forallχ (MIT 24.241 Fall 2024)

An Introduction to Formal Logic
Compiled on November 26, 2024

Benjamin Brast-McKie
Massachusetts Institute of Technology

Rewritten for MIT from Ichikawa and Mangus' version of forallχ which included sections from the Calgary Remix by Magnus, Button, Loftis, Trueman, Thomas-Bolduc, and Zach.

MIT 24.241 Fall 2024 edition

Typesetting was carried out in LaTeX2ε. The style for natural deduction proofs is based on fitch.sty (v0.4) by Peter Selinger. Tree typesetting from prooftrees (v0.6) by Clea F. Rees.

This copy of `forall`χ is current as of November 26, 2024.

Contents

Preface to the Fall 2024 MIT Edition

I inherited a version of Ichikawa and Mangus' version of `forall`χ from Josh Hunt who worked to bring the text closer to *The Logic Book* by Bergmann, Moor, and Nelson for teaching Logic I at MIT. Since then, I have dropped certain chapters, substantially rewritten the chapters that have been preserved, and included new chapters on soundness and completeness for a Fitch-style natural deduction system for first-order logic. At this point, few traces of the version I started with remain. Rather than attempting to record all of the changes that I have made, I will state the main changes that I still hope to make:

1. Include a primer on set theory, relations, and functions in an *Appendix*.
2. Include a glossary of terms and symbols.
3. Include cheatsheets for the proof rules for PL and $\text{FOL}^=$ in an *Appendix*.
4. Include an index of lemmas, theorems, and corollaries.
5. Include practice problems for each week along with solutions.

Whereas introductory logic courses often draw students who are new to writing mathematical proofs, the undergraduate students taking logic at MIT have strong formal backgrounds. This textbook was rewritten on their behalf and aims to provide a philosophically and formally rigorous introduction to formal logic through the soundness and completeness of first-order logic with identity. I have taken pains to provide rigorous proofs throughout, including commentary about how the proofs work in addition to discussing their motivations.

Although this text is written with greater formal rigor than is common in most introductory textbooks in logic, I have aimed to preserve the friendly and accessible character of `forall`χ in addition to preserving its title. Additionally, I have included discussions of the philosophical foundations of logic which are often omitted in introductory books and simply alluded to in more advanced treatments without being discussed or defended.

Teaching

The terms at MIT span fifteen weeks, two of which are devoted to soundness and completeness for propositional logic (PL) and another two weeks to extend these results to first-order logic with identity ($\mathrm{FOL}^{=}$). In order to introduce the material at a slower rate or over the course of a shorter term, the metalogical portions of the text can easily be omitted without compromising the integrity of the text.

In addition to the textbook, I have included source files for the lecture notes (compiled here as a PDF) as well as the syllabus for the course at MIT on GitHub. Feel free to contact me if you would like access to the exams, written problem sets, and assignments on Carnap.

Collaboration

Although you are welcome to fork the repository on GitHub adapting these resources as you please, I would also be happy to accept pull requests and would greatly appreciate being notified of any errors. If you have any suggestion or questions, I would encourage you to open an issue above. At some point I hope to include documentation describing how to adapt these resources for those new to LaTeX and GitHub.

If you are interested in using the text editor that I used to streamline work on this project, you can check out the NeoVim config that I maintain, as well as the VSCodium config that I started for those who are looking for something a little more user friendly.

Benjamin Brast-McKie
MIT, August 2024
brastmck@mit.edu

Chapter 0

What is Logic?

Logic is the study of *formal reasoning*, that is, what follows from what in virtue of logical form. Although the reasoning that we will be concerned with will include mathematical symbols, the choice of symbols is arbitrary, and so this is not the sense of formality at issue. After all, many fields engage in reasoning with mathematical looking symbols but we will not be concerned to describe the reasoning that goes on in those fields.

There are of course many other ways to reason besides the manner we will be concerned with in this book. In order to characterize the subject-matter of logic, this chapter will contrast formal reasoning with a number of other common types of reasoning. This will help to provide a theoretical target which subsequent chapters describe first *semantically* and then *syntactically* before proving that these two accounts describe the same thing twice over.

Reasoning moves the reasoner from some number of considerations to a further consideration. In particular, we will be concerned with DECLARATIVE SENTENCES in English, that is, English sentences that are either true or false, and not both at once.[1] A DEDUCTIVE ARGUMENT in English is a nonempty sequence of declarative sentences where a single sentence is designated the CONCLUSION and all other sentences (if any) are referred to as the PREMISES. For instance, consider the following argument where the conclusion follows the horizontal line:

A1. Kat is sitting down.
A2. If Kat is sitting down, then Kat is not standing.
A3. Kat is not standing.

The horizontal line is pronounced 'therefore'. The premises of an argument aim to provide *reasons for taking the conclusion to be true* even if not all of the premises are true, known to be true, likely, or even possible. For instance, even if we don't know whether A1 and A2 are true, we may admit that if A1 and A2 are both true, then the conclusion A3 is also true.

[1] Whereas italics will be used for emphasis, key vocabulary will be capitalized throughout.

Since not all arguments in English take such explicit forms as the argument above, part of what we will be concerned with is regimenting arguments in English by translating them into an artificial formal language. Although there is often more than one way to regiment an argument stated in English, once an argument has been regimented in a formal language, no further ambiguity remains as to how to evaluate that formalized argument. An important first step in this process is to identify the premises and conclusion, writing them as a numbered list where the conclusion occurs on the last line with a line separating it from the premises.

In the course of this book, we will introduce the syntax for the formal languages $\mathcal{L}^{\text{PL}}$ and $\mathcal{L}^{\text{FOL}}$. For the time being, this chapter will restrict consideration to arguments in English written in premise-conclusion form. In the case of the argument above, we may observe that believing the premises provides a strong reason to believe the conclusion independent of whether we happen to believe the premises or not. Of course, not all arguments succeed in providing such strong support for their conclusions. For instance, consider the following argument:

B1. Siya is out sailing.
B2. Nicky is hard at work.
B3. The house is empty.

Even though there might be ways to modify the argument in order to make it more compelling, as stated, the argument above is just a series of (logically) unrelated sentences. Accordingly, one might be tempted to protest that this is no argument at all, but rather a random list of sentences. Although this book will not take a stand on how the word 'argument' ought to be used in general, we won't require arguments to be compelling in order to count as arguments.

Whereas the first argument we provided was very compelling and the second argument was not compelling at all, many arguments fall somewhere between these two. Here is an example:

C1. Edinburgh is North of Oxord.
C2. Oxford is North of London.
C3. Edinburgh is Nort of London.

Although this argument might seem to be much more compelling than the second argument, its conclusion does not follows from its premises by virtue of logic alone. Put otherwise, the conclusion in this final argument is not a purely logical consequence of its premises, where something similar may be said for the second argument given above. Whereas later chapters will carefully define what it is for a sentence to be a logical consequence of a set of sentences in formal languages, the remainder of the present chapter will focus on characterizing an informal analogue for arguments in English. These considerations will provide an intuitive conception of what it is for a conclusion to follow as a matter of logic from some number of premises, motivating the study of the formal languages presented in the next chapter.

In addition to providing a means by which to resolve the ambiguities of natural language, regimenting English arguments in formal languages may be used to provide an account of their logical form. As we will see, much will turn on the expressive power of the formal language since the conclusion of an argument in English may be a logical consequence of its premises when regimented in one formal language and not in another. Whereas this book will introduce two formal languages, this is not a stopping point but rather a place to begin.

0.1 Arguments

A crucial part of analyzing an argument is identifying its conclusion. Every argument has a conclusion— the conclusion is the claim the argument aims to establish. Premises are starting-points, used to lend support to the conclusion. In English, the conclusion is often signified by words like 'so' or 'therefore'. Premises might be marked by words like 'because'. These words can provide some clue as to just what the argument is supposed to be.

Premise indicators: 'since', 'because', 'given that'

Conclusion indicators: 'therefore', 'hence', 'thus', 'then', 'so'

In a natural language like English, arguments *sometimes* begin with their premises and end with their conclusions, but not always. Since arguments in which the conclusion does not occur at the end may be rearranged, we may restrict attention to arguments in which the conclusion occurs on the last line without loss of generality. For instance, in a single sentence, one might argue that, "People often get wet in Cambridge because it often rains in Cambridge and people get wet when it rains." We may then rearranging this argument as follows:

D1. People get wet when it rains.
D2. It often rains in Cambridge.
D3. People often get wet in Cambridge.

