such that $\phi \vdash \varphi$ then $\phi \models_{\mathcal{M}} \varphi$, but then $\phi \models_{\mathcal{M}} p$ and $\phi \models_{\mathcal{M}} \neg p$, which is tautologically impossible, a contradiction, so if ϕ is inconsistent then it is not satisfiable, then the original form of the proposition is also true: if ϕ is satisfiable, then it is consistent.

\Rightarrow Suppose ϕ is consistent. Consider the extension ϕ^* of ϕ to a maximally consistent set (by 12.1 maximization existence) and consider the ordered set of variables $X = \{X_1, X_2, \dots, X_n\}$ in ϕ . We will show that $\exists! \mathcal{M}$ (i.e. there exists a unique model \mathcal{M}) that satisfies ϕ^* , and since $\phi \subseteq \phi^*$ then it is also a model of ϕ , thus showing that ϕ is satisfiable, which will complete the proof. This part of the proof is also called the *model existence theorem* for propositional logic, since it shows that for every consistent set ϕ there exists a model which satisfies ϕ .

Observe that for every $X_i \in X$ then by 12.1 either $\phi \vdash X_i$ or $\phi^* \vdash \neg X_i$, so in the former we assign X_i to T and in the latter we assign X_i to F . This way, we construct a model \mathcal{M} . Now we need to show that \mathcal{M} is indeed a model of ϕ^* , i.e. that $\forall \varphi \in \phi^*, \models_{\mathcal{M}} \varphi$. We do this by induction on the structure of φ . To simplify the proof, we consider φ in its equivalent form expressed via \neg, \vee only, but the inductive step can easily be extended to all 4/5 connectives of \mathcal{CPL} .

In the base case where the length of φ is 1, then φ is a variable, so it already is assigned to T by our construction of \mathcal{M} . Now let φ have length $n+1$, so either φ is of the form $\neg\psi$, where ψ is a WFF of length less than n so by the induction hypothesis $\models_{\mathcal{M}} \psi$ so $\models_{\mathcal{M}} \neg\psi$, or φ is of the form $\psi \vee \chi$, which is true iff ψ or χ or both are satisfied by \mathcal{M} , so by construction either $\phi^* \vdash \psi$ or $\phi^* \vdash \chi$ or both, so by the induction hypothesis (since both ψ and χ have length less than n) then $(\phi^* \vdash \psi) \Rightarrow (\models_{\mathcal{M}} \psi)$ and similarly for χ , so $\models_{\mathcal{M}} (\psi \vee \chi)$, which completes the proof by induction.

To conclude, we have constructed a model \mathcal{M} that satisfies ϕ^* , which itself was an extension of an arbitrary consistence set ϕ , so \mathcal{M} always satisfies ϕ , so ϕ is satisfiable. \square

THEOREM 14 (consistency implies completeness)

If ϕ is consistent, then it is complete, i.e. for every WFF p , $\phi \models p \implies \phi \vdash p$.

Proof. Let p be a WFF such that $\phi \models p$. Consider $\phi \cup \{\neg p\}$, it is either consistent or inconsistent. If it is inconsistent, then by Theorem 12 proof by contradiction (reductio ad absurdum) in P_2 $\phi \vdash p$. If it is satisfiable, then by Theorem 13 consistency iff satisfiability $\exists \mathcal{M}$ such that $\models_{\mathcal{M}} \phi \cup \{\neg p\}$, and in particular $\phi \models_{\mathcal{M}} \neg p$, but since $\phi \models p$ for every model then $\phi \models_{\mathcal{M}} p$ as well, which is impossible, so $\phi \cup \{\neg p\}$ cannot be consistent \square

THEOREM 15 (compactness theorem for P_2)

A set of WFFs ϕ is satisfiable (i.e. has a model) if and only if every finite subset $\pi \subseteq \phi$ is satisfiable.

Proof. \implies Let ϕ be satisfiable and consider some model \mathcal{M} that satisfies ϕ , then \mathcal{M} satisfies all sentences in ϕ , and in particular any finite subset of sentences $\pi \subseteq \phi$, so such subset π is satisfiable.

\Leftarrow By Theorem 13 consistency iff satisfiability it suffices to show that if every finite subset $\phi \subseteq \phi$ is consistent, then ϕ is consistent. We will prove that contrapositive instead: suppose ϕ is inconsistent, then exists some WFF p such that $\phi \vdash p, \phi \vdash \neg p$. Since these proofs are finite, each proof only requires a finite subset π_1, π_2 of ϕ to be valid. Now, consider $\pi_1 \cup \pi_2$ - it is the union of two finite subsets of ϕ , so it is also a finite subset of ϕ , but since $\pi_1 \vdash p$ and $\pi_2 \vdash \neg p$, then $\pi_1 \cup \pi_2$ is inconsistent, which completes the proof by contrapositive. \square

First-Order Logic

Consider the following argument:

1. All humans are mammals
2. Plato is human

Hence:

3. Plato is a mammal (by 1, 2)

While we understand the argument to be valid, how may we formalize it? In the language of propositional logic, one attempt would be to formalize the argument as follows: we introduce 3 variables:

1. A ("human")

2. B (“mammal”)

3. C (“Plato”)

Then the premises become $A \implies B$ and $C \implies A$, and the conclusion becomes $C \implies B$ (which can be verified to be valid and sound), however if we consider the meaning of these statements we find that they are not accurate translations of our argument, because now our argument is “if human then mammal, if Plato then human, so if Plato then mammal”, now if we suppose “Plato” then we can prove “human” and “mammal”, but what is the meaning of “human” and “mammal” as standalone propositions? Clearly this makes no logical sense. Instead we may try to formalize the argument as follows:

1. A (“Plato is human”)

2. B (“Plato is a mammal”)

3. C (“Plato”)

And then all variables truly are propositions and the conclusion becomes “Plato is a mammal”, but what about Socrates? Using the premises in the informal argument it is clear that by “Socrates is human” and by the first premise we can conclude that Socrates is also human, but to prove this for Socrates we will need an entirely new argument, which does not properly express the shared property of Socrates and Plato (being human) that leads to the shared conclusion.

The issue here is that, in propositional logic, we can only speak about *propositions*, and not about *non-logical objects* such as Plato and Socrates and *predicates* (or properties) such as `IsHuman` or `IsMammal` and *quantifiers* such as “all”. First-order logic, which is also called *predicate logic* (perhaps that term is more enlightening given our brief discussion) answers those needs by formally introducing non-logical objects, predicates and quantifiers as an extension to propositional logic.

Since most of math studies non-logical objects (in field theory we study objects which are fields, in graph theory we study graphs, etc.), then it follows that a first-order logic (*FOL*) is the proper formal setting for most of math.

Formal Language

Modern notions of FOL are ones where we define specific FOL systems for each field of study instead of defining one grand FOL system in which we can formalize (almost) all of math (*almost* since not all of math can be conducted in a FOL system). We will see some examples shortly after presenting the formal definitions.

DEFINITION 0.1 (formal language of first-order logic)

The alphabet of the formal language of first-order logic, denoted \mathcal{L} , consists of logical symbols:

1. A set of variable symbols, formally given by letters from the English alphabet with an optional index

2. The set of connectives of \mathcal{CPL} , i.e. $\neg, \vee, \wedge, \implies, \iff$.

3. A set of quantifiers:

1. A universal quantifier, \forall (for all)

2. A existential quantifier, \exists (exists)

Quantifiers produce new formulas from a variable symbol and a formula, as we will soon show.

4. A set of grouping (punctuation) symbols - parenthesis and a comma.

5. An equality symbol $=$.

As well as a vocabulary $\{c_0, c_1, \dots, f_0, f_1, \dots, R_0, R_1, \dots\}$ which consists of (not necessary finitely many):

1. constant symbols, given by c_0, c_1, \dots

2. function symbols given by f_0, f_1, \dots , where each function has an associated arity (i.e. number of arguments), representing operations.

3. relation symbols given by R_0, R_1, \dots , representing properties of n -tuples (n being the arity of the relation).

Different FOLs have different vocabularies.

Note that the vocabulary is not a set of constants, functions and relations, but rather a set of symbols representing constants, functions and relations. To provide semantical meaning to these symbols, we will define a new object, called a structure.

EXAMPLE 15.1 (vocabulary of a FOL)

1. The vocabulary of the FOL of set theory is (\emptyset, \in) where \emptyset is a constant symbol and \in is a relation symbol.

2. The vocabulary of the FOL of field theory is $(0, 1, +, \cdot)$ where $0, 1$ are constant symbols, and $+, \cdot$ are function symbols.

3. The vocabulary of the FOL of number theory is $(0, 1, S, +, \times)$ where $0, 1$ are constant symbols and $S, +, \times$ are function symbols (S is the symbol for the successor function).

DEFINITION 0.2 (syntax of a FOL language L)

The syntax of a FOL language is concerned with two syntactic objects:

1. terms, which are strings representing elements. (this will make more sense when we define a structure)
2. formulas, which are strings representing statements.

In \mathcal{CPL} , we only dealt with the latter.

The following are terms:

1. A variable symbol
2. A constant symbol
3. $f_i(t_1, t_2, \dots, t_n)$ where f_i is a function symbol with arity n and all of t_1, t_2, \dots, t_n are terms.

The following are (well-formed) formulas:

1. $t_1 = t_2$ where t_1, t_2 are terms
2. $R_i(t_1, \dots, t_n)$ where R_i is a relation symbol with arity n and all t_i are terms
3. $\neg\varphi, (\varphi \vee \psi), (\varphi \wedge \psi), (\varphi \implies \psi), (\varphi \iff \psi)$ where φ, ψ are formulas
4. $\exists v_i\varphi$ or $\forall v_i\varphi$ where φ is a formula and v_i is a variable symbol

The first two types of formulas (those without connectives and quantifiers) are called atomic formulas.

EXAMPLE 15.2 *WFFs in FOL*

Suppose we wish to translate the following statement to formal language: all humans are mammals. Since the statement includes a quantifier (all) we need a FOL language to express it formally. humans and mammals are then properties (or predicates, or unary relations), and we translate the statement as followings: $\forall x(\text{IsHuman}(x) \implies \text{IsMammal}(x))$.

To convince ourselves that it is indeed a WFF, we inspect the structure of the statement:

1. x is a variable symbol, so it is a valid term
2. IsHuman and IsMammal are relations and x is a term by 1, so by definition $\text{IsHuman}(x)$ and $\text{IsMammal}(x)$ are atomic formulas.
3. $(\text{IsHuman}(x) \implies \text{IsMammal}(x))$ is of the form $\varphi \implies \psi$ where φ, ψ are formulas by 2, so by definition it is also a formula.
4. Finally, $\exists x(\text{IsHuman}(x) \implies \text{IsMammal}(x))$ is of the form $\exists v_i \varphi$ where $v_i = x$ a term and φ is a formula by 3, so again by definition the whole statement is a WFF.

Let us now consider the vocabulary of number theory, and consider the following statement: $\forall x \exists y ((S(x) = y) \wedge (S(x) = (x + 1)))$, which expresses the idea that for every number, its successor is given by $x + 1$ and there exists a variable which we interpret as its successor.

Convince yourself that this statement isn't nonsense: the successor function and the addition operation with 1 should agree, otherwise one of the two does not operate as we would expect, and similarly we should also be able to represent the successor of a given number (variable) using a different number (variable) and not have to carry the successor function or addition operation in our calculations.