One of our central aims is to provide rigorous formal method for evaluating whether an argument's conclusion is a logical consequence of its premises by checking if the conclusion is true "whenever" the premises are true. We will have much more to say about exactly what "whenever" means here, but for now we will focus on describing the theoretical role that deductive arguments strive to play. For contrast, *inductive arguments* aim to increase our degree of confidence in a conclusion by appealing to some number of instances of a general inference which provide evidence in support of that general inference. Additionally, *abductive arguments* appeal to the best explanation for some number of claims in order to provide support for a further claim. Whereas deductive arguments do not admit *any* counterexamples, even the strongest inductive and abductive arguments remain open to counterexamples.

For better or for worse, we will have nothing more to say about inductive and abductive arguments in this course.[2] This is despite the fact that much of science and ordinary life operates using inductive and abductive rather than deductive reasoning. The good news is that our systems will be versatile and have some elegant formal properties, making it a good candidate for a wide range of applications. In particular, deductive logic is of vital importance for mathematics and computer science, significantly reshaping the world that we live in by making the rise of the information age possible.

To exposition, we will use 'argument' to mean 'deductive argument' throughout what follows.

0.2 Sentences and Propositions

The premises and conclusions of an argument in English are grammatical sentences, but not all grammatical English sentences are suitable for figuring in arguments. For example, questions count as grammatical English sentences, but logical arguments never have questions as premises or conclusions. As mentioned above, we are specifically interested in declarative sentences that can be true or false, and never both true and false at once. In order to set them aside, here are some types of the sentences we will not be concerned with:

Questions 'Are you sleepy yet?' is a perfectly grammatical interrogative sentence. Whether you are sleepy or not, the question itself is neither true nor false. Suppose you answer the question: 'I am not sleepy.' This answer is either true or false, and so is a declarative sentence. Generally, questions will not be declarative sentences, but answers to questions often will. For instance, 'What is this course about?' is not a declarative sentence, but 'No one knows what this course is about' is a declarative sentence.

Imperatives Commands such as 'Wake up!', 'Sit up straight!', and so on are grammatical imperative sentences. Although it might be good for you to sit up straight or it might not, the command is neither true nor false. Note, however, that commands are not always phrased with imperatives. 'You will respect my authority' *is* either true or false— either you will or you will not— and so it counts as a declarative sentence.

Exclamations Expressions like 'Ouch!' or 'Boo, Yankees!' are sometimes described as exclamatory sentences, but they are neither true nor false. We will treat 'Ouch, I hurt my toe!' as meaning the same thing as 'I hurt my toe.' The 'Ouch' does not add anything that could be true or false, and so will simply be dropped if it occurs at all.

In contrast to the types of sentences above, declarative sentences describe *ways for some particular things to be* which we will refer to as PROPOSITIONS. Put otherwise, declarative sentences *express* propositions, where the same proposition may be expressed in many different ways. For instance, the English sentence 'snow is white' expresses the same proposition as the German sentence 'Schnee ist weiß'. Each says the world is a certain way.

[2] We *will* have occasion to employ an entirely different style of reasoning called 'mathematical induction'.

For our purposes, the defining feature of a proposition is that it either *obtains* or it does not. For instance, suppose that Kat is sitting down. The proposition that Kat is sitting down is a thing (a person)— namely Kat— being a certain way, i.e., sitting down. Given our supposition, the proposition that Kat is sitting down obtains. Similarly, we may suppose Tsovinar is not singing, and so the proposition that Tsovinar is signing does not obtain. Despite failing to obtain, the proposition that Tsovinar is signing is a way for things to be all the same— i.e., Tsovinar singing— it is just that things are not in fact that way.

Whereas a proposition is a way for some things to be which may either obtain or fail to obtain, a declarative sentence is a grammatical string of symbols which, given an interpretation of those symbols, is either true or false. For instance, supposing that Kat is sitting down— i.e., the proposition that Kat is sitting down obtains— the sentence 'Kat is sitting down' is true. More generally, a declarative sentences in English is true on an interpretation if it expresses a proposition that obtains, and false otherwise. The TRUTH-VALUE of a declarative sentence on an interpretation is either the value 'True' or 'False' depending on whether the proposition that sentence expresses on that interpretation happens to obtain.

Given that we will only be concerned with whether a proposition obtains or not, we may make the simplifying assumption that there are only two proposition: TRUTH which obtains and FALSITY which does not. Simpler still, we may ignore talk of propositions altogether by focusing on the corresponding truth-values of sentences on an interpretation, using '1' in place of 'True' and '0' in place of 'False'. More specifically, we will INTERPRET the sentences of our various languages by assigning them truth-values, ignoring any further differences that this may neglect. The present investigation belongs to CLASSICAL LOGIC insofar as we will require interpretations to assign sentences to either true or false and never both at once.

There are many sentences where it is unclear or controversial whether they have a truth-value. Think of sentences such as 'Almonds are yummy' or 'War is never justified'. In an argument, we can assign a truth-value to such sentences, even if one might be skeptical that they have a truth-value independent of any subject. We will handle these sentences just like sentences which have objective truth-values such as the sentence, 'Tsovinar is signing'. Indeed, often what is at stake in our more interesting arguments is whether various normative or evaluative claims are true or false. So clearly, we have *some* way of reasoning about such sentences using truth-values where this form of reasoning is that we will aim to describe here. What else there is to say about normative and evaluative sentences is beyond the scope of this course.

It is also important to keep clear the distinction between truth and knowledge. Although a declarative sentence is the kind of thing that can be true or false, this does not mean that we will always be in a position to know whether it is true or false. For example, given a definition of what counts as a gold atom, the sentence, 'There are an even number of gold atoms on Earth right now' may be taken to have a truth-value even though it would be virtually impossible to determine. Similarly, there are many controversies where people disagree about what is true or false, and it very hard to settle the debate. However, for the sake of an argument, we may treat controversial premises as true in order to see what follows logically as a result without committing ourselves to these assumptions.

0.3 Logical Consequence

For our purposes, there are two important ways that arguments can go wrong. To begin with, suppose that the following argument were presented in a court of law:

E1. The victim was shot by a bullet from the gun that was found at the defendant's house.
E2. The fingerprints on the gun were shown to match the defendant.
E3. The gun was registered in the defendant's name.
E4. The defendant had recently been fired by the victim.
E5. The defendant shot the victim.

An argument is only compelling if its premises are true. If the premises above can be shown to be false, the defendant may well be innocent. More generally, we should not feel at all persuaded to believe a conclusion on the basis of an argument with false premises. This is the first way that an argument can go wrong: not all of the premises are true.

Assuming that the premises E1 – E4 are true, the conclusion E5 would seem to follow. What we mean by 'follow' in this instance is that the truth of the premises makes the truth of the conclusion extremely likely, and perhaps so likely that it is beyond a reasonable doubt, compelling a jury to find the defendant guilty. However, what may be good enough for a court of law is not good enough for logic. This is not to disparage the criminal justice system but to observe that the truth of the conclusion in the argument above does not follow *logically* from the truth of its premises. However likely the conclusion may be given the truth of its premises, it is still *possible* for the premises to be true and the conclusion false. For instance, consider a possibility in which it was the defendant's partner who shot the victim. Even though all of the premises are true, the conclusion is false in such a possibility. We may not know if such a possibility took place and it may seem rather unlikely, but it doesn't matter. So long as it is possible for the premises to be true and the conclusion to be false we have reason to deny that the conclusion follows logically from its premises.

Is this what logical validity is about: ruling out *possibilities* in which the premises are true and the conclusion is false where possibilities are ways for things to be? Although logical validity is sometimes glossed this way, the answer is 'No!'. Let's take a look at some arguments that rule out possibilities in which the premises are true and the conclusion false:

F1. The atom is gold.
F2. The atom has 79 protons.

G1. Suela is a fox (the animal).
G2. Suela is female.
G3. Suela is a vixen.

In both of the arguments above, every possibility in which premises are true is one in which the conclusion is true, and so the conclusion follows as a matter of NECESSITY from the premises. Put otherwise, the premises STRICTLY IMPLY the conclusion in each of the arguments above. Certainly these are stronger arguments than what we might find in a court of law. Are there arguments that are even more powerful than this? The answer is 'Yes!', and this brings us to the subject-matter of this course. Consider the following argument:

H1. Socrates is human.
H2. Every human is mortal.
H3. Socrates is mortal.

Here too the premises strictly imply the conclusion since there is no possibility in which the premises are true and the conclusion is false. However, unlike the previous arguments, we do not need to know what 'Socrates', 'human', or 'mortal' each mean. In order to get a sense of this, let's consider one more argument:

I1. Gyre is a mome rath.
I2. All mome raths are slithy.
I3. Gyre is slithy.

Even without knowing who Gyre is, anything about mome raths, or what it is to be slithy, we may say with equal certainty that it is not possible for the premises to be true and the conclusion false. In fact we can say more than this: there is no *interpretation* of the premises and conclusion where the former are true and the latter is false.

We will provide a precise definition of what an interpretation is when we set up semantic theories for the languages that we will study throughout this course (Chapters 2, 8, and 9). For the time being, it will help to get some sense of an informal analogue with which we are already familiar. To do so, let's return to the gold argument from before.

Surly any possibility in which the atom is gold is also one in which the atom has 79 protons. After all, having 79 protons is part of what it is to be gold, and so something couldn't be gold without having 79 protons. Who could argue with that?

Although no one should balk at the gold argument given the normal interpretation of its sentences— what is often called the *intended interpretation*— the same cannot be said if we entertain unintended interpretations. For instance, suppose we were to take 'gold' to mean what 'carbon' means in the intended interpretation, that is: carbon. The proposition expressed on this unintended interpretation could have been expressed by an intended interpretation of the sentence 'The gold atom is carbon'. Since carbon only has 6 protons and not 79, the conclusion is false when the premise is true on this unintended interpretation. But why should we care about unintended interpretations? Shouldn't we restrict attention to just the intended interpretations of our sentences that we are accustomed to using?

One reason for considering all interpretations and not just an intended interpretation that we seem to most of the time is that it is very hard to say what is true on an intended interpretation, nor is it clear that there is just one intended interpretation. Even more significantly, considering all interpretations will allow us to distinguish especially strong types of arguments that hold independent of how we interpret the language.

What it is for a sentence to be a LOGICAL CONSEQUENCE of a set of sentences is for the former to be true on any interpretation in which every sentence in the latter is true. Although we will improve on this characterization in later chapters, we may draw on an intuitive understanding of the interpretations of a language for the time being. An argument is LOGICALLY VALID just in case the conclusion of the argument is a logical consequence of its set of premises.

Functionally, it is helpful to think of logically valid arguments as arguments that can be relied on no matter how (or whether) you understand the non-logical terms like 'is gold' or 'Socrates' that occur in the sentences of the argument. Even though the gold argument is extremely compelling when we maintain an intended interpretation of our language, the gold argument is not a logically valid argument since there is an interpretation of the language in which its premises (just one in this case) are true and its conclusion is false.

We have identified the second way that an argument can go wrong: it can admit of an interpretation in which the premises are true but the conclusion is false. Recall Gyre and those slithy mome raths from before. We may not know much about these sorts of things, but we can be sure that Gyre is slithy if Gyre is a mome rath and all mome raths are slithy. We may know considerably more about Socrates being human and mortal, but similar reasoning applies. Indeed, these two arguments may be observed to have the same *logical form*. It is these logical forms of reasoning that make up the subject-matter of this course.

When an argument has neither of the defects considered above— i.e., when it is both logically valid and has true premises on the intended interpretation— we may say that it is *sound*. Sound arguments are a good thing, but fall outside the scope of this course. Why is that? Because securing the truth of the premises is often an empirical (or in general an extra-logical) matter and presumes that a single interpretation is to be privileged over the others. Rather, we will only be concerned with identifying which arguments are logically valid, not which interpretation we should focus on or which premises are true on that interpretation. We will also exclude consideration of a wider understanding of valid arguments which includes arguments that are really convincing— e.g., in a court of law— but not logically valid. Accordingly, we will refer to arguments simply as *valid* or, when they are not valid, as *invalid*.

A parting question: how would you begin to describe the space of all valid arguments? It is a great intellectual achievement of the late 19th and early 20th centuries that we have devised systematic methods for answering this question (relative to a language). In this course we will consider two types of answers, one belonging to proof theory, and the other belonging to model theory (also called semantics). In the metalogical portions of this course, we will show how these two methods converge on the same answer, describing one and the same space of valid forms of reasoning despite doing so in radically different ways.

0.4 Logical Form

We've seen that a valid argument does not need to have true premises nor a true conclusion. Conversely, having true premises and a true conclusion on a particular interpretation is not enough to make an argument valid. Consider this example:

J1. Kamala Harris is a U.S. citizen.
J2. Justin Trudeau is a Canadian citizen.
J3. UBC is the largest employer in Vancouver.

The premises and conclusion of this argument are all true on the intended interpretation. Nevertheless, this is quite a poor argument. In particular, the definition of validity is not satisfied: there are interpretations in which the premises are true while the conclusion is false. Although the conclusion is true on the intended interpretation, this may fail to hold on other interpretations. For example, we may interpret 'UBC' to mean what 'Lululemon' does on the intended interpretation. Accordingly, the premises are true, and yet the conclusion is false.

The important thing to remember is that validity is not about the truth or falsity of the sentences in the argument on any particular interpretation. Instead, it is about the *logical form* of the argument. But what is the logical form of an argument?

We have begun to see some valid arguments like the Socrates argument and the Gyre argument. But was that one argument or two? Recall that an argument in English is a sequence of declarative sentences in which the conclusion occurs on the last line. Since the sentences in the Socrates and Gyre arguments differed, they are different arguments. Nevertheless, these arguments share the same logical form. Here is a valid argument with a different logical form:

K1. Oranges are either fruits or musical instruments.
K2. Oranges are not fruits.
K3. Oranges are musical instruments.

This is a valid argument: there is no interpretation in which the premises are true and the conclusion is false. Since, given the intended interpretation, it has a false premise— premise K2— it does not establish its conclusion (it is not a sound argument), but it does have a valid *logical form*. Here is another example of an argument with a valid logical form:

L1. If it is raining, then the streets are wet.
L2. The streets are not wet.
L3. It is not raining.

As we will see, there are many logical forms that arguments can have. In order to characterize the abstract forms themselves, we will use variables. To begin with, we will consider the variables 'φ', 'ψ', 'χ', ... for sentences, calling these SCHEMATIC VARIABLES. Schematic variables allow us to talk about the sentences of a language but they are not themselves sentences of that language. Rather φ, ψ, and χ have sentences as *values*. It is helpful to compare the use of variables like 'x' in mathematics. The symbol 'x' is used to stand for a number but 'x' is not itself a name for a number the way that '2' is a name for the number two. We'll return to this distinction in Chapter 1, and again later on when we begin to prove general facts about the formal languages and proof systems that we will consider.

Without introducing any notation beyond schematic variables for sentences, we may represent the logical form of the previous argument as follows:

M1. If φ, then ψ.
M2. It is not the case that ψ.
M3. It is not the case that φ.

Instead of an argument itself, what we have above is a recipe where substituting declarative sentences for the variables 'φ' and 'ψ' returns an argument that, like the raining argument, is valid. By replacing sentences with variables, we were able to abstract away the non-logical parts of the raining argument, leaving behind the logical form of the original raining argument. We may refer to the result as an ARGUMENT SCHEMA. Argument schemata are built up out of three elements: schematic variables, punctuation, and LOGICAL CONSTANTS. The logical constants included above were represented using 'If..., then...' and 'It is not the case that...'. We will introduce more elegant representations of these logical constants in the next chapter. Until then, it is worth considering whether we can identify an argument schema for the Socrates argument in the very same way as in the raining argument.

Suppose that we were to maintain our restriction to schematic variables for sentences from before. The plan is to replace sentences with variables and try to recover an argument schema just like we did previously. Since 'Socrates is human' is a sentence and doesn't have any parts that are also sentences, it is an ATOMIC SENTENCE, and so all we can do is replace it with a variable. Let's choose 'φ'. We find something similar in considering the second premise 'Every human is mortal'. Suppose we choose the 'ψ'. The conclusion is also an atomic sentence, and so let's substitute 'χ'. This returns the following argument schema:

N1. φ
N2. ψ
N3. χ

Although this does *schematize* the Socrates argument, it does not leave behind anything which we might appeal to in explaining why the Socrates argument was valid. For instance, if we replace the variables with any other sentences, we do not necessarily get a valid argument. Have we made some mistake? No. Rather, the validity of the Socrates argument is not visible at the logical resolution that we have been working. Instead of abstracting on sentences, we need to analyze the sub-sentential parts, identifying logical constants at this higher level of logical resolution. In particular, we will need to split sentences up into predicates and singular terms, introducing logical constants for quantification like ‘for all’ and ‘there is some’. This ambition will be addressed in later chapters on *First-Order Logic* (FOL). Until then, we will keep things simple to start, focusing on *Propositional Logic* (PL).

0.5 Other Logical Notions

We have begun to characterize the subject-matter of logic by schematizing arguments. As brought out above, substituting schematic variables for the non-logical elements of an argument can be used to identify the logical form of the argument. Given a sufficient degree of logical resolution power, we may appeal to the logical form of an argument in order to explain why the argument is valid. We will make this process precise in the following chapter by restricting attention to a rigorously defined formal language. For the time being, we may conclude the present chapter by introducing a few more terms to look out for in what follows.