Now, let's show that this is indeed well formed:

1. $(x + 1)$ is a less formal way of writing $+(x, 1)$ which is a term since $+$ is a function symbol with arity 2, 1 is a constant symbol and x is a variable symbol.
2. $S(x)$ is a term since x is a variable symbol and S is a function symbol.
3. $(S(x) = (x + 1))$ is an atomic formula since both sides of the equality are terms by 1, 2.
4. $S(x) = y$ is an atomic formula since $S(x)$ is a term (by 2) and y is a term since it is a variable.
5. $(S(x) = y) \wedge (S(x) = (x + 1))$ is a formula since it is of the form $\varphi \wedge \psi$ where both φ, ψ are WFFs (by 3, 4).
6. $\exists y(S(x) = y) \wedge (S(x) = (x + 1))$ is a WFF since it is of the form $\exists v_i \varphi$ by 5.
7. Finally, $\forall x \exists y(S(x) = y) \wedge (S(x) = (x + 1))$ is of the form $\forall v_i \varphi$ by 6 so it is a WFF.

DEFINITION 0.3 *closed term and formula*

A term t in \mathcal{CPL} is called closed if it does not contain any variable symbol.

A formula φ in \mathcal{CPL} is called closed if it does not contain any variable symbol.

EXAMPLE 15.3 ⟨closed term and formula⟩

Given a vocabulary π with constant symbols a, b, c , variable symbols v_1, v_2 , relation symbols R, Q (both binary) and function symbols f, h, g (all unary), then:

1. $f(h(a))$ is a closed term since it is a term with no variable symbols
2. v_1 is a term that is not closed since it contains a variable symbol
3. $R(a, b)$ is a closed formula since it is a formula with no variable symbols
4. $f(a, v_1)$ is a term that is not closed since v_1 is a variable symbol
5. $R(f(a), g(h(c))) = R(h(a), a)$ is a closed formula since it is a formula which does not contain any variable symbol

Consider the WFF $\forall x(0 \leq x)$ in the vocabulary $\pi = \{0, \leq\}$ (convince yourself first that it is well-formed). To give meaning to the formula, we need to define what the symbols $0, \leq$ represent, and define the objects that the quantifier \forall ranges over. The latter is called the *domain of discourse* or *universe* of the language. We have used this term before by appealing to the reader's intuition, and now we formalize it:

DEFINITION 0.4 ⟨domain of discourse⟩

The domain of discourse, or universe, of a FOL, is the non-empty set of objects (or entities) that can be quantified by the quantifiers of the language. It is usually denoted by \mathbb{D} or D .

In the example we're currently discussing, if we take D to be \mathbb{N} and $0, \leq$ to be the additive identity on \mathbb{N} and the total order on \mathbb{N} , then $\forall x(0 \leq x)$ is clearly true. However, if we take D to be \mathbb{Z} with the usual notion for $0, \leq$, then the statement is false since, for example, $-1 \in \mathbb{Z}$ and 0 is strictly greater than -1 . These tuples - $(\mathbb{Z} : 0, \leq)$ and $(\mathbb{N} : 0, \leq)$ can both be used to provide meaning to the symbols in the vocabulary π . We call both *structures* of π , or π -structures.

DEFINITION 0.5 *⟨structure⟩*

Let $\pi = \{c_0, c_1, \dots, f_0, f_1, \dots, R_0, R_1, \dots\}$ be the vocabulary of some FOL, then a structure of π is a tuple \mathcal{M} consisting of:

1. A domain of discourse $D^{\mathcal{M}}$
2. A set of elements from $D^{\mathcal{M}}$ corresponding to the constant symbols in π , such that the corresponding element of c_i is denoted $c_i^{\mathcal{M}} \in D^{\mathcal{M}}$
3. A set of functions $f_i^{\mathcal{M}} : (D^{\mathcal{M}})^n \rightarrow D^{\mathcal{M}}$ where n is the arity of f_i and each $f_i^{\mathcal{M}}$ corresponds to some f_i symbol
4. A set of relations $R_i^{\mathcal{M}} : (D^{\mathcal{M}})^n \rightarrow \mathbb{B}$ where n is the arity of R_i and each $R_i^{\mathcal{M}}$ corresponds to some R_i symbol

$$\mathcal{M} = (D^{\mathcal{M}} : c_0^{\mathcal{M}}, c_1^{\mathcal{M}}, \dots, f_0^{\mathcal{M}}, f_1^{\mathcal{M}}, \dots, R_0^{\mathcal{M}}, R_1^{\mathcal{M}})$$

Structures provide us with a setting in which we can interpret and evaluate statements in predicate logic, let's see how:

DEFINITION 0.6 *⟨variable assignment function⟩*

Let \mathcal{L} be a FOL with structure \mathcal{M} and let V be the set of variable symbols of \mathcal{L} . Let $s : V \rightarrow D^{\mathcal{M}}$ where $D^{\mathcal{M}}$ is the domain of \mathcal{M} , we call s a variable assignment function.

This definition can immediately be extended to terms, in a similar manner to 0.16 truth assignment/e-valuation of WFFs :

DEFINITION 0.7 *⟨term assignment/interpretation⟩*

Now, let T be the terms of \mathcal{L} , C the constant symbols and F the function symbols, both with some ordering on the elements corresponding to the ordering on the corresponding elements in \mathcal{M} . We extend s to a homomorphism $\bar{s} : T \rightarrow D^{\mathcal{M}}$ as follows:

1. $\forall v \in V, \bar{s}(v) = s(v)$, i.e. \bar{s} restricted to V is simply s .
2. $\forall c_i \in C, \bar{s}(c_i) = c_i^{\mathcal{M}}$, i.e. \bar{s} takes constant symbols to the corresponding constant element in \mathcal{M} .
3. $\forall f_i \in F \forall t \in T^{\text{arity}(f_i)}, \bar{s}(f_i(t_1, \dots, t_n)) = f_i^{\mathcal{M}}(\bar{s}(t_1), \dots, \bar{s}(t_n))$

The reader is encouraged to verify that these 3 cases indeed generate all terms in \mathcal{L} , and to further verify that these cases freely generate the terms, so by Theorem 7 recursion theorem \bar{s} is well-defined and unique.

EXAMPLE 15.4

Let $\pi = \{1, \cdot\}$ (which is the vocabulary for the first order logic language of group theory) and consider the term $a \cdot b$. Without a structure for π there's little more we can make of this term on its own, but given a structure $\mathcal{M}(\mathbb{N} : 1, +)$ (where $1, +$ are the additive identity and binary addition operation on the natural numbers) and assignment $s : a \mapsto 2, b \mapsto 3$, then we get

$$\bar{s}(a \cdot b) = +(\bar{s}(a), \bar{s}(b)) = +(2, 3) = 5$$

Alternatively, given another structure $\mathcal{A} = (\text{GL}(3, \mathbb{R}) : I, \cdot)$ where $\text{GL}(3, \mathbb{R})$ is the group of 3×3 matrices with scalars from \mathbb{R} whose determinant is nonzero (this group is called the general linear group with $n = 3, \mathbb{F} = \mathbb{R}$), I is the identity matrix with 3 columns, and \cdot is matrix multiplication, and assignment $s : a \mapsto I, b \mapsto 3I$, then we get

$$\bar{s}(a \cdot b) = \cdot(I, 3I) = I \cdot 3I = 3(I \cdot I) = 3I$$

Finally, we can consider formulas, again this extension is similar to the one provided in our discussion of propositional logic in spirit, but now we also have to consider formulas with quantifiers over $D^{\mathcal{M}}$, which makes things a bit trickier.

DEFINITION 0.8 (truth-assignment for atomic formulas)

Let ψ be an atomic formula in a FOL language \mathcal{L} , and consider the interpretation of ψ in structure \mathcal{M} under variable assignment s , then:

1. If ψ is of the form $t_1 = t_2$ where t_1, t_2 are variables, then ψ evaluates to T if and only if $\bar{s}(t_1) = \bar{s}(t_2)$ (note that in this expression, equality is not just a symbol in the language \mathcal{L} , but an actual relation defined in \mathcal{M}).
2. If ψ is of the form $R_i(t_1, \dots, t_n)$, then ψ evaluates to true if and only if $R_i^{\mathcal{M}}(\bar{s}(t_1), \dots, \bar{s}(t_n)) = T$, where $R_i^{\mathcal{M}}$ is the corresponding relation $(D^{\mathcal{M}})^n \rightarrow \mathbb{B}$ to R_i in \mathcal{M} .

If ψ is not of either of these forms, it is not atomic.

While we can naturally extend this definition to include formulas involving connectives, we can't easily extend it to formulas involving quantifiers at the moment. For that, we revisit and extend a definition from predicate logic:

DEFINITION 0.9 *⟨satisfiability (revisited)⟩*

We extend our definition of satisfiability from 0.17 satisfiability to FOL.

Let \mathcal{L} be a FOL and consider a statement φ in structure \mathcal{M} with assignment s , then if φ evaluates to T in \mathcal{M} under s , we say that φ is satisfied in \mathcal{M} with assignment s and denote this relation via $\models_{\mathcal{M}} \varphi[s]$ or $\mathcal{M} \models_s \varphi$.

Note that this is synonymous to saying that φ is evaluated to T in \mathcal{M} under s .

DEFINITION 0.10 *⟨remapping⟩*

A remapping of a variable assignment function s is a new variable assignment function s' such that $s'(x) = s(x)$ for all x except for specified variable symbols which may be remapped to other values. We sometimes use a shorthand for s' and instead of defining it explicitly, we write $s[v_i \mapsto a_i, v_j \mapsto a_j, \dots]$ where v_i, v_j, \dots are the only variable symbols being remapped to elements a_i, a_j, \dots accordingly.

In the context of satisfiability, we denote that a remapping of s satisfies φ in \mathcal{M} via $\mathcal{M} \models_{s[v_1 \mapsto a_1, \dots]} \varphi$ or via $\models_{\mathcal{M}} \varphi[s[v_1 \mapsto a_1, \dots]]$.

Armed with these definitions, we can make sense of truth-assignment for formulas involving quantifiers, and can now proceed to define truth-evaluation of WFFs in general using structural recursion, as we did when defining term evaluation.

DEFINITION 0.11 *⟨truth-assignment for WFFs in FOL⟩*

Let φ be a WFF in a FOL language \mathcal{L} and consider its interpretation in \mathcal{M} with assignment s , then:

1. If φ is an atomic formula, it is evaluated according to 0.8 truth-assignment for atomic formulas
2. If φ is of the form $\neg\psi$ where ψ is a WFF, then it evaluates to true if and only if $\models_{\mathcal{M}} \psi[s]$ does not hold, i.e. does not evaluate to true (where evaluation of ψ is defined using this definition), same as with truth-evaluation of a formula of the same form in propositional logic.
3. If φ is of the form $\psi C \chi$ where C is one of the binary connectives $\vee, \wedge, \implies, \iff$ and both ψ, χ are WFFs, then φ is evaluated by the same rules as with similarly formed expressions in propositional logic (for example, if $C = \vee$, it is true if and only if both ψ and χ evaluate to true in \mathcal{M} under s).
4. If φ is of the form $\forall v \psi$, where ψ is a WFF and v is a variable symbol, then φ evaluates to true if and only if $\forall a \in D^{\mathcal{M}} (\mathcal{M} \models_{s[v \mapsto a]} \psi)$, i.e. every remapping of v under s satisfies ψ .
5. If φ is of the form $\exists v \psi$ where ψ is a WFF and v is a variable symbol, then φ evaluates to true if and only if $\exists a \in D^{\mathcal{M}} (\mathcal{M} \models_{s[v \mapsto a]} \psi)$, i.e. there exists a remapping of v under s that satisfies ψ .

Put more concisely, rules 4, 5 can be reworded as

$$\begin{aligned} (\mathcal{M} \models_s \forall v \psi) &\iff (\forall a \in D^{\mathcal{M}} (\mathcal{M} \models_{s[v \mapsto a]} \psi)) \\ (\mathcal{M} \models_s \exists v \psi) &\iff (\exists a \in D^{\mathcal{M}} (\mathcal{M} \models_{s[v \mapsto a]} \psi)) \end{aligned}$$

This way we can convert statements with quantifiers to statements not involving quantifiers, and use rules 1 – 3 to evaluate those statements.

Note that we do not impose any syntactic limitation on when variables appearing in a formula should be quantified, however if a variable is quantified then the act of quantifying it changes how we interpret that variable in the rest of the formula (since we define the interpretation of quantified formula by evaluating un-quantified formulas with the quantified variable re-assigned, or re-interpreted). This distinction motivates a definition:

DEFINITION 0.12 *⟨free and bound variables⟩*

Let φ be a WFF and consider the variables used in φ . We categorize them into two distinct categories:

1. bound variables - variables which are quantified
2. free variables - variables which are not bound, i.e. are not quantified

If φ has no free variables, we call φ a sentence.

We also introduce a new notation for expression a formula - φ can also be written as $\varphi[v_1, v_2, \dots](x_1, x_2, \dots)$ where v_1, v_2, \dots are the bound variables of φ and x_1, x_2, \dots are the free variables of φ . If φ has no bound variables, we omit the square brackets, and if φ has no free variables, we omit the parenthesis. Note that this notation is not part of the formal language.

THEOREM 16

Let φ be a WFF in FOL \mathcal{L} with vocabulary π in structure \mathcal{M} and let V be the set of free variables of φ . Let s_1, s_2 be two variable assignments such that $\forall v \in V (s_1(v) = s_2(v))$ i.e. s_1, s_2 coincide on the free variables of φ , then:

$$\mathcal{M} \models_{s_1} \varphi \iff \mathcal{M} \models_{s_2} \varphi$$

Proof. Let s_{12} be the restriction on s_1, s_2 to V . We prove by induction on φ . If φ consists of a single symbol, then it has no quantifiers so clearly all variables are free, so s_{12} interprets all variables in φ so clearly either both s_1, s_2 satisfy φ or neither, which covers the induction base.

Now suppose φ has length $n1$, so it must be in one of the following forms:

1. φ is an atomic formula
2. φ is given by $\neg\psi$ where ψ is a WFF
3. φ is given by $\psi C \chi$ where ψ, χ are WFFs and C is a binary connective
4. φ is given by $\forall v \psi$ where v is a variable symbol and ψ is a WFF
5. φ is given by $\exists v \psi$ where v is a variable symbol and ψ is a WFF

case 1: if φ is an atomic formula then it does not contain any qualifiers so all variables are free so $\overline{s_1}$ and $\overline{s_2}$ agree on all terms involved in φ , and since we assume the same structure \mathcal{M} then it follows that both must agree on the truth assignment of φ (this can also be shown by induction if one finds this argument insufficient).

case 2, 3: by the induction hypothesis the theorem holds for ψ (in case 2) and ψ, χ (in case 3) so by definition of truth-assignment to WFFs with connectives it follows that the theorem holds for $\neg\psi$ and $\psi C \chi$ as well.

case 4, 5: Recall that $\mathcal{M} \models_s \forall v \psi$ if and only if $\forall v \in D^{\mathcal{M}} (\mathcal{M} \models_{s[v \mapsto a]} \psi)$, so the free variables of ψ are those in φ with the addition of v , but since we consider remapping of v , then $\forall a \in d^{\mathcal{M}}$, the remappings $s_1[v \mapsto a]$ and $s_2[v \mapsto a]$ are in agreement on v , and by the assumption of the theorem they are in agreement for all other free variables of ψ , so they are in agreement for all free variables of ψ . Now by the induction hypothesis, the theorem holds for ψ and for each v , so by definition s_1, s_2 either both satisfy $\forall v \psi$, or they both do not. A similar argument can be produced for \exists .

□

COROLLARY 16.1 *⟨model in FOL⟩*

If φ is a sentence (i.e. φ has no free variables) in FOL \mathcal{L} in structure \mathcal{M} , then the truth-assignment of φ is independent of the choice of variable assignment s , i.e. it is always either true or false for every assignment s in $D^{\mathcal{M}}$. In this case, we write $\mathcal{M} \models \varphi$, or $\models_{\mathcal{M}} \varphi$, and we say that \mathcal{M} is a model of φ .

Proof. Consider S the set of all variable-assignment from V the variables of \mathcal{L} to $D^{\mathcal{M}}$. Since φ is a statement, it has no free variables so all elements in S vacuously agree on the free variables of φ . Now consider two assignments s_1, s_2 . By 0.12 free and bound variables $(\mathcal{M} \models_{s_1} \varphi) \iff (\mathcal{M} \models_{s_2} \varphi)$. Since this holds for every s_1, s_2 then either $\forall s \in S (\mathcal{M} \models_s \varphi)$, or $(\forall s \in S (\mathcal{M} \not\models_s \varphi))$, which completes the proof

□

This is a generalization of 0.21 model, which also had variable interpretations (to either T or F , via a truth-assignment function), but didn't have a domain of discourse, function or relation interpretations, since these are irrelevant in propositional logic.

Let's turn our attention to tautological implications in predicate logic. Given a set of WFFs ϕ and a WFF φ , we said that ϕ tautologically implies φ (and we denoted this relation as $\phi \models \varphi$) if and only if every truth assignment f that satisfies all of ϕ also satisfies φ , i.e. if and only if the truth table for $\bigwedge_{\psi \in \phi} \psi$ is true in all places where the truth table of φ is also true. Now suppose ϕ, φ are stated in a FOL language. In a FOL, we don't have a truth assignment function for the variables but instead have variable assignments s and a structure \mathcal{M} , but we still can draw semantical implications: recall the motivating example for our discussion of FOL, which can now be expressed with vocabulary $\pi = \{x, p, \text{IsMammal}, \text{IsHuman}\}$ where x is a variable symbol, p is a constant symbol, and $\text{IsMammal}, \text{IsHuman}$ are unary relation symbols:

$$\{\forall x (\text{IsHuman}(x) \implies \text{IsMammal}(x)), \text{IsHuman}(p)\} \models \text{IsMammal}(p)$$

We know this implication is valid since the first formula means that $\text{IsHuman}(x) \implies \text{IsMammal}(x)$

for every x , in particular for $x = p$, so by *MP* on 1, 2 we know that the conclusion is implied by the premise. To formalize this idea, we introduce *logical implication*, which is a form of semantical implication for FOL, not to be confused with tautological implication (and tautologies) which define semantical implication in propositional logic.

DEFINITION 0.13 *⟨logical implication⟩*

Let ϕ be a set of WFFs and let φ be a WFF, then we say that φ is logically implied by ϕ , denoted $\phi \models \varphi$, if and only if for every structure \mathcal{M} of the vocabulary the WFF is expressed in and for every variable assignment s , $(\models_{\mathcal{M}} \phi[s]) \implies (\models_{\mathcal{M}} \varphi[s])$. We denote this implication by $\phi \models \varphi$.

Note that when $\phi \cup \{\varphi\}$ are all sentences, we can omit s from the definition as per Theorem 16.

EXAMPLE 16.1 *⟨logical implications⟩*

1. Consider the formulas $\forall x R(x)$ and $R(y)$, and consider a structure-assignment pair (\mathcal{M}, s) such that $\models_{\mathcal{M}} \forall x R(x)[s]$, then by definition $\forall a \in D^{\mathcal{M}} (\models_{\mathcal{M}} R(x)[s[x \mapsto a]])$, and in particular for $x \mapsto s(y)$, which implies $\models_{\mathcal{M}} R(y)[s]$, so $\forall x R(x) \models R(y)$.

2. Consider $\forall y R(y)$ and $\exists x R(x)$. By the same argument as 1 we know that the former means that $\forall a \in D^{\mathcal{M}} (\models_{\mathcal{M}} R(y)[y \mapsto a])$, and since $D^{\mathcal{M}}$ is non-empty then $\exists a \in D^{\mathcal{M}}$ so that $\models_{\mathcal{M}} R(y)[y \mapsto a]$, so by a similar argument to 1 and by definition of truth-assignment for formulas involving the existential quantifier, it follows that $\forall y R(y) \models \exists x R(x)$.

DEFINITION 0.14 *⟨logical equivalence⟩*

Let φ, ψ be WFFs, then φ and ψ are logically equivalent if $\varphi \models \psi \wedge \psi \models \varphi$. We denote this relation by $\varphi \equiv \psi$.

EXAMPLE 16.2 *⟨logical equivalence⟩*

From definition we can show that $(\varphi \implies \psi) \equiv (\neg\psi \implies \neg\varphi)$. Happily this is an expected result, since it is a generalization of a tautology from propositional logic to a logical equivalence in FOL.

A very useful logical equivalence is $\neg(\forall v \psi) \equiv \exists v(\neg\psi)$.

LEMMA 16.1

Let ϕ be a set of WFFs and let ψ be a WFF, then:

$$\phi \models \psi \iff \phi \cup \{\neg\psi\} \text{ is not satisfiable}$$

Proof. \implies Let ϕ be a set of WFFs and let ψ be a WFF, further suppose that $\phi \models \psi$. Now let \mathcal{M} be a structure and let s be a variable assignment, then by 0.13 logical implication ($\mathcal{M} \models_s \phi$) \implies ($\mathcal{M} \models_s \psi$), so every structure and every assignment that satisfy ϕ also satisfy ψ , so $\phi \cup \{\neg\psi\}$ is not satisfiable, otherwise we would have a structure and assignment pair that satisfies ϕ but not ψ .

\impliedby Let $\phi \cup \{\neg\psi\}$ be not satisfiable and take a structure-assignment pair (\mathcal{M}, s) that satisfy ϕ (if non exists, that the statement $\phi \models \psi$ is vacuously true), but since $\phi \cup \{\neg\psi\}$ is not satisfiable then (\mathcal{M}, s) does not satisfy it, but it does satisfy ϕ is it must be that $\not\models_{\mathcal{M}} \neg\psi[s]$, but then $\models_{\mathcal{M}} \psi[s]$, and this is true for every (\mathcal{M}, s) that satisfies ϕ , so $\phi \models \psi$.

□

THEOREM 17

It is possible to discard \exists from our language without losing expressive power, by replacing WFFs of the form $\exists v\varphi$ with $\neg\forall v\neg\varphi$.

Proof. We prove this by showing that $\exists v\varphi \equiv \neg\forall v\neg\varphi$, and it is left as an exercise.

□

We also define a tautology in FOL as a proper extension of a tautology in propositional logic:

DEFINITION 0.15 **⟨tautology (FOL)⟩**

A tautology in FOL is obtained from a tautology φ in propositional logic by replacing the propositional variables in the tautology with WFFs in FOL. A tautology is a WFF that is true in any model and any assignment.