0.5.1 Tautologies, Contradictions, and Logical Contingencies

Instead of being concerned with the truth value of sentences on any particular interpretation, we will be concerned with the truth-values of sentences across all interpretations of the language in question where interpretations will be carefully defined. For instance, in a valid argument, the conclusion is true in any interpretation in which all of the premises are true even though neither the true premises nor conclusion are required to be true in any particular interpretation. Nevertheless, there are certain sentences that are true on all interpretations. To bring this out, compare the following sentences:

O1. It is raining.
O2. Either it is hot outside, or it is not hot outside.
O3. John is sitting down and it is not the case that John is sitting down.

Sentence O1 is true on some interpretations and false on others and so is said to be LOGICALLY CONTINGENT or just CONTINGENT for short. Sentence O2 is different. Even though we may not know what the weather is like, we know that it is either hot or it isn’t. Moreover, O2 is true no matter how you interpret ‘It is hot outside’, and so O2 is true on all interpretations. Accordingly, O2 is a LOGICAL TRUTH, or what is also called a TAUTOLOGY. By contrast, sentence O3 is false on all interpretations, and so is referred to as LOGICALLY FALSE or a CONTRADICTION. In particular, you do not need to know what John is up to, or know how to interpret the sentence ‘John is sitting down’ in order to know that O3 is false.

0.5.2 Logical Entailment and Equivalence

In addition to the logical properties that sentences may have on their own, we may also consider the logical relations that hold between two sentences. For example:

P1. Clara went to the store.
P2. Someone went to the store.

Regardless of how we interpret the sentences above, P2 is true in any interpretation in which P1 is true. Put otherwise, P1 LOGICALLY ENTAILS P2. Whereas logical consequence related zero or more premises two a single conclusion, logical entailment relates exactly two sentences. Nevertheless, one sentence logically entails another just in case the latter is a logical consequence of the former.

When two sentences logically entail each other, those sentences are said to be LOGICALLY EQUIVALENT. For instance, consider the following sentences:

Q1. If Sunil went to the store, then he washed the dishes.
Q2. If Sunil did not wash the dishes, then he did not go to the store.

Not only does Q1 logically entail Q2 but Q1 is also logically entailed by Q2. It follows that Q1 and Q2 have the same truth-value in all interpretations. For contrast, consider:

R1. Edinburgh is North of London.
R2. London is South of Edinburgh.

Although these sentences have the same truth-value in the intended interpretation, they do not have the same truth-value in all interpretations. For instance, if we took 'is North of' to mean what 'has a larger population than' means on the intended interpretation while maintaining the intended interpretation of the other terms, then R2 would be true and R1 would be false, and so R2 does not entail R1. It follows that R1 and R2 are not logically equivalent: they do not have the same truth-value in all interpretations.

Not only does R2 fail to entail R1 we may show that R1 does not entail R2 since there is an unintended interpretation where R1 is true and R2 is false. In a similar manner to above, we might take 'is South of' to mean what 'has a smaller population than' means on the intended interpretation while maintaining the intended interpretation of the other terms. This also suffices to show that R1 and R2 are not logically equivalent.

Although one may well argue form R1 to R2 or *vice versa*, these arguments do not hold by virtue of logic alone, but rather by virtue of the intended interpretation of 'is North of' and 'is South of'. Nevertheless, these are powerful arguments in contexts in which we are happy to hold an intended interpretation fixed given a common understanding of the language.

0.5.3 Satisfiability

Consider these three sentences:

S1. Sam is shorter than John.
S2. Sam is taller than John.
S3. If Sam is shorter than John, then Sam is not taller than John.

Independent of how we interpret the sentences above we may determine that there is no interpretation which makes all of these sentences true. For instance, we might reason as follows: suppose there were some interpretation which makes all three sentences true. However, it follows from S1 and S3 that S2 is false. This contradicts our supposition. Since we have arrived at a contradiction on the supposition that there was an interpretation that makes all three sentences true, we may reject our supposition, concluding that there is no interpretation which makes all three sentences true.

A set of sentences is SATISFIABLE just in case there is some interpretation which makes every sentence in the set true, and UNSATISFIABLE otherwise. In particular, the set $\{\mathrm{S1}, \mathrm{S2}, \mathrm{S3}\}$ is unsatisfiable. Given an unsatisfiable set of sentences Γ, if every sentence in Γ is also a sentence in Σ, then Σ is also unsatisfiable. The opposite, however, is not true: if every sentence in Δ is in Γ and Γ is unsatisfiable, it does not follow that Δ is unsatisfiable. For instance, every subset of $\{\mathrm{S1}, \mathrm{S2}, \mathrm{S3}\}$ which has at most two members is satisfiable. This includes the empty set $\varnothing$ since, vacuously, every sentence in $\varnothing$ is true in all interpretations.

Here is another concrete example of a satisfiable set of sentences:

T1. The Earth has more than one moon.
T2. Jupiter has exactly one moon.

Even though both of the sentences above are false on the intended interpretation, the set of sentences $\{\mathrm{T1}, \mathrm{T2}\}$ is satisfiable on account of the existence of an interpretation in which both sentences are true. For instance, we might interpret 'Jupiter' to mean what 'Earth' means on the intended interpretation and interpret 'Earth' to mean what 'Jupiter' means on the intended interpretation while maintaining the intended interpretation of the other terms.

Despite the fact that satisfiability and unsatisfiability applies to sets of sentences rather than individual sentences, we may observe that any set of sentences which includes a contradiction is unsatisfiable. Similarly, if Γ' is a set of sentences that results from adding a tautology to a set of sentences Γ, then Γ' is satisfiable just in case Γ' is satisfiable. In particular, any set of sentences which only includes tautologies will be satisfiable. We will have much more to say about the role that satisfiability will play in studying proof systems in later chapters. But before all of that it will be important to introduce a formal language.

So far we have relied on an intuitive conception of an interpretation in order to introduce such notions as satisfiability and logical consequence, concepts which are of central importance in logic. However familiar English may be, there is no formally precise definition of what counts as an English sentence, and so there is no definition to be had of what counts as an interpretation of the sentences of English. In order to provide a mathematically precise definition of an interpretation, it will be important to provide an equally precise definition of the language we are interpreting. We will attend to this task in the following chapter.

Chapter 1

Propositional Logic

This chapter introduces an artificial language $\mathcal{L}^{\text{PL}}$ for *Propositional Logic*. The basic units of this language are complete sentences which, given an interpretation, express propositions. Since we will only be concerned with whether a proposition is obtains or does not, interpreting $\mathcal{L}^{\text{PL}}$ will amount to assigning the sentences of $\mathcal{L}^{\text{PL}}$ to either truth or falsity which we will represent by '1' and '0'. How this goes will be the topic of the next chapter, focusing for the time being on the construction of the sentences of $\mathcal{L}^{\text{PL}}$ and translating English sentences and arguments into $\mathcal{L}^{\text{PL}}$. This brings us to our first definition:

> A REGIMENTATION of an English sentence in $\mathcal{L}^{\text{PL}}$ is any sentence in $\mathcal{L}^{\text{PL}}$ which captures (some amount of) the logical form of that English sentence.

This definition is vague, and necessarily so. As we will see, there will typically be more than one way to regiment a sentence in English, and different regimentations may capture more or less of the English sentence's logical form. Rather than admitting a mathematical definition, regimentation is like any translation an imprecise matter where some regimentations are better than others, and others may be on a par with each other. We may then say:

> A REGIMENTATION of an argument in English is an argument in $\mathcal{L}^{\text{PL}}$ whose sentences regiment the sentences of the argument in English.

Recall from before that an argument in English is a sequence of declarative sentences. We will see a number of examples of arguments and their regimentations throughout this chapter. However vague, it is good to have the definitions above in mind as you read, considering other ways that you might regiment the sentences and arguments that we consider.

1.1 Sentence Letters

In $\mathcal{L}^{\text{PL}}$, the capital Roman letters 'A', 'B', 'C', ... with or without natural numbers for subscripts are the SENTENCE LETTERS of $\mathcal{L}^{\text{PL}}$. These are the basic building blocks from which complex sentences will be constructed. Since a sentence letter could regiment any English sentence, it is important to provide a SYMBOLIZATION KEY which specifies which sentence letters represent which English sentences. For example, consider this argument:

A1. Today is New Year's Day.
A2. If today is New Year's Day, then people are swimming in English Bay.
A3. People are swimming in English Bay.

This is a valid argument in English. In regimenting it, we want to preserve the logical form of the argument which makes it valid. What happens if we replace each sentence with a letter? Our symbolization key would look like this:

A: Today is New Year's Day.
B: If today is New Year's Day, then people are swimming in English Bay.
C: People are swimming in English Bay.