EXAMPLE 17.1 **⟨tautology (FOL)⟩**

Consider the propositional tautology (verify by checking the truth table if you're not convinced) $(A \implies \neg B) \implies (B \implies \neg A)$. By identifying A with $\forall vP(v)$ and B with $\exists uQ(u, x)$ we can produce the following FOL tautology:

$$(\forall vP(v) \implies \neg \exists uQ(u, x)) \implies (\exists uQ(u, x) \implies \neg \forall vP(v))$$

Proof System

We turn our attention to the axiomitization of a formal proof system for FOL. The formal proof system we use is an extension of the proof system presented for propositional logic. After presenting the new axioms of the proof system, we will prove the soundness and completeness theorems (known as Godel's completeness theorem) for that system, and discuss some implications of Godel's completeness theorem.

Recall that a proof system (or a deduction system) consists of a set of logical axioms and a set of inference rules. In our discussion of propositional logic, we provided a proof system with 3 axiom schemas and MP as the only rule of inference, and showed it was a sound and complete system. Let's quickly revisit the definitions of soundness and completeness in FOL

DEFINITION 0.16 **<soundness and completeness (FOL)>**

Let ϕ be a set of WFFs and let φ be a WFF, in some FOL \mathcal{L} , then:

1. $\phi \vdash \varphi$ means that φ is syntactically implied from ϕ in P , i.e. it is provable from ϕ in P
2. $\phi \models \varphi$ means that φ is logically implied by ϕ .

A proof system P is sound if for every ϕ, φ , $(\phi \vdash \varphi) \implies (\phi \models \varphi)$.

A proof system P is complete if for every ϕ, φ , $(\phi \models \varphi) \implies (\phi \vdash \varphi)$.

Consider the logical implication $\forall x R(x) \models R(y)$, presented in 16.1 logical implications. Clearly it cannot be derived from the axiom schemas of 0.40 Łukasiewicz's P_2 (which provide a method for generating *tautologies* in FOL via 0.15 tautology (FOL)), since these schemas have no way of *discarding* a qualifier (remember, proof systems should be purely syntactical). Now consider another logical implication: $R(y) \implies \exists x R(x)$ (verify by definition that this is indeed a logical implication), again we cannot derive it since we have no way of *introducing* a qualifier. Finally, we also have no hope to show that $x = x$ is a tautology since we don't have an equality symbol within the language of propositional logic.

Let's consider $\forall x R(x) \models R(y)$ again. The RHS is formed by discarding the qualifier in the LHS, and substituting x for y . Notice that the LHS has the structure $\forall x \varphi$ where φ is a WFF such that x is free (in this example, φ is $R(x)$ which has x as a free variable), and the RHS is simply φ with y substituting x . Furthermore, we can also replace x with a *term* - clearly if $\forall x R(x)$ then in particular for $f(a, b, c, \dots)$ which is in the domain of discourse we have $R(f(a, b, c, \dots))$, so this substitution is also valid and interesting. Let's formalize this idea

DEFINITION 0.17 **<substitution>**

Let φ be a WFF with a variable x appearing free (or not at all), then a substitution of φ , denoted φ_t^x , is obtained from φ by replacing every occurrence of x with the term t .

We have shown a couple of examples in the previous paragraph.

Before we attempt to prove that $\forall x\varphi \models \varphi_t^x$, let's consider $\forall x\exists yR(x, y)$ which matches the structure $\forall x\varphi$, and consider the interpretation of the formula in the structure $(\mathbb{N},)$ (i.e. the domain of discourse is the natural numbers and R is the relation), then the formula states “for every number, there exists a larger number”. Now consider the substitution φ_y^x , which gives us $\exists yR(y, y)$ which means “there exists a number greater than itself”, which is a contradiction.

The issue here is that now y is not free, so we substituted a free variable with a bound variable. We would also run into the same issue if we substitute x for a term which includes y , for example with $y + 1$ (because then we'd have a statement that claim “there exists a number y such that y is greater than its successor”, which is also clearly false). This observation motivates the following definition:

DEFINITION 0.18 *⟨substitutable term⟩*

Let φ be a WFF and let t be a term, then t is substitutable for some free variable x in φ if and only if t does not contain a variable y that is bound in φ .

EXAMPLE 17.2 *⟨substitutable term⟩*

We have already seen a couple of basic examples, so let's consider a more involved case: consider the formula φ given by $\forall z(z = x) \vee \exists x(y = f(x, y))$. Consider the following questions:

1. Can $f(z, 3)$ substitute x ?
2. Can $f(z, 3)$ substitute y ?
3. Can $f(z, 3)$ substitute z ?

To answer these questions, perhaps it would be better to express φ more formally using parenthesis to emphasize the scope of each quantifier: φ is given by $(\forall z(z = x)) \wedge (\exists x(y = f(x, y)))$, so the sub-formula involving z is $\forall z(z = x)$, which has x free, however it also has z bounded by the universal quantifier, and since z appears in $f(z, 3)$ it cannot substitute x in that sub-formula, so it cannot substitute z in ψ , which answers 3. As for y , it only appears in the sub-formula $\exists x(y = f(x, y))$, where it is free, and similarly z is not bound in that sub-formula, so we can substitute $f(z, 3)$ for y in that sub-formula and in φ , so the answer to 2 is yes. Finally, x is bound by the second sub-formula so it cannot be substituted (and also z is bound in the first sub-formula).

In conclusion, the answer to the questions is:

1. No
2. Yes
3. No

DEFINITION 0.19 (universal instantiation)

Universal instantiation is an axiom schema given by

$$\forall x\varphi \implies \varphi_t^x$$

Where t is substitutable for x in φ .

PROPOSITION 7

Universal instantiation is valid.

Proof. Let φ be a WFF where x is a free variable and let t be a term substitutable for x in φ . Consider $\forall x\varphi \implies \varphi_t^x$. It evaluates to false only when $\forall x\varphi$ is true and φ_t^x is false (by definition of the truth table of \implies), so suppose $\forall x\varphi$ is true, and let (\mathcal{M}, s) be a structure-assignment pair for the FOL. Since $\forall x\varphi$ is true, then by definition $\forall a \in D^{\mathcal{M}} (\mathcal{M} \models_{s[x \mapsto a]} \varphi)$. Since t is a term, it is assigned a variable $t^{\mathcal{M}}$ after evaluation, so in particular for $a = t^{\mathcal{M}}$ we have $\mathcal{M} \models_{s[x \mapsto t^{\mathcal{M}}]} \varphi$. Now suppose t does not have a variable which is bound by φ , then instead of remapping s we can substitute t for x and get an equivalent statement (the restriction on t to not include bound variables is due to the fact that otherwise the statements aren't equivalent), so we have $\mathcal{M} \models_s \varphi_t^x$, and since we considered any \mathcal{M} and any s , we have shown that $\forall x\varphi \implies \varphi_t^x$ (when substitution is allowed) is always true, so universal instantiation is a valid axiom schema \square

COROLLARY 17.1 (instantiation without substitution)

The following follows from universal instantiation:

$$\forall x\varphi \implies \varphi$$

Proof. Consider φ_x^x , since x occurs free in φ then it can be substituted, now consider x as a substitution term - it does not contain any variable which is bound in φ , so x is substitutable for x in φ , so by universal instantiation $\forall x\varphi \implies \varphi_x^x$, but φ_x^x is just φ , so $\forall x\varphi \implies \varphi$ \square

To let our proof system introduce quantifiers, we add a new inference rule:

DEFINITION 0.20 (universal generalization)

Universal generalization, denoted *GEN*, is an inference rule which states that if $\phi \vdash \psi$ then $\phi \vdash \forall x\psi$ if x does not occur free in ϕ . (note that if ϕ is a set of sentences then this restriction is always satisfied, further note that there is no restriction on x appearing in ψ).

PROPOSITION 8

Universal generalization is sound.

Proof. Suppose $\phi \models \psi$, so for every structure-assignment pair (\mathcal{M}, s) it follows that $(\models_{\mathcal{M}} \phi[s]) \implies (\models_{\mathcal{M}} \psi[s])$. Now let $a \in D^{\mathcal{M}}$ and consider the remapping $s[x \mapsto a]$. Since $s[x \mapsto a]$ is an assignment and since x does not appear free in ϕ then the proposition is true for the structure-assignment pair $(\mathcal{M}, s[x \mapsto a])$ by assumption as well, so we have $\models_{\mathcal{M}} \phi[s[x \mapsto a]]$, but since this is true for every a we have $\forall a \in D^{\mathcal{M}} (\models_{\mathcal{M}} \phi[s[x \mapsto a]])$, and since we suppose $\phi \models \psi$ it follows that $\forall a \in D^{\mathcal{M}} (\models_{\mathcal{M}} \psi[s[x \mapsto a]])$, which by definition means that $\forall x\psi$ is true in (\mathcal{M}, s) , but since our choice of \mathcal{M} and s was unrestricted, it follows that $\forall x\psi$ is always true under our assumption, so universal generalization is sound □

Finally, we need another axiom or rule that will let us move quantifiers in a formula, otherwise we would not be able to derive formulas such as $(\forall x(P(y) \implies P(x))) \implies ((P(y) \implies \forall xP(x)))$, even though they are always true (intuitively this holds because x does not appear free in $P(y)$ so quantifying x over $P(y)$ is inconsequential, formally this requires a proof which is left for the reader), and we also need axioms for equality, since we should be able to verify that $x = x$ always holds and that if $x = y$ then all terms which are identical except for x, y are also equal and that propositions involving such terms are equivalent (since if $a = b$ then also $a + 3 = b + 3$ and $R(a, 1) \implies R(b, 1)$). We take all of this into account in the following definition of our formal proof system:

DEFINITION 0.21 *⟨formal proof system of FOL⟩*

Let \mathcal{L} be a FOL. Our formal proof system consists of the following deduction rules:

1. MP (modus ponens)
2. GEN (universal generalization)

And the following set of axiom schemas (A1 – A3 are the axioms of propositional logic):

- A1. $\varphi \implies (\psi \implies \varphi)$
- A2. $(\varphi \implies (\psi \implies \chi)) \implies ((\varphi \implies \psi) \implies (\varphi \implies \chi))$
- A3. $(\neg\psi \implies \neg\tau) \implies (\tau \implies \psi)$
- A4. $\forall x\varphi \implies \varphi_t^x$ (0.19 universal instantiation)
- A5. $\forall x(\varphi \implies \psi) \implies (\varphi \implies \forall x\psi)$ where x does not occur free in φ .

As well as the following axiom schemas concerning equality:

- E1. $x = x$
- E2. $(x = y) \implies (t = s)$ where t, s are terms such that s is produced from t by substituting x for y in one or more places
- E3. $(x = y) \implies (\varphi \implies \psi)$ where φ, ψ are WFFs such that ψ is produced from φ by substituting x for y in one or more places

Now that the system has been presented, we can start exploring it to produce metatheorems about the system. Our main goal here is to eventually prove that this system is sound and complete.

THEOREM 18 *⟨deduction theorem in FOL⟩*

The deduction theorem is valid in our proof system for FOL as well:

$$(\phi \cup \{\varphi\} \vdash \psi) \implies \phi \vdash (\varphi \implies \psi)$$

Proof. We base our proof on Theorem 10 deduction theorem in P_2 . The base case is the same, however now when $n=1$ it is also possible that ψ is inferred from GEN, i.e. ψ is of the form $\forall x\chi$ where χ is a WFF, and since χ is shorter than ψ by the induction hypothesis $\phi \cup \{\varphi\}$ implies $\varphi \implies \chi$, and since we assume that φ is deduced from χ via GEN which is deduced from $\phi \cup \{\varphi\}$, so x does not occur free in φ (otherwise GEN can't be used), so we can apply A5 and get:

1. $\varphi \implies \chi$ (hypothesis)
2. $\forall x(\varphi \implies \chi) \implies (\varphi \implies \forall x\chi)$ (A5, by the induction hypothesis x does not occur free in φ)
3. $\varphi \implies \forall x\chi$ (MP on 1, 2)
4. $\varphi \implies \psi$ (since ψ is just $\forall x\chi$ by assumption)

All other cases are already handled in the original proof from propositional logic.