We could then regiment the argument in this way:

B1. A
B2. B
B3. C

This is a regimentation of the argument, but it's not a very interesting one. In particular, our regimentation does not encode any logical connection between A1, A2, and A3. What was compelling about the original argument has been lost in translation. After all, 'A', 'B', and 'C' could be any sentences whatsoever. Just because 'A' and 'B' are true (on a given interpretation), it does not follow that 'C' is also true (on that interpretation).[1]

The symbolization key provided above is by no means the only symbolization key that we could have provided. It is important to observe that A2 is not just *any* sentence. Rather, A2 contains the A1 and A3 *as parts*. Thus, our symbolization key for the New Year's argument only needs to include the following sentences since we can build A2 from just these pieces. Consider the following alternative to the symbolization key given above:

[1] We will provide a formal definition of validity for $\mathcal{L}^{\text{PL}}$ in Chapters 2 and 8.

T: Today is New Year's Day.
S: People are swimming in English Bay.

Although it is often convenient to use letters corresponding to the sentences' subject matter as in the example above, no such requirement is built into the rules of $\mathcal{L}^{\text{PL}}$. We may now use the key given above to provide a more interesting regimentation of the argument:

C1. T
C2. If T, then S.
C3. S

By making use of the English expression 'If... then...', we have managed to preserve enough of the logical structure of the argument in English to provide a valid regimentation. For our formal language, we ultimately want to replace all of the English expressions with logical notation, but this is a good start.

The English sentences that can only be regimented in $\mathcal{L}^{\text{PL}}$ by sentence letters are called ATOMIC SENTENCES. As we will see in later chapters, the internal structure of atomic sentences may be encoded in a formal language which includes predicates and singular terms. However, for the time being, we do not have these expressive resources at our disposal. Instead, atomic sentences are the smallest logical joints at which we may carve while regimenting English sentences in $\mathcal{L}^{\text{PL}}$. Accordingly, the internal structure that an English sentence might have (i.e., its sub-sentential logical form) is lost when regimented by a sentence letter. From the point of view of $\mathcal{L}^{\text{PL}}$, the sentence letters are as basic as it gets. Although the sentence letters can be used to build up more complex sentences, they cannot be taken apart.

1.2 The Sentential Operators

Sentential operators are used to build complex sentences from sentence letters. Here are five common sentential operators which we will be able to express in $\mathcal{L}^{\text{PL}}$:

symbol	what it is called	rough translation
$\neg$	negation	'It is not the case that...'
$\wedge$	conjunction	'Both... and ...'
$\vee$	disjunction	'Either... or ... (or both)'
$\rightarrow$	conditional	'If ... then ...'
$\leftrightarrow$	biconditional	'... if and only if ...'

Natural languages like English are vague and imprecise, and carry many complex subtleties of meaning. Providing a descriptive theory of these complexities belongs to linguistics, not logic. In contrast to English, our formal language $\mathcal{L}^{\text{PL}}$ will be clear and precise, defined by explicit rules that hold without exception. This precision and universality has many advantages, but also comes at a cost: our language is artificial insofar as it's conventions are entirely stipulated, and who is to say which stipulations are the right ones to make?

The question of which logic to accept for which applications is a deep and controversial issue within philosophical logic. Rather than attempting to settle that question here, it will be enough for our purposes here to appeal to one method by which we may evaluate competing logics: abduction. By contrast to inductive arguments, or the deductive arguments with which we will primarily be concerned, abductive arguments draw support from the results that a theory yields. The reason that classical logic (i.e., the propositional and first-order logics that we will be considering) holds the majority among logicians and philosophers is due to its strength and simplicity, making it of great utility for a wide range of applications.

To take just one example, mathematics is almost entirely conducted in a first-order theory. For instance, set theory may be articulated with the expressive resources that we will provide. Nevertheless, the logics that we will consider also have their limits. For instance, the modal logics that you would learn about in an intermediate logic course have also been shown to have powerful applications within linguistics, computer science, and philosophy, and may be naturally combined with the logics with which we will be concerned. Rather than any kind of stopping point, the logics covered in this course make for a natural place to begin.

Despite the advantages afforded by the classical logics we will be studying, these logics will be rather artificial by comparison to the informal patterns of reasoning in English with which you are already familiar. Consequently, the "translations" provided by the table above are only approximate. We'll see some of the differences come out below.

It is also important to mention that although the conventions we will use here are common, there are other conventions used elsewhere. Here is a table with some alternatives that you might come across elsewhere (and should avoid using here):

symbols used here	symbols used elsewhere
$\neg$	$\sim$
$\wedge$	$\&$
$\vee$	$\mid$
$\rightarrow$	$\supset$
$\leftrightarrow$	$\equiv$

Although these symbols would do just as well, they are not quite as common in modern texts or else have come to take on other meanings. They can also be somewhat harder to write on the blackboard with the exception of $\sim$ which we will use in class in order to ease exposition.

It is also worth noting that just because the same symbol that we will use has been used elsewhere does not mean that it picks out the same thing. In general, formal texts like this define their own conventions and you should assume the same for other texts. Nevertheless, the conventions used here are extremely common.

1.3 Negation

Consider how we might regiment these sentences:

D1. Logic is hard.
D2. It is false that logic is hard.
D3. Logic isn't hard.

In order to regiment sentence D1, we will need one sentence letter as below:

H: Logic is hard.

Since sentence D2 is obviously related to sentence D1, we do not want to introduce a different sentence letter since this would obscure their logical relationship. To put it partly in English, D2 may be partially regimented as 'It is not the case that H.' In order to regiment D2 along these lines, we will use the symbol '$\neg$' for negation. Thus we may regiment D2 as '$\neg H$'. A sentence of this type— one that begins with a '$\neg$' symbol— is called a NEGATION. The sentence it negates— in this case 'H'— is called the NEGAND.

What of D3? It says that logic isn't hard, which is just another way of negating D3. Accordingly, we can regiment D3 in the same way that we regimented D3 with '$\neg H$'.

When regimenting English sentences in $\mathcal{L}^{\text{PL}}$, the word 'not' is usually a pretty good clue that '$\neg$' will be an appropriate symbol to use. But it's important to think about the actual meaning of the sentence, and not rely too much on which words appear in it.

> NEGATION TEST: For any sentence φ, a sentence can be regimented by $\neg\varphi$ if it can be paraphrased in English as 'It is not the case that φ'.

Consider the following examples:

E1. Rodrigo is mortal.
E2. Rodrigo is immortal.
E3. Rodrigo is not immortal.

Suppose we let 'R' regiment E1. What about sentence E2? Since being immortal is the same as not being mortal, we may take E2 to be the negation of E1, regimenting it with '$\neg R$'.

Sentence E3 can be paraphrased as 'It is not the case that Rodrigo is immortal'. Using negation twice, we may regiment E3 by '$\neg\neg R$'. The two negations in a row each work as negations, so the sentence means 'It is not the case that, it is not the case that R'. It is the negation of the negation of 'R'. One can negate any sentence of $\mathcal{L}^{\text{PL}}$— even a negation— by putting the '$\neg$' symbol in front of it. In the case of '$\neg\neg R$', this is a negation whose negand is '$\neg R$', which in turn is a negation whose negand is 'R'.

But sometimes things are not quite a simple as we might initially expect. Here is an example that illustrates some of the complexities to look out for:

F1. Elliott is happy.
F2. Elliott is unhappy.

Suppose we take 'H' to F1. We might be tempted to regiment sentence F2 by '$\neg H$'. But is being unhappy the same thing as not being happy? Here the answer is, 'No'. For instance, Elliott might simply be meh. If you find out that someone is not happy, you cannot infer that they are unhappy (though in some cases this inference might well make sense). Hence, we shouldn't treat F2 as the negation of F1. So long as we are allowing 'unhappy' to mean something besides 'not happy', then we need to use a new sentence letter to regiment F2.

Although we will have more to say about this in the following chapter, it is important to provide a preliminary sense of when a negation is true. In particular, for any sentence φ, if φ is true, then $\neg\varphi$ is false. Similarly, if φ is false, then $\neg\varphi$ is true. Using '1' in place of 'true' and '0' in place of 'false', we can summarize this in the following TRUTH TABLE for negation:

φ	$\neg\varphi$
1	0
0	1

The left column shows the possible truth-values for the negand and the right column shows the corresponding truth-value of the negation. Accordingly, the truth table given above specifies the *truth-conditions* for '$\neg$', i.e., the conditions under which a negated sentence is true (similarly false). By using numerals, we avoid any clash with our sentence letters.[2]

Since 1 and 0, are the only possible values that we will consider in this course, the truth table above defines a function from the truth-value of the negand to the truth-value of the negation. We will refer to such functions from truth-values to truth-values as TRUTH-FUNCTIONS. You can think of these as providing a meaning for the logical terms that we will use where these meanings are held fixed across all interpretations of our language. It is for this reason that the sentential operators are also sometimes called LOGICAL CONSTANTS. We will have much more to say about all of this in the following chapter. For now we may continue to introduce the remaining sentential operators that we will include in the language $\mathcal{L}^{\text{PL}}$.