□

THEOREM 19

The following WFFs are always deducible (i.e. if φ is the formula, then $\vdash \varphi$):

1. $P(x) \implies (\exists y P(y))$
2. $\forall x(\varphi \implies \psi) \implies (\forall x \varphi \implies \forall x \psi)$
1. $\forall x(P(x) \implies \exists y P(y))$

Proof. 1. It is good to start by understanding the proposition: “if x satisfies a relation then there exists some y which satisfies this relation”, which is obvious to us semantically, however a formal proof is syntactical. The proof “builds itself” in the sense that it is trivial to know which axioms are needed: first, since we don’t have an axiom that introduces the existential qualifier directly without negation, it is obvious that we need to instantiate $A3$ (since $\neg \exists v \varphi$ is $\forall v \neg \varphi$). Furthermore, the formula we wish to prove is of the form $\varphi \implies \psi$, so it makes sense for us to use $A3 + MP$ to infer $\varphi \implies \psi$. By taking φ to be $P(x)$ (whose negation is $\neg P(x)$) and ψ to be $\exists y P(y)$ (whose negation is $\forall y \neg P(y)$), we:

$$1. ((\forall y \neg P(y)) \implies \neg P(x)) \implies (P(x) \implies (\exists y P(y))) (A3)$$

$\forall y \neg P(y) \implies \neg P(x)$ is of the form $\forall v \varphi \implies \varphi_t^v$ where $t = x$ and $v = y$ and $\varphi = \neg P(y)$, and this substitution is valid since y is free in $\neg P(y)$ and x is not bound in $P(y)$, so by $A4$:

2. $\forall y \neg P(y) \implies \neg P(x) (A4)$
3. $P(x) \implies (\exists y P(y)) (MP \text{ on } 1, 2)$

Which completes the proof.

2. We read the formula as “if for every x , A implies B , then A holding for every x implies that B holds for every x ”. We prove 2 is deducible as follows: First, we suppose $\forall x(\varphi \implies \psi) (H1)$ and $\forall x \varphi (H2)$, then:

1. $\forall x(\varphi \implies \psi) \implies (\varphi \implies \psi) (A4 \text{ by } 17.1 \text{ instantiation without substitution})$
2. $\forall x \varphi \implies \varphi (A4 \text{ by } 17.1 \text{ instantiation without substitution})$
3. $\varphi \implies \psi (MP \text{ on } 1, H1)$
4. $\varphi (MP \text{ on } 2, H2)$
5. $\psi (MP \text{ on } 3, 4)$
6. $\forall x \psi (GEN \text{ on } 5)$

Which shows that $\{\forall x(\varphi \implies \psi), \forall x \varphi\} \vdash \forall x \psi$. Now by Theorem 18 deduction theorem in FOL we know that this implies $\forall x(\varphi \implies \psi) \vdash (\forall x \varphi \implies \forall x \psi)$, and again by the deduction theorem we get $\vdash \forall x(\varphi \implies \psi) \implies (\forall x \varphi \implies \forall x \psi)$, which completes the proof.

Formula (2) is pretty similar to $A5$, although it is more relaxed (it does not require x to occur free in φ) and has a bit of a different structure (the qualifier is expanded to both formulas). In some proof

systems, it is given as an axiom instead of A5.

3. Since this is essentially just $\forall v\varphi$ where φ is the previous formula, it is tempting to use GEN on φ : by 1 we know $\vdash \varphi$, so by GEN $\{\varphi\} \vdash \forall x\varphi$. However, this deduction is incorrect - Since φ is $P(x) \implies (\exists yP(y))$, then x is free in φ , so GEN cannot be applied. However, in the formulas in steps 1, 2 are axioms, so GEN can be applied on both, so we start our proof with 1, 2 and then continue with GEN:

1. $((\forall y\neg P(y)) \implies \neg P(x)) \implies (P(x) \implies (\exists yP(y)))$ (A3)
2. $\forall y\neg P(y) \implies \neg P(x)$ (A4)
3. $\forall x((\forall y\neg P(y)) \implies \neg P(x)) \implies (P(x) \implies (\exists yP(y)))$ (GEN on 1)
4. $\forall x(\alpha \implies \beta)$ (4 with a change of notation, α means formula 2 and β is the RHS of formula 1)
5. $\forall x(\forall y\neg P(y) \implies \neg P(x))$ (GEN on 2)
6. $\forall x\alpha$ (6 with a change of notation)
7. $\forall x(\alpha \implies \beta) \implies (\forall x\alpha \implies \forall x\beta)$ (deducible by the proof for (2))
8. $\forall x\alpha \implies \forall x\beta$ (MP on 5, 8)
9. $\forall x\beta$ (MP on 7, 9)
10. $\forall x(P(x) \implies \exists yP(y))$ (10, substituting β for the explicit formula it represents)

□

THEOREM 20 (tautology provability)

Let Λ be the set of logical axioms of the proof system (i.e. all possible instantiations of the axiom schemas), then

$$(\phi \vdash \varphi) \iff ((\phi \cup \Lambda) \models \varphi)$$

Proof. \implies Take a model \mathcal{M} for $\phi \cup \Lambda$ (if none exists then this is vacuously true), and let φ be a theorem in ϕ , i.e. $\phi \vdash \varphi$. We prove by induction on the length of the proof for $\phi \vdash \varphi$: if $n = 1$, then either $\varphi \in \phi$, or φ is an instantiation of an axiom so $\varphi \in \Lambda$, so in all cases $\varphi \in (\phi \cup \Lambda)$, so clearly $(\phi \cup \Lambda) \models \varphi$.

For $n > 1$, φ must be deduced either via MP, or by GEN. In the case of MP, then we have proofs for $\psi \implies \varphi$ and ψ which are included in the proof for φ , so by the induction hypothesis $(\phi \cup \Lambda) \models \{\psi \implies \varphi, \psi\}$, and we have seen earlier that this implies that $(\phi \cup \Lambda) \models \varphi$, as desired. If φ is deduced via GEN, then it is of the form $\forall v\psi$ where v is not bound in any of the formulas of ϕ , and similarly by the induction hypothesis this means that $(\phi \cup \Lambda) \models \psi$. Now, consider $\forall v\psi$ - it is satisfied in \mathcal{M} if for every assignment to v , ψ is true, but since \mathcal{M} is a model for $\phi \cup \Lambda$ then it satisfies all formulas for all possible assignment, in particular to all possible assignment to v , and from the induction hypothesis it models ψ as well, so by this argument it also models $\forall v\psi$, which is φ .

\Leftarrow Let Suppose $(\phi \cup \Lambda) \models \varphi$, and consider the formulas in propositional logic. By Theorem 15 compactness theorem for P_2 , this means that there exists a finite subset $\Gamma \subseteq \phi \cup \Lambda$ such that $\Gamma = \{\gamma_1, \gamma_2, \dots, \gamma_n\}$ tautologically implies φ , i.e. $\Gamma \models \varphi$. Suppose that φ is not in $\phi \cup \Lambda$, then it is also not in Γ , so it must be that $\gamma_1 \Rightarrow \gamma_2 \Rightarrow \dots \Rightarrow \gamma_n \Rightarrow \varphi$ is a tautology and is in Λ , so it can be derived from φ syntactically, so then by repeatedly applying MP we eventually derive φ .

□

THEOREM 21 (soundness theorem of the proof system for FOL)

The proof system for FOL is sound, i.e. if $\phi \vdash \varphi$ then $\phi \models \varphi$.

Proof. In Theorem 8 soundness of the proof system P_2 we have shown an inductive argument for the soundness of P_2 . The base case remains the same, and we only need to consider inference of φ from GEN in the inductive step. The outline of the proof for this case is the same as the proof for that case in Theorem 20 tautology provability, so the details are omitted here

□

THEOREM 22 (rule T in FOL)

Suppose $\alpha = \{\alpha_1, \alpha_2, \dots, \alpha_n\}$ are syntactically implied by ϕ , i.e. $\phi \vdash \alpha_1, \phi \vdash \alpha_2, \dots, \phi \vdash \alpha_n$, and suppose $\alpha \models \beta$, i.e. β is tautologically implied by α , then $\phi \vdash \beta$.

Proof. It follows from the statement of the theorem that $\alpha_1 \Rightarrow \alpha_2 \Rightarrow \dots \Rightarrow \alpha_n \Rightarrow \beta$ is a tautology, then by Theorem 20 tautology provability it follows that $\phi \vdash (\alpha_1 \Rightarrow \alpha_2 \Rightarrow \dots \Rightarrow \alpha_n \Rightarrow \beta)$, but also by assumption $\phi \Rightarrow \alpha_i$ for all $\alpha_i \in \alpha$, then by applying MP exactly n times we can deduce β

□

Godel's Completeness Theorem

Godel's completeness theorem establishes the (semantical) completeness of for FOL. We follow the proof provided by Loen Henkin (and not Godel's proof). Before we begin, let's recall the key steps of our semantical completeness proof for propositional logic:

1. We show that every set of formulas ϕ can be extended to a maximally consistent set, i.e. a set ϕ^* where $\forall \varphi \notin \phi^*, \phi^* \cup \{\varphi\}$ is inconsistent.
2. We show that for every maximally consistent set, we can find a model \mathcal{M} . A model in the context of propositional logic is just a truth assignment to the propositional variables which satisfies all WFFs in the set. This is called a *model existence theorem*.
3. We prove by contradiction that if ϕ is satisfiable then it is complete.

In this section, we will show how these ideas generalize to FOL. Indeed, the general structure of the proof for Godel's completeness theorem is the same, but the main difficulty here is in finding a model for a maximally consistent set: in the case of propositional logic, we could easily build a model by asking for each propositional variable v , is $v \in \phi^*$? if the answer was yes, we assigned v to T in our model, otherwise we assigned v to false. However in first order logic, our model has to provide:

1. a domain of discourse
2. an interpretation for the constants
3. an interpretation for relations
4. an interpretation for functions

To model φ^* , we will use the *term model*. First, we define the following equivalence relation:

DEFINITION 0.22 (closed term equivalence relation)

Let T be the set of closed terms in a language \mathcal{L} with vocabulary π , and let ϕ be a consistent set of sentences in \mathcal{L} , then we define a binary relation E on T given by

$$E : T \times T \rightarrow \mathbb{B}, (s, t) \mapsto (\phi \vdash (s = t))$$

and we denote sEt if and only if $E(s, t) = T$, otherwise we denote $s \not Et$.

PROPOSITION 9

E is an equivalence relation on T .

Proof. To show that E is an equivalence relation, we need to show that it is:

1. Reflexive, i.e. $\forall t \in T, tEt$
2. Symmetric, i.e. $\forall s, t \in T, tEs \implies sEt$
3. Transitive, i.e. $\forall s, t, w \in T, (sEt \wedge tEw) \implies sEw$

We will make use of the equality axioms defined in 0.21 formal proof system of FOL.

1. (reflexivity)

1. $x = x$ (E1)
2. $(x = x) \implies (t = t)$ (E2)
3. $t = t$ (MP on 1, 2)

2. (symmetry) We wish to show $(\phi \vdash (t = s)) \implies (\phi \vdash (s = t))$, then suppose $\phi \vdash (t = s)$. We shall prove $s = t$:

1. $(x = y) \implies ((x = x) \implies (y = x))$ (E3 with $\varphi = (x = x), \psi = (y = x)$)
2. $\forall y((x = y) \implies ((x = x) \implies (y = x)))$ (GEN on 1)
3. $\forall x \forall y((x = y) \implies ((x = x) \implies (y = x)))$ (GEN on 2)

4. $\forall x \forall y ((x = y) \implies \forall y ((x = x) \implies (y = x))) \implies \forall x ((x = s) \implies ((x = x) \implies (s = x)))$
(A4 with $\varphi = (3)$, note that y occurs free in (3))
5. $\forall x ((x = s) \implies ((x = x) \implies (s = x)))$ (MP on 3, 4)
6. $\forall x ((x = s) \implies ((x = x) \implies (s = x))) \implies ((t = s) \implies ((t = t) \implies (s = t)))$ (A4 with $\varphi = 5$, note that x occurs free in (3))
7. $(t = s) \implies ((t = t) \implies (s = t))$ (MP on 5, 6)
8. $t = s$ (deducible, hypothesis)
9. $(t = t) \implies (s = t)$ (MP on 7, 8)
10. $t = t$ (deducible, by reflexivity proof)
11. $s = t$ (MP on 8, 9)

Note that in the proof for symmetry, we have also proven a nice property which can be used to shorten formal proofs:

LEMMA 22.1 *⟨generalization and instantiation lemma⟩*

Given:

1. $\phi \vdash \varphi$
2. $V = (v_1, v_2, \dots, v_n)$ is a set of variables which are not bound in φ
3. $T = (t_1, t_2, \dots, t_n)$ is a set of terms such $\varphi_{t_i}^{v_i}$ is a proper substitution (i.e. each t_i is substitutable for v_i) for φ .

Denoting ψ as $((\dots ((\varphi_{t_n}^{v_n})_{t_{n-1}}^{v_{n-1}}) \dots)_{t_1}^{v_1})$, i.e. we as φ such that all v_i variables have been substituted with t_i terms, then:

1. $\phi \vdash \forall v_1 \forall v_2 \dots \forall v_n \varphi$ (simply by applying GEN n times)
2. $\varphi \vdash \psi$ (by repeated application of A4 and MP)
3. $\phi \vdash \psi$ (By MP on 1, 2)

The proof for 2 in the lemma is a generalization of steps 1 – 7 provided above, and is left as an exercise for the reader.

3. (transitivity) suppose $\phi \vdash \{(s = t), (t = v)\}$, we shall prove $s = v$:

1. $(x = y) \implies ((y = z) \implies (x = z))$ (E3 with $\varphi = (y = z)$ and $\psi = (x = z)$)
3. $(t = s) \implies ((t = v) \implies (s = v))$ (deducible from 1 by the lemma, with x, y, z substituted with t, s, v respectively)
4. $s = t$ (hypothesis)
5. $t = s$ (from 4 by symmetry)
6. $(t = v) \implies (s = v)$ (MP on 3, 5)
7. $t = v$ (hypothesis)
8. $s = v$ (MP on 6, 7)

□

Since E is indeed an equivalence relation, we can talk about the equivalence classes of T under the equivalence relation E , and about the quotient set T/E which is the set of all equivalence classes of T : recall that an *equivalence class* of T under E is a subset $[x] \subseteq T$ such that $\forall a, b \in [x], aEb$, and that the quotient set T/E is a set which consists of all equivalence classes of T . It can be shown that all equivalent classes of an equivalence relation E on a set are *disjoint* (suppose by contradiction that they are not, so there exists two elements a, b such that $a \in [x]$ and $b \in [x]$ as well as $b \in [y]$ where $[x] \neq [y]$ (otherwise they are both the same equivalence class), so $a \sim b$ but also $a \sim y$ for every $y \in [y]$, so by symmetry and transitivity $b \sim y$ for every $y \in [y]$, but then $y \in [x]$, but this holds for every $a \in [x]$ and for every $y \in [y]$, so $[x], [y]$ are identical, a contradiction).

DEFINITION 0.23 \langle Quotient set of closed terms (M^π) \rangle

Consider a vocabulary π with closed terms T and an equivalence relation E defined as in 0.22 closed term equivalence relation, then we denote the quotient set T/E by $M^\pi = T/E$.

Each equivalence class in M^π is represented by some element x in that class, called the *representative element* of the class. Usually, we denote the class by $[x]$ where x is the representative element (this is true in general and is unrelated to formal logic).

EXAMPLE 22.1 (equivalence relation and quotient set)

We step out of the field of formal logic to show an example for an equivalence relation on \mathbb{N} (including 0):

Fix some $n \in \mathbb{N}$. For every $x \in \mathbb{N}$, $\exists k \in \mathbb{N} \exists r \in \mathbb{N}$ such that rn and $x = kn + r$, where r is called the remainder of x with respect to n . It can be shown that this representation exists and is unique. For simplicity, define $f : \mathbb{N} \rightarrow \mathbb{N}$ to be the function that maps x to its r term in that representation (this is well-defined since the representation exists and is unique) We define a binary relation \sim on \mathbb{N} as follows: $\forall a, b \in \mathbb{N} (a \sim b \iff f(a) = f(b))$, i.e. $a \sim b$ if their remainders with respect to n are identical.

First, we verify this is an equivalence relation:

1. Reflexivity is immediate since $f(a) = f(a)$ for every a
2. Symmetry is immediate since equivalence is symmetric
3. Transitivity also immediately follows from the transitivity of equivalence

We call this equivalence relation $\text{mod } n$, and say that $a = b (\text{mod } n)$ if $a \sim b$ (where \sim is the relation described above).

Since it is an equivalence relation, we can consider the equivalence classes of $\text{mod } n$. For example, if $n=1$ then all numbers such that their remainder from n is 1 form an equivalence class which also contains the number 1 (since $n=1$ then clearly $1 = 0 \cdot n + 1$ so $f(a) = 1$), so we take 1 to be the representative of the class. In general, since r can take exactly $\min\{0, n-1\}$ values, we have exactly n disjoint equivalence classes (since for every natural number in $[0, n-1]$ we have a remainder equal to that value, so that value belongs to an equivalence class represented by that value, and since these are the only allowed values for r then all other numbers are categorized into these classes by the equivalence relation).

Thus, the quotient set of \mathbb{N} under $\text{mod } n$ is given by $\mathbb{N} / \sim = \{[0], [1], [2], \dots, [n-1]\}$, with the representative element of each equivalence class written inside the square braces.

Now we can present the model we will use to show the satisfiability of maximally connected sets:

DEFINITION 0.24 (the term model)

Given a FOL \mathcal{L} with vocabulary π and a set of sentences ϕ , with M^π as defined in 0.23 Quotient set of closed terms (M^π), then the term model \mathcal{M}^T is a structure of π where:

1. The domain of discourse is $D^{\mathcal{M}} = M^\pi$, i.e. the quotient set of T/E where E is the equivalence relation defined on clos terms in 0.22 closed term equivalence relation
2. Constant symbols c_i are interpreted as $[c_i] \in D^{\mathcal{M}}$, i.e. as their equivalence class. More concisely - $c_i^{\mathcal{M}} = [c_i]$.
3. Function symbols f_i are interpreted as $f_i^{\mathcal{M}}([t_1], \dots, [t_n]) = [f_i(t_1, \dots, t_n)]$ (recall that a function produces a term so if all terms involved in the invocation of the function are close terms, they are also categorized by E into equivalence classes).
4. Relation symbols $R_i(t_1, \dots, t_n)$ are defined using a relation $R_i^{\mathcal{M}}$ (previously we defined relations using operators whose co-domain is \mathbb{B} , while these definitions are equivalent it is easier in this context to interpret a relation symbol as a proper relation on members of $D^{\mathcal{M}}$) such that $([t_1], \dots, [t_n]) \in R_i^{\mathcal{M}} \iff \phi \vdash R_i(t_1, \dots, t_n)$

EXAMPLE 22.2 (term model usage)

Consider the sentence $f(b, c) = f(a, c)$ where f, h are function symbols and a, b, c are constant symbols, and suppose $a = b$ is deducible, then using the term model we interpret this sentence as follows:

1. $a^{\mathcal{M}} = [a], b^{\mathcal{M}} = [b], c^{\mathcal{M}} = [c]$, and since $a = b$ is deducible (by hypothesis) then $[a] = [b]$ by definition of E .
2. $(f(b, c))^{\mathcal{M}} = f^{\mathcal{M}}([b], [c])$ but since $[b] = [a]$ then $f^{\mathcal{M}}([b], [c]) = f^{\mathcal{M}}([a], [c]) = [f(a, c)]$
3. $(f(a, c))^{\mathcal{M}} = f^{\mathcal{M}}([a], [c]) = [f(a, c)]$
4. Since $(f(b, c))^{\mathcal{M}} = (f(a, c))^{\mathcal{M}}$, then the sentence $f(b, c) = f(a, c)$ is true in the term model.

In the example, we deduced $f(b, c) = f(a, c)$ from $a = b$ in our model. We will now show that this is also logically implied, i.e. $\{a = b\} \vdash (f(b, c) = f(a, c))$:

1. $(x = y) \implies (f(x, c) = f(y, c))$ (E2)
2. $(a = b) \implies (f(a, c) \implies f(b, c))$ (By 22.1 generalization and instantiation lemma , with a replacing x and b replacing y)
3. $a = b$ (hypothesis)
4. $f(a, c) = f(b, c)$ (MP on 2, 3)
5. $f(b, c) = f(a, c)$ (symmetry on 4, which we have shown to be deducible)

We will now generalize this result to show that \mathcal{M}^T is in general well-defined:

PROPOSITION 10

The term model is well defined, i.e.

1. $f_i^{\mathcal{M}}$ is well defined for every f_i in π ,
2. $R_i^{\mathcal{M}}$ is well defined for every R_i in π .

Proof. 1. To show that $f_i^{\mathcal{M}}$ is well defined we need to show that the choice of representatives of its arguments does not matter, i.e. if f_i is unary we wish to show that $\forall [x] \in D^{\mathcal{M}} (\forall t, s \in [x] ([f(t)] = [f(s)]))$. The proof is essentially the same as the one shown above for $\{a = b\} \vdash (f(b, c) = f(a, c))$, so we omit it for brevity.