[2]Note that *Carnap* will not have this virtue: truth tables will use 'T' for *true* and 'F' for *false*.

1.4 Conjunction

Consider the following sentences:

G1. Jessica is strong.
G2. Luke is strong.
G3. Jessica is strong and Luke is also strong.

At the very least, we will need separate sentence letters for G1 and G2:

J: Jessica is strong.
L: Luke is strong.

Sentence G3 can be paraphrased as 'J and L'. In order to fully regiment this sentence, we will use the CONJUNCTION symbol '$\wedge$' for 'and'. We may then take '$J \wedge L$' to regiment G3. Given any conjunction, the sentences to the left and right of '$\wedge$' are referred to as CONJUNCTS. In the case of G3, both 'J' and 'L' are conjuncts.

Notice that we make no attempt to provide a distinct symbol for 'also' as it occurs in sentence G3. Words like 'both' and 'also' function to draw our attention to the fact that two sentences are being conjoined but are not doing any further logical work. Thus we do not need to represent 'both' and 'also' in $\mathcal{L}^{\text{PL}}$. Note that Sentence G3 would have meant the same thing had it simply said 'Jessica is strong and Luke is strong'.

Here are some more examples:

H1. Jessica is strong and grumpy.
H2. Jessica and Matt are strong.
H3. Although Luke is strong, he is not grumpy.
H4. Matt is strong, but Jessica is stronger than Matt.

Sentence H1 is a conjunction. Since the sentence says two things about Jessica, it is natural to use her name only once. It might be tempting to try this when regimenting H1. Letting 'J' regiment 'Jessica is strong', one might attempt to begin by paraphrasing H1 as 'J and grumpy', but this would be a mistake. After all, 'J' is just a sentence letter and $\mathcal{L}^{\text{PL}}$ doesn't keep track of the fact that it was intended to be about Jessica. Moreover, 'grumpy' is not a sentence, and so on its own it is neither true nor false. Instead, we may paraphrase sentence H2 as the conjunction '$J \wedge G_1$' where 'G_1' regiments 'Jessica is grumpy'. More generally:

Conjunction Test: A sentence can be regimented by $(\varphi \wedge \psi)$ if it can be paraphrased in English as 'Both φ and ψ' where each conjunct is a sentence.

Sentence H2 says one thing about two different subjects. It says of both Jessica and Matt that they are strong, and in English we use the word 'strong' only once. In regimenting sentences in $\mathcal{L}^{\text{PL}}$, we want to make sure each conjunct is a sentence on its own, and so we may paraphrase H2 by repeating the predicate: 'Jessica is strong and Matt is strong.' Once we add a new sentence letter 'M' for 'Matt is strong', we may take '$J \wedge M$' to regiment H2.[3]

Sentence H3 is a bit more complicated. The word 'although' tends to suggest a kind of contrast between the first part of the sentence and the second part. Nevertheless, the sentence is still telling us two things: Luke is strong and he is not grumpy. So we can paraphrase sentence H3 as, 'Luke is strong and Luke is not grumpy.' The second conjunct contains a negation, so we may paraphrase further: 'Luke is strong and it is not the case that Luke is grumpy'. Letting 'G_2' regiment 'Luke is grumpy', we can take '$L \wedge \neg G_2$' to regiment H3.

Once again, this is an imperfect translation of the English sentence H3. Whereas H3 suggests that there is a contrast between Luke begin strong and not grumpy, our regimentation merely says that Luke is strong and not grumpy. Nevertheless, our regimentation preserves some of the important features of the original sentence, specifically the logical features of that sentence. That is, the sentences says that Luke is strong and that Luke is not grumpy.

The word 'but' in H4 indicates a similar contrast between its conjuncts. Since contrasts like this are irrelevant for the purpose of regimenting sentences in $\mathcal{L}^{\text{PL}}$, we can paraphrase the sentence as 'Matt is strong and Jessica is stronger than Matt. It remains to say how to regiment the second conjunct. We already have the sentence letters 'J' and 'M', but neither of these says anything comparative. Thus we need an entirely new sentence letter. Letting 'S' regiment 'Jessica is stronger than Matt', we may take '$M \wedge S$' to regiment H4.[4]

Here is the resulting symbolization key:

J: Jessica is strong.
G_1: Jessica is grumpy.
M: Matt is strong.
G_2: Luke is grumpy.
S: Jessica is stronger than Matt.

It is important to keep in mind that the sentence letters 'J', 'G_1', 'M', 'G_2', and 'S' have no meaning beyond their truth-values. We used 'J' and 'M' to regiment different English

[3] One might contest this claim by instead taking 'Jessica and Matt' to name a plurality where 'are strong' is a plural predicate. We will be overlooking such complexities which are better handled in plural logics.

[4] In chapter 7, we will learn an even better way to regiment relations.

sentences that are about people being strong, but this similarity is completely lost when we regiment sentences in $\mathcal{L}^{\text{PL}}$. Nor does $\mathcal{L}^{\text{PL}}$ recognize any similarity between 'G_1' and 'G_2'. Such connections will be preserved in $\mathcal{L}^{\text{FOL}}$, but for the time being we will accept these limitations.

For any two sentences φ and ψ of $\mathcal{L}^{\text{PL}}$, the conjunction $\varphi \wedge \psi$ is true if and only if both of its conjuncts— φ and ψ— are true. We can summarize this in the truth table for conjunction:

φ	ψ	$\varphi \wedge \psi$
1	1	1
1	0	0
0	1	0
0	0	0

The two left columns indicate the truth-values of the conjuncts. Since there are four possible combinations of truth-values, there are four rows. The conjunction is true when both conjuncts are true, and false otherwise. Whereas negation takes one truth-value as input, and returns one truth-value as output, conjunction takes two truth-values as inputs (the truth-values of its conjuncts) and returns a single truth-value as an output (the truth-value of the conjunction). Put otherwise, conjunction is a *binary operator* and negation is a *unary operator*. Despite this difference, the truth table given above specifies the truth-conditions for conjunction in a similar manner to the way we specified truth-conditions for negation.

Note that conjunction is commutative: $\varphi \wedge \psi$ always has the same truth-value as $\psi \wedge \varphi$. We may then ask: is 'and' in English commutative? Consider the following claims:

I1. Dan went home and took a shower.
I2. Dan took a shower and went home.

Do these sentences say the same thing? Not really. Often the order of the conjuncts suggests the temporal order of the events. Accordingly, we may take I1 to claim that *first* Dan went home, and only *then* did he take a shower, whereas I2 says that these events happened in the opposite order. But what of our truth table for conjunction?

Although $\mathcal{L}^{\text{PL}}$ helps to identify certain logical features in English, it cannot recover everything that we might want to recover. This doesn't mean that one cannot provide a non-commutative theory of conjunction, but we won't be providing such a theory in this course. Rather we stipulate that conjunction for the purposes of this course is commutative where the truth table for conjunction encodes this stipulation. Put otherwise, you can think of the type of conjunction we will be concerned with as an abstraction from the complexities that conjunction in English may include. This doesn't make the commutative theory of conjunction less interesting than a non-commutative theory. After all, simplicity is a good thing, providing for broad applications where a non-commutative notion of conjunction may get in the way.

It is worth considering an example to make these claims more concrete. For instance, consider mathematical claims. Assuming that all claims in pure mathematics are eternal— they are true at all times if they are true at any time— we do not need to keep track of temporal order when engaging in mathematical reasoning. Moreover, even if one were concerned to encode temporal order, getting to grips with $\mathcal{L}^{\text{PL}}$ and $\mathcal{L}^{\text{FOL}}$ will provide an important foundation.

1.5 Disjunction

Consider the following sentences and symbolization key:

J1. Denison will climb with me or he will watch movies.
J2. Either Denison or Ellery will climb with me.

D: Denison will climb with me.
E: Ellery will climb with me.
M: Denison will watch movies.

Sentence J1 may be paraphrased 'Either D or M'. To fully regiment J1, we will introduce the DISJUNCTION symbol '$\vee$' where the sentences to its left and right are called DISJUNCTS. We may then take the disjunction '$D \vee M$' to regiment J1 where and D and M are the disjuncts. It is important to stress that '$\vee$' expresses an *inclusive* reading of disjunction which requires *at least one disjunct* to be true, admitting the possibility where both disjuncts are true.

Sentence I2 is only slightly more complicated. Although there are two subjects, the English sentence only includes the verb once. We can paraphrase I2 as 'Either Denison will climb with me or Ellery will climb with me.' We may then take '$D \vee E$' to regiment I2.

Disjunction Test: A sentence can be regimented by $(\varphi \vee \psi)$ if it can be paraphrased in English as 'Either φ or ψ' where each disjunct is a sentence.