To generalize the argument, consider f with arity n and consider the terms $f(c_1, c_2, \dots, c_n), f(d_1, d_2, \dots, d_n)$, such that $d_i E c_i$, i.e. both belong to the same equivalence class, so we can suppose $d_i = c_i$. Now, suppose both invocation of f agree on the first k terms, i.e. c_i and d_i are syntactically the same terms up to $i = k + 1$ so we can write $f(d_1, d_2, \dots, d_k, c_{k+1}, \dots, c_n)$ instead of $f(c_1, \dots, c_n)$, then

1. $(x = y) \implies (f(d_1, \dots, d_k, x, c_{k+2}, \dots, c_n) = f(d_1, \dots, d_k, y, c_{k+2}, \dots, c_n))$ (E2)
2. $(c_{k+1} = d_{k+1}) \implies (f(d_1, \dots, d_k, c_{k+1}, \dots, c_n) = f(d_1, \dots, d_k, d_{k+1}, c_{k+2}, \dots, c_n))$ (22.1 generalization and instantiation lemma on 1 with c_{k+1}, d_{k+1} substituting x, y respectively)
3. $f(d_1, \dots, d_k, c_{k+1}, \dots, c_n) = f(d_1, \dots, d_{k+1}, c_{k+2}, \dots, c_n)$ (MP on 1, 2)

By repeating this process for all n term pairs $((c_1, d_1), \dots, (c_n, d_n))$ we arrive at $f(\bar{c}) = f(\bar{d})$, so $f^{\mathcal{M}}$ is well-defined as expected.

2. The proof is similar to 1, except we use E3 (which states $(x = y) \implies (\varphi \implies \psi)$) and use MP twice to conclude that, given $\vdash R_i(c_1, \dots, c_n)$, it follows that $\vdash R_i(d_1, \dots, d_n)$, assuming d_i, c_i belong to the same equivalence class under E for every i .

□

We wish to show that the term model *structure* is indeed a *model* for a maximally consistent set ϕ^* , i.e. that $\phi^* \models \mathcal{M}^T \varphi$ if and only if $\phi^* \vdash \varphi$.

THEOREM 23

Let ϕ^* be a maximally consistent set and let φ be a quantifier free sentence, and let \mathcal{M}^T be the term model structure for φ^* , then:

$$\phi^* \models_{\mathcal{M}^T} \varphi \iff \phi^* \vdash \varphi$$

Proof. Consider two cases:

First case: φ is atomic, then φ is either

1. $s = t$ where s, t are terms
2. $R(t_1, \dots, t_n)$ where t_1, \dots, t_n are terms.

1. Suppose φ is of the form $s = t$. To show \implies , suppose \mathcal{M}^T satisfies φ , then that means sEt , but by construction of the term model this means that $\phi^* \vdash (s = t)$. To show \impliedby , suppose $\phi^* \vdash (s = t)$, so again sEt so it follows that $s = t$ is satisfied by \mathcal{M}^T for any assignment.

2. Suppose φ is of the form $R(t_1, \dots, t_n)$. Recall that by construction of the term model, $([t_1], \dots, [t_n]) \in R^{\mathcal{M}} \iff \phi^* \vdash R(t_1, \dots, t_n)$, which proves the theorem for this case.

Second case: φ is not atomic, then φ is of the form:

1. $\neg\psi$ where ψ is quantifier free
2. $\psi C\chi$ where ψ, χ are quantifier free and C is one of the binary connectives $\vee, \wedge, \implies, \iff$.

To prove the theorem in this case, we proceed with induction on the length of φ . The base case is already covered by our proof for the case where φ is atomic.

1. Suppose φ is of the form $\neg\psi$, then by the induction hypothesis the theorem holds for ψ . To show \implies , suppose $\phi^* \models_{\mathcal{M}^T} \neg\psi$, which is true only if $\phi^* \not\models_{\mathcal{M}^T} \psi$, so by the induction hypothesis it follows that $\phi^* \not\vdash \psi$, but since ϕ^* is maximally consistent this means $\phi^* \vdash \neg\psi$. To show \impliedby , suppose $\phi^* \vdash \neg\psi$, so $\phi^* \not\vdash \psi$ since it is maximally consistent, so by the induction hypothesis $\phi^* \not\models_{\mathcal{M}^T} \psi$, so by definition $\phi^* \models_{\mathcal{M}^T} \neg\psi$.

2. Suppose φ is of the form $\psi C\chi$ where C is a connective, then by the induction hypothesis the theorem holds for ψ, χ , so by considering the truth-table of $\psi C\chi$ for every C and using Theorem 22 rule T in FOL we can immediately prove the theorem for every connective C . For example, if C is \implies , i.e. φ is $\psi \implies \chi$, then we observe that by the truth table of the proposition, it is *false* (i.e. $\phi^* \models_{\mathcal{M}^T} \neg(\psi \implies \chi)$) if and only if ψ and $\neg\chi$ are both satisfied, now by the induction hypothesis we know that in this case, $\phi^* \vdash \psi$ and $\phi^* \vdash \neg\chi$, so $\phi^* \vdash \neg(\psi \implies \chi)$ by rule T, so by contraposition the theorem holds for $C = \implies$.

□

Note that the restriction on φ to be quantifier free is required, as shown in the following example:

EXAMPLE 23.1 ⟨term model for group theory⟩

Let π be a vocabulary for group theory $\pi = \{e, c, \cdot\}$ where e is the identity element, c is a constant and \cdot be the multiplication operation, and let Λ be the set of group axioms, which are:

G1. (identity) $\exists x(\forall a(x \cdot a = a))$

G2. (associativity) $\forall x \forall y \forall z(x \cdot (y \cdot z) = (x \cdot y) \cdot z)$

G3. (inverse) $\forall a \exists x(x \cdot a = e)$

Consider the term model \mathcal{M}^T : the terms in π are $e, c, c \cdot c, (c \cdot c) \cdot c, c \cdot (c \cdot c), \dots$. By associativity, the equivalence relation E on the terms forms the following quotient set $D^{\mathcal{M}} = \{[e], [c], [c \cdot c], [c \cdot (c \cdot c)], \dots\}$. As a shorthand, we write c^n to denote multiplication of c with itself n times (by associativity the order is irrelevant), so we can write $D^{\mathcal{M}} = \{[e], [c], [c^2], [c^3], \dots\}$.

If \mathcal{M}^T is a model for Λ , then we should be able to satisfy all axioms of Λ , but consider axiom G3 (existence of an inverse element): given $a \mapsto [c]$ (or any other element that is not $[e]$), we cannot find an inverse in $D^{\mathcal{M}}$, so we cannot satisfy G3.

This is not a contradiction of Theorem 23 since G3 is not quantifier free, but this example shows that \mathcal{M}^T is not a model.

This means that \mathcal{M}^T is not a model for a maximally consistent set ϕ^* with vocabulary π , unless we make some assumptions on π .

DEFINITION 0.25 ⟨term witness⟩

Given a set of formulas ϕ and an existential statement ψ of the form $\exists x \varphi(x)$ (recall that $\varphi(x)$ is a notation which means that x is the only free variable of φ) such that $\phi \vdash \forall x \psi(x)$, then a term witness for ψ is a closed term t such that $\phi \vdash \psi_t^x$.

EXAMPLE 23.2 ⟨term witness⟩

If $\phi \vdash \exists x(x = c)$, and c is a constant symbol, then c is a closed term, so if $\phi \vdash (c = c)$ (which we have proven to always be deducible earlier) means that c is a term witness of $\forall x(x = c)$. Consider group theory with vocabulary $\pi = \{e, c, \cdot\}$: one of the axioms of group theory states $\forall a \exists x(x \cdot a = e)$, then by universal instantiation we can derive $\exists x(x \cdot c = e)$ (by substituting c for a). As we have seen in 23.1 term model for group theory we cannot model this axiom using the term model with this vocabulary since there is no closed term t such that $t \cdot c = e$, however if we extend π to include an inverse function, i.e. $\pi^* = \pi \cup \{()^{-1}\}$, then for every closed term t we can take its inverse t^{-1} which by definition is still a closed term, so then $c^{-1} \cdot c = e$ is true with c^{-1} being the term witness for that existential statement.

DEFINITION 0.26 (theory with term witnesses)

We say that a set of WFFs (i.e. a theory) ϕ contains term witnesses if for every existential statement $\exists x\varphi(x)$ such that $\phi \vdash \exists x\varphi(x)$, there exists a closed term $t \in T$ (where T is the set of closed terms in π the vocabulary of ϕ) such that $\phi \vdash \varphi(t)$.

It is worth noting that while *maximally consistent* is a restriction on the formulas and not on the vocabulary, *contains term witnesses* is a restriction on *both* (recall, again, the example provided for group theory with and without the inverse function symbol).

THEOREM 24 (model existence theorem)

Let ϕ^* be a maximally consistent set with term witnesses, and let \mathcal{M}^T be the term model on ϕ^* and its vocabulary π^* , then for every WFF φ ,

$$\phi^* \models_{\mathcal{M}^T} \varphi \iff \phi^* \vdash \varphi$$

Proof. It is helpful to start by contrasting this theorem with Theorem 23, which states the same result for a maximally consistent ϕ^* that does not necessarily contain term witnesses for every *qualifier free sentence*. This theorem requires ϕ^* to contain term witnesses and then states that \mathcal{M}^T is a *model* for ϕ^* . We will expand the proof for that theorem to handle any WFF:

first case: φ is atomic: we need to account for the cases where φ is atomic but the terms are not closed, i.e. they may consist of variable symbols. Let v be a variable symbol, and consider the statement $\exists x(x = v)$. This statement is a tautology, so by Theorem 22 rule T in FOL it is deducible from ϕ^* , but since ϕ^* contains term witnesses then $\exists t \in T$ such that t is a closed term and $t = v$, so v belongs to the equivalence class $[t]$, and due to transitivity the choice of representative t is irrelevant, so this operation is well-defined. Now that we can classify variable symbols as well, the proof for the case where φ is atomic in Theorem 23 can trivially be extended to cover the case where φ is an atomic *formula* (and not just an atomic *sentence*).

second case: The only case we need to cover is the case where φ is of the form $\exists x\psi$, since $\neg\exists x\neg\psi \equiv \forall x\psi$ (we have shown that $\exists x\psi = \neg\forall x\neg\psi$) earlier, so the contrapositive - which is this form - is also true) so once we show the theorem holds for $\exists x\psi$, and since we have already shown it holds for $\neg\psi$, the proof that it holds for $\forall x\psi$ follows immediately by the logical equivalence. So, suppose φ is of the form $\exists x\psi$,

To show \Leftarrow , suppose $\phi^* \vdash \exists x\psi$, so there exists a term witness t such that $\phi^* \vdash \psi_t^x$, and we can apply the induction hypothesis on ψ_t^x so it follows that $\phi^* \models_{\mathcal{M}^T} \psi_t^x$, so it follows that $\exists t \in D^{\mathcal{M}}$ such that $\phi^* \models_{\mathcal{M}^T} \psi[s(x \mapsto t)]$ for every assignment s , so by definition $\phi^* \models_{\mathcal{M}^T} \exists x\psi$.

To show \implies , suppose $\phi^* \models_{\mathcal{M}^T} \forall x\psi$, so again there exists some closed term t such that $\phi^* \models_{\mathcal{M}} \psi_t^x$, and by the induction hypothesis $\phi^* \vdash \psi_t^x$. We will now use the notation $\psi(x)$ and $\psi(t)$ instead for simplicity. We wish to show that if $\phi^* \vdash \psi(t)$ then $\phi^* \vdash \exists x\psi(x)$. To do this, we will derive the equivalent statement, i.e. $\neg\forall x\neg\psi(x)$,

1. $\forall x\neg\psi(x) \implies \neg\psi(t)$ (A4)
2. $(\forall x\neg\psi(x) \implies \neg\psi(t)) \implies (\neg\neg\psi(t) \implies (\neg\forall x\neg\psi(x)))$ (A3)
3. $\neg\neg\psi(t) \implies (\neg\forall x\neg\psi(x))$ (MP on 1, 2)
4. $(\neg\neg\psi(t) \implies (\neg\forall x\neg\psi(x))) \implies (\psi(t) \implies (\neg\forall x\neg\psi(x)))$ (tautology of propositional logic, so deducible by Theorem 22 rule T in FOL)
5. $\psi(t) \implies (\neg\forall x\neg\psi(x))$ (MP on 3, 4)
6. $\psi(t)$ (hypothesis)
7. $\neg\forall x\neg\psi(x)$ (MP on 5, 6)
8. $\exists x\psi(x)$ (logically equivalent to 7)

This completes the proof and shows that if ϕ^* is a maximally consistent set with term witnesses, then the term model \mathcal{M}^T is a *model* for ϕ^* .

□

We have reached an important landmark - we have shown that for every ϕ^* this is maximally consistent and has term witnesses, it is satisfiable. We call this result a model existence theorem. Using this result, we can employ our proof from propositional logic to show that ϕ^* is (semantically) complete (given the axioms and inference rules of the proof system). Completeness then implies compactness by Theorem 15 compactness theorem for P_2 (the same proof applies to FOL theories). The only missing piece in our completeness proof is to show that every consistent set ϕ can be extended to a maximally consistent set with term witnesses ϕ^* .

Theory extensibility theorem We will show that given a consistent set ϕ , it can be extended to a maximally consistent set with term witnesses ϕ^* .

LEMMA 24.1

Let ϕ be a consistent set and let c be a constant symbol that does not appear in ϕ and in some sentence φ , then the set $\phi \cup \{\exists x\varphi \implies \varphi_c^x\}$ is consistent.

Intuitively, the lemma states that, given a “fresh” constant symbol c , i.e. a symbol which is not used in ϕ and this not used in a model of ϕ to satisfy the formulas of ϕ , then we can interpret the new symbol c as a *term witness* for an existential sentence $\exists x\varphi$, and we expect this operation to maintain the consistency of ϕ because we don’t expect this extension of ϕ to introduce inconsistencies.

Proof. Suppose by contradiction $\phi' = \phi \cup \{\exists x\varphi \implies \varphi_t^x\}$ is inconsistent, then by Theorem 12 proof by contradiction (reductio ad absurdum) in P_2 it follows that $\phi \vdash \neg(\exists x\varphi \implies \varphi_t^x)$, which is tautologically equivalent to $(\exists x\varphi) \wedge \neg\varphi_t^x$, so by Theorem 22 rule T in FOL $\phi \vdash \exists x\varphi \wedge (\neg\varphi_t^x)$, but $\phi \vdash A \wedge B$ if and only if $\phi \vdash A$ and $\phi \vdash B$ by previous results, so $\phi \vdash \exists x\varphi$ and $\phi \vdash \neg\varphi_t^x$, so:

1. $\exists x\varphi$ (see above)
2. $\neg\forall x\neg\varphi$ (logically equivalent to 1)
3. $\neg\varphi_t^x$ (see above)
4. $\forall x\neg\varphi$ (GEN on 3)

Denote $\psi := \forall x\neg\varphi$, then $\phi \vdash \psi$ (this is step 4 in the proof), but also $\phi \vdash \neg\psi$ (this is step 2 in the proof), contradicting the consistency of ϕ , which completes our proof by contradiction. □

Using this lemma, it is trivial to show the following

LEMMA 24.2

Let ϕ be a consistent set, then ϕ can be extended to a new consistent set with term witnesses by:

1. *Adding a new constant c_φ to the vocabulary for every WFF $\varphi \in \phi$, which will serve as the term witnesses of each φ and extending the vocabulary π to a vocabulary $\pi^* = \pi \cup c_\phi$ where c_ϕ is the set of all such constants.*
2. *Defining the set $\Psi = \{\exists x\varphi \implies \varphi_{c_\varphi}^x \mid \varphi \in \phi\}$*
3. *Taking $\phi^* = \phi \cup \Psi$*

Then ϕ^ is consistent, contains term witnesses, and contains ϕ .*

Proof. We only provide a sketch of the proof. The idea is to show a proof by induction using some enumeration on ϕ , and then adding each $\exists x\varphi \implies \varphi_{c_\varphi}^x$ statement sequentially. By 24.1, adding this statement to a consistent set ϕ maintains consistency □

THEOREM 25 (theory extensibility theorem)

Let ϕ be a consistent set of WFFs in FOL, then ϕ and its vocabulary π can be extended to ϕ^ and π^* such that ϕ^* is maximally consistent and contains term witnesses.*

Proof. To make ϕ maximally consistent over the vocabulary π , we can repeat the process described in 12.1 maximization existence. Then we take the maximally consistent set over π and extend it (and π)

to a maximally consistent set with term witnesses as described in 24.2, which produces the desired result. Note that if the set of WFFs or the set of constants is uncountable, we use *transfinite induction* instead of structural induction to prove these lemmas \square

Finally, we can state Godel's semantical completeness theorem for FOL, which is a natural extension of the semantical completeness theorem for propositional logic:

THEOREM 26 (Godel's completeness theorem)

Let ϕ be a set of WFFs in FOL, then:

1. If $\phi \models \varphi$ then $\phi \vdash \varphi$
2. If ϕ is consistent, then it is satisfiable.

Proof. Now that we have shown how a consistent set ϕ can be extended to a maximally consistent set with term witnesses such that it has a model, the proof is immediate (we have already proved this in our discussion of propositional logic) \square

What's Next?

In this section, we provide a brief overview of more advanced topic in mathematical logic.

Higher-Order Logic

So far, we have discussed:

1. Propositional logic, which deals with *propositional variables*
2. First-order, or predicate, logic, which deals with variables of some *domain* that satisfy *predicates*, and *quantifies* these variables

Second-order logic is a natural extension of first-order logic, where we allow quantifying over *predicates* as well. For example, the following formula is well-defined in second-order logic (with equality):

$$\forall x \forall R (R(x) \iff (x = v))$$

Note that we not only quantify over variables as we did in FOL, but we quantify over the relations R as well. Quantifying over relations means we can quantify over sets, for example the following statement

$$\forall R (\exists x R(x))$$

States that every relation has at least one element satisfying that relation, which if we interpret the formula in the setting of set theory, means that every set has at least one element.

Second-order logic lets us express reachability - suppose we have two persons x, y such that x is a parent of y , in first-order logic we express this as $\text{IsParent}(x, y)$. Suppose we wish to express that x is reachable from y , which in graph theory means that there exists a path from x to y , in the graph. We can translate this idea to formal logic with the interpretation from graph theory as follows: suppose $\text{IsParent}(x, y)$ is translated as “is there an edge connecting x, y ”, then a path is simply a set that is *closed* under the IsParent relation. This idea can be expressed as follows

$$\forall P(\forall x \forall y((P(y) \wedge \text{IsParent}(x, y)) \implies P(x)))$$

To express the idea that a is reachable from b , we expand the previous formula to state that every set that contains a and closed under IsParent also contains b :

$$\forall P((P(a) \wedge \forall x \forall y(P(y) \wedge \text{IsParent}(x, y)) \implies P(x))) \implies P(b)$$

Due to second-order logic letting us quantify over sets, one common criticism about second-order logic is that it is set theory in sheep’s clothing, especially since one can find correspondences between results in *ZFC* set theory to results in second-order logic.

If second-order logic extends first-order logic by letting us quantify over *sets* (of elements), then a *third-order logic* extends second-order logic by letting us quantify over *sets of sets*. This idea can be extended arbitrarily. We call logic systems of third-order and higher *higher-order logic*.

These ideas will not be formalized here.

Godel’s Incompleteness Theorem

Godel’s incompleteness theorem is a famous result in formal logic which applies to some first-order logic proof systems that satisfy some properties. While we will not discuss the proof itself or its implications formally, it would be a shame to not mention it at all. In this section, we will state provide a semi-formal statement of the theorem, which would hopefully clarify common confusion and misinterpretations of the theorem.

DEFINITION 0.1 *⟨arithmetic system⟩*

A formal FOL system ϕ is called arithmetic if it can carry out the arithmetics of the natural numbers, i.e. if it contains as tautologies sentences equivalent to the Peano axioms, which are an axiomatization of natural number arithmetics using a vocabulary which includes additive and multiplicative identities, addition and multiplication symbols, and a successor function symbol.

While this half-definition is not without issues, the key takeaway is that if a language can perform arithmetics on elements in a way one would expect to be able to perform arithmetics on the natural numbers, then the system is said to be arithmetic.

Godel's incompleteness theorem actually consists of *two* theorems, and the kind of incompleteness Godel speaks of is not one of those we have already presented (functional and semantical completeness), which is to be expected since we have actually shown that *all* consistent FOL systems are *semantically* complete (in Godel's very own *completeness* theorem). Godel's incompleteness theorems speak of *syntactical* completeness:

DEFINITION 0.2 *⟨syntactical completeness⟩*

A formal system ϕ is syntactically complete if for every sentence φ in the language of the system, either $\phi \vdash \varphi$ or $\phi \vdash \neg\varphi$.

COROLLARY 26.1

If a formal system ϕ is consistent and not syntactically complete, then there exists a sentence φ such that $\phi \not\models \varphi$ and also $\phi \not\models \neg\varphi$, i.e. φ is true in some models of ϕ and false in others.

Proof. If ϕ is not syntactically complete then neither $\phi \vdash \varphi$ nor $\phi \vdash \neg\varphi$, and since ϕ is consistent then by Godel's completeness theorem it is *semantically* complete, which means that neither $\phi \models \varphi$ nor $\phi \models \neg\varphi$, which is possible if and only if in some models of ϕ , φ is satisfied and in others, it is not \square

THEOREM 27 *⟨Godel's first incompleteness theorem⟩*

If a formal FOL system ϕ is arithmetic and consistent and the proof system is effective, then there exists a sentence G_ϕ expressed in the language \mathcal{L} such that $\phi \not\vdash G_\phi$ and $\phi \not\vdash \neg G_\phi$, i.e. ϕ is syntactically incomplete.

One immediate result of this theorem is that a system which models a certain amount of elementary arithmetics *cannot* model all of mathematics. This idea - to formalize a grand, unified, consistent and

syntactically complete theory of mathematics - was the driving force behind Whitehead and Russell's Principia Mathematica (commonly abbreviated as PM). However in doing so, Whitehead and Russell introduced arithmetics to their model, which by Theorem 27 Godel's first incompleteness theorem inevitably makes it *syntactically incomplete*.

In his original paper, Godel proved the syntactical incompleteness of PM.

Godel's first incompleteness theorem, along with 26.1, implies that for every system ϕ there exists models of ϕ which are not equivalent.

THEOREM 28 (Godel's second incompleteness theorem)

If a formal FOL system ϕ contains basic arithmetics, is consistent and effective, then it cannot prove its own consistency, i.e. if $\text{Cons}(\phi)$ is a statement that states ϕ is consistent, then neither $\phi \vdash \text{Cons}(\phi)$ nor $\phi \vdash \neg \text{Cons}(\phi)$.

Among the most common misinterpretations and misstatements of Godel's incompleteness theorems, are:

1. *There are certain truths in math that cannot be proven* (this is a common misinterpretation of the first theorem)
2. *I can take an established result in some theory and choose to ignore it without being inconsistent* (this is a misinterpretation of 26.1)
3. *Math itself is inconsistent/wrong* (this is a somewhat common misinterpretation of the second theorem)
4. *Outlandish claims attempting to draw some universal/natural truth from those theorems*

Hopefully this section provides enough justification as to why these statements are flat-out wrong.