Since '$\vee$' is a symbol that we have introduced, we get to *stipulate* its truth-conditions. And we stipulate that we are using inclusive or, so that $A \vee B$ is true provided that at least one disjunct is true, including the scenario where both are true. Is this a good stipulation for how to treat 'or'? In a study of the semantics of English, it would be appropriate to pursue this question much further. But this book is about logic, not linguistics. Rather than concerning ourselves with describing the exact patterns by which 'or' is used in English, we will be concerned to identify a formal analogue which is simpler and more consistent in its use. Thus we have good reason to stipulate that '$\vee$' is inclusive even if 'or' in English is not.

So '$D \vee E$' is true if 'D' is true, if 'E' is true, or if both 'D' and 'E' are true. It is false only if both 'D' and 'E' are false. We can summarize this with the truth table for disjunction:

φ	ψ	$\varphi \vee \psi$
1	1	1
1	0	1
0	1	1
0	0	0

Like conjunction, disjunction is a binary operator which takes two truth-values as inputs and returns a single truth-value as output. The truth table above stipulates a meaning for '$\vee$' by indicating the truth-conditions for $\varphi \vee \psi$ where φ and ψ are any sentences whatsoever. We may succinctly restate the truth-conditions for conjunction by observing that $\varphi \vee \psi$ is false when φ and ψ are false, and true otherwise. This makes disjunction inclusive.

Consider the following sentences and symbolization key:

K1. Either you will not have soup or you will not have salad.
K2. You will have neither soup nor salad.
K3. You will have soup or salad, but not both.

S_1: You will get soup.
S_2: You will get salad.

Sentence K1 may be paraphrased by 'Either it is not the case that you will have soup, or it is not the case that you get salad.' Regimenting this claim requires both disjunction and negation since it is the disjunction of two negations '$\neg S_1 \vee \neg S_2$'.

Sentence K2 also requires negation. It can be paraphrased as, 'It is not the case that: either that you get soup or that you get salad.' In other words, it is a negation of a disjunction. We need some way of indicating that the negation does not just negate the right or left disjunct, but rather the entire disjunction. In order to do this, we will put parentheses around the disjunction as in '$\neg(S_1 \vee S_2)$'.[5] The parentheses are doing important work, since the sentence '$\neg S_1 \vee S_2$' regiments 'Either you will not have soup or you will have salad', which is different.

Since K3 has a more complicated structure, we can avoid making mistakes by break it into two parts. The first part says that you will have soup or you will have salad. We regiment this by '$(S_1 \vee S_2)$'', including parentheses so that we don't get mixed up later on. The second part says that you will not have both soup and salad. We can paraphrase this as, 'It is not the case that you will have soup and you will have salad' which we may regiment with '$\neg(S_1 \wedge S_2)$'. In order to put these two parts together we may recall that 'but' is typically regimented by conjunction. Thus we may take '$(S_1 \vee S_2) \wedge \neg(S_1 \wedge S_2)$' to regiment K3.

[5] A second, equivalent, way to regiment K2 is '$(\neg S_1 \wedge \neg S_2)$'. We'll see why this is equivalent later on. The equivalence of '$\neg(A \vee B)$' and '$(\neg A \wedge \neg B)$' is an instance of one of De Morgan's laws.

1.6 The Material Conditional

Consider the following sentences and symbolization key:

L1. If you cut the red wire, then the bomb will explode.
L2. The bomb will explode only if you cut the red wire.
L3. The bomb will explode if you cut the red wire.

R: You cut the red wire.
B: The bomb will explode.

Sentence L1 can be partially regimented by 'If R, then B'. We will use the MATERIAL CONDITIONAL symbol '$\rightarrow$' (or CONDITIONAL for short) where the sentence on the left is the ANTECEDENT and the sentence on the right is the CONSEQUENT. Thus we may take '$R \rightarrow B$' to regiment L1 where 'R' is the antecedent and 'B' is the consequent.

The sentence L2 is also a conditional sentence. Since the word 'if' appears in the second half of the sentence, it might be tempting to regiment this in the same way as sentence L1. However, the conditional '$R \rightarrow B$' says what the partial regimentation '*if* R, *then* B' says or, equivalently, what 'B *if* R' says. Neither of these claims say that your cutting the red wire is the *only* way that the bomb will explode. For instance, someone else might cut the wire, or else the bomb might be on a timer. The sentence '$R \rightarrow B$' does not say anything about what to expect if 'R' is false. By contrast, L2 says that the only conditions under which the bomb will explode are ones where you cut the red wire, i.e., if the bomb explodes, then you have cut the wire. Thus sentence L2 may be regimented by '$B \rightarrow R$'.

Notice that we have the same antecedent/consequent pattern for both sentences: the sentence before the 'only if' is the antecedent and the sentence after the 'only if' is the consequent. Hence, we could also write sentence L1 as 'You cut the red wire only if the bomb will explode', which we could regiment by '$R \rightarrow B$'. We find something different in the sentence L3 where the 'if' occurs in the middle. Sentence L3 can be paraphrased by L1, and so regimented in the same way by '$R \rightarrow B$'. Regimenting conditional claims in English is tricky business and will require considerably more care than regimenting negations, conjunctions, and disjunctions.

It is important to remember that the operator '$\rightarrow$' says only that, if the antecedent is true, then the consequent is true. It says nothing about the *explanatory* connection between the antecedent and consequent. For instance, regimenting sentence L2 as '$B \rightarrow R$' does not mean that the bomb exploding would somehow have caused you to cut the wire. Both sentence L1 and L2 suggest that, if you cut the red wire, your cutting the red wire would explain why the bomb exploded. Nevertheless, they differ in the *logical* connection that they assert. If sentence L2 were true, then an explosion would tell us that you had cut the red wire. Without an explosion, sentence L2 tells us nothing about what you did with the red wire.

Material Conditional Test: A sentence can be regimented by $(\varphi \to \psi)$ if it can be paraphrased in English as 'If φ, then ψ', where the antecedent and consequent are both sentences.

For any sentences φ and ψ, if the conditional $\varphi \to \psi$ is true and the antecedent φ is true, then the consequent ψ is also be true. Hence, if the antecedent φ is true but the consequent ψ is false, then the conditional $\varphi \to \psi$ is false. But what is the truth-value of $\varphi \to \psi$ when either φ is false or ψ is true? Suppose, for instance, that the antecedent φ happens to be false. It would seem that the conditional $\varphi \to \psi$ does not assert any claim about the truth-value of the consequent ψ, and so— at least in ordinary English— it is unclear what truth-value $\varphi \to \psi$ should have on the assumption that φ is false.

In English, the truth of conditionals often depends on what *would* be the case if the antecedent *were true* even if the antecedent is false. Put otherwise, such reasoning is not truth-functional, i.e., we need to know more about the sentence in question than just its truth-value. This poses a serious challenge for regimenting conditionals in $\mathcal{L}^{\text{PL}}$. In order to consider what the world would be like if R were true, we would need to know something about what R says about the world and we are quickly led into deep questions about the way the world is, laws of nature, and counterfactual reasoning— topics for philosophy of language, metaphysics, and the philosophy of science, but not this class. Rather, we will restrict consideration to the truth-values of sentences, ignoring circumstances in which their truth-values are different.

What we are after is a truth-function which approximates the meaning of conditional claims in English such as L1 and L2. More specifically, we want to specify the truth-value of $\varphi \to \psi$ as a function of the truth-values for φ and ψ, and nothing else. This is a big limitation, since there are not many truth-functions of two values out there. In fact there are only 16. Thus we will choose the best among them to approximate the 'If... then...' construction in English. We will specify this truth-function with the following truth table:

φ	ψ	$\varphi \to \psi$
1	1	1
1	0	0
0	1	1
0	0	1

Observe that when the antecedent φ is false, the conditional $\varphi \to \psi$ is true regardless of the truth-value of the consequent ψ. If both φ and ψ are true, then the conditional $\varphi \to \psi$ is true. In short, $\varphi \to \psi$ is false just in case φ is true and ψ is false.

More than any other sentential operator, the material conditional provides an extremely rough approximation of English conditional claims in $\mathcal{L}^{\text{PL}}$ with some counter-intuitive consequences. For example, a conditional in $\mathcal{L}^{\text{PL}}$ is true whenever the consequent is true, independent of the truth-value of the antecedent. Additionally, a conditional in $\mathcal{L}^{\text{PL}}$ is true any time the antecedent is false, independent of the truth-value of the consequent. These are odd consequence since at least some conditionals in English with true consequents and/or false antecedents seem clearly to be false. For instance, consider the following examples:

M1. If there are no philosophy courses at MIT, then Logic I is a philosophy course at MIT.
M2. If this book has fewer than thirty pages, then it will win the Pulitzer prize.

Both M1 and M2 seem clearly false. But each of them, regimented in $\mathcal{L}^{\text{PL}}$, would come out true. Sentences such as M1 and M2 are sometimes referred to as *paradoxes for the material conditional* since their regimentations in terms of the material conditional are true. However, such sentences are only paradoxical if we take '$\rightarrow$' to mean what 'If . . . , then . . . ' means in English. Rather, the material conditional is a purely artificial piece of formal vocabulary which we have introduced by stipulating how it's truth-value is determined. Although the material conditional only provides a rough approximation of the 'If . . . , then . . . ' in English, who said that English has the best vocabulary for theoretical applications? After all, mathematics is also full of artificial, entirely stipulated definitions which are nevertheless of great utility in part because of the precise nature of their meanings given their explicit definitions.

Despite the oddness of taking the regimentations of sentences such as M1 and M2 to be true, the material conditional preserves many of the most important logical features of conditionals claims in English. Indeed, the material conditional is perfectly adapted to the purposes of mathematics, and this alone covers a very important part of human reasoning. Programming languages also make use of material conditional claims, and these have proved to be profoundly useful. Rather than trying to capture all the subtleties of conditional English constructions, we may take the material conditional to be an approximation of and abstraction from the complexities of a natural language such as English.

Whereas the truth-conditions for material conditional sentences are precisely defined, there is very little in English for which we have clear definitions, and certainly not the word 'if'! Indeed, philosophers, linguists, and logicians have spent over a century developing sophisticated mathematical theories to model the behaviour of 'If . . . , then . . . ' in English, and we are still very far from any kind of conclusive theory. Given how unclear we are about the meaning of conditionals constructions like 'If . . . , then . . . ' in English, it is difficult to rely on these constructions in theoretical applications. By contrast, the material conditional is easy to understand completely even if it pulls apart from similar sounding claims in English.

We will have lots of occasions to observe the utility of the material conditional throughout this course. For now, I'll just ask you to go along with this approach to conditionals even if the material conditional sometimes seems to provide strange results. Believe it or not, the methods of modern logic have been applied for now over a hundred years attempting, among many other things, to provide a fully adequate regimentation of 'if' as it occurs in English. Although considerable progress has been made, conclusive success is still forthcoming.

1.7 The Material Biconditional

Consider the following sentences and symbolization key:

N1. The figure on the board is a triangle only if it has exactly three sides.
N2. The figure on the board is a triangle if it has exactly three sides.
N3. The figure on the board is a triangle if and only if it has exactly three sides.

T: The figure is a triangle.
S: The figure has three sides.

Sentence N1, for reasons discussed above, can be regimented as '$T \to S$'.

Sentence N2 is different. Since N2 can be paraphrased as, 'If the figure has three sides, then it is a triangle', we may take '$S \to T$' to regiment N2.

Sentence N3 says that T is true *if and only if* S is true. Although we could regiment N3 by conjoining two conditional claims, we will introduce the MATERIAL BICONDITIONAL symbol '$\leftrightarrow$' (or BICONDITIONAL for short) for this purpose. Because we could always write $(\varphi \to \psi) \wedge (\psi \to \varphi)$ instead of $\varphi \leftrightarrow \psi$, we do not strictly speaking *need* to introduce a new symbol for the biconditional.[6] Nevertheless, logical languages often include such a symbol, and in our case $\mathcal{L}^{\text{PL}}$ will include one, making it easier to regiment phrases like 'if and only if'.

Instead of referring to the sentence on the left-hand side of a biconditional as the antecedent and the sentence on the right-hand side as the consequent, we will refer to the sentences on either side of a biconditional as the ARGUMENTS of the biconditional, where this is a general term for the sentences on which a logical constant operates. Whereas negation takes one argument (the negand), all of the other operators in $\mathcal{L}^{\text{PL}}$ take two arguments. It is important to clarify that the arguments of an operator have nothing to do with the arguments consisting of sentences which we will evaluate for logical validity.

> MATERIAL BICONDITIONAL TEST: A sentence can be regimented by $(\varphi \leftrightarrow \psi)$ if it can be paraphrased in English as 'φ if and only if ψ' or 'φ just in case ψ', where the arguments are both sentences.

In order to specify the truth-conditions for sentences like $\varphi \leftrightarrow \psi$, consider the following:

[6] If fact the only truth-function we really need is called 'nand' (not-both), but doing so would be tedious!

φ	ψ	$\varphi \leftrightarrow \psi$
1	1	1
1	0	0
0	1	0
0	0	1

The truth table above indicates the truth-conditions for the biconditional by taking $\varphi \leftrightarrow \psi$ to be true if φ and ψ have the same truth-value, and false if φ and ψ have different truth-values. Although we know that φ and ψ will have different truth-values if $\varphi \leftrightarrow \psi$ is false, this does not tell us whether φ is true and ψ is false, or *vice versa*. Similarly, knowing that $\varphi \leftrightarrow \psi$ is true does not tell us whether both φ and ψ are true, or whether both are false.

1.8 Unless

We have now introduced all of the operators for $\mathcal{L}^{\text{PL}}$. We can use them together to regiment many kinds of sentences. Consider the following examples and associated symbolization key:

O1. Unless you wear a jacket, you will catch a cold.
O2. You will catch a cold unless you wear a jacket.

J: You will wear a jacket.
D: You will catch a cold.

We can paraphrase sentence O1 as 'Unless J, D.' This means that if you do not wear a jacket, then you will catch a cold, and so we may regiment O1 as '$\neg J \rightarrow D$.'

This same sentence O1 also means that if you do not catch a cold, then you must have worn a jacket. With this in mind, we might regiment O1 as '$\neg D \rightarrow J$'. You might then wonder which of these is the correct regimentation of sentence O1. The answer is that both regimentations are correct. Not only are these regimentations logically equivalent, there no reason to prefer one regimentation to another. Rather, both regimentations are equally natural.

What about O2? We may begin by taking 'Unless you wear a jacket, you will catch a cold' to paraphrase O2, and so either '$\neg J \rightarrow D$' or '$\neg D \rightarrow J$' may be taken to regiment O2 where neither is better than the other. As a result, O2 is logically equivalent to sentence O1.

When regimenting sentences like sentence O1 and sentence O2, it is easy to get turned around. Since the conditional is not symmetric, it would be wrong to regiment either sentence as '$J \rightarrow \neg D$' or '$D \rightarrow \neg J$'. Fortunately, there are symmetric ways to regiment O1 and O2. In particular, both sentences say that you will wear a jacket or— if you do not wear a jacket— then you will catch a cold. So we can regiment O1 and O2 as '$J \vee D$'. Although linguistically less natural, this regimentation is easier to remember. It helps that '$Q \vee P$' is logically equivalent to '$P \vee Q$', so if you use disjunction, you don't have to worry about the order.

1.9 Well-Formed Sentences

The sentence 'Apples are not blue' is a sentence of English and '$\neg A$' is a sentence of $\mathcal{L}^{\text{PL}}$. Although we can identify sentences of English when we encounter them, we do not have a precise definition of what counts as an English sentence. Students used to learn grammarians' attempts to formalize some such rules, but contemporary linguists agree that this was a hopeless project. Natural languages like English are just not susceptible to such precisification. By contrast, it is possible to define what counts as a sentence in $\mathcal{L}^{\text{PL}}$ where it is in part for this reason that we introduce an artificial language like $\mathcal{L}^{\text{PL}}$ in the first place.

Whenever a language becomes the object of study, we call the language that is being studied the OBJECT LANGUAGE and the language in which we are conducting our study the METALANGUAGE. The object language we will be concerned with in this chapter is $\mathcal{L}^{\text{PL}}$. In this section, we will provide a precise definition of the sentences of $\mathcal{L}^{\text{PL}}$. The definition itself will be given in the metalanguage which in our case will consist of English enriched with certain amount of mathematical vocabulary, e.g., the schematic variables 'φ' and 'ψ'.

It is vitally important to distinguish between the object language and metalanguage, doing our best to avoid mixing them up. We will be helped by the fact that the sentences of our object language $\mathcal{L}^{\text{PL}}$ are entirely formal, whereas the sentences of our metalanguage are mostly informal, though they may contain some mathematical elements. For instance, the sentence '$\neg A$' is a sentence in the object language $\mathcal{L}^{\text{PL}}$ because it only uses symbols of $\mathcal{L}^{\text{PL}}$. In contrast, the sentence "The expression '$\neg A$' is a sentence of $\mathcal{L}^{\text{PL}}$" is not a sentence of $\mathcal{L}^{\text{PL}}$, but rather a sentence in the metalanguage that we use to talk *about* '$\neg A$' which is a sentence of $\mathcal{L}^{\text{PL}}$.

1.9.1 The Use/Mention Distinction

So far, we have talked a lot *about* sentences and will continue to do so throughout this text. Of course, we have also used sentences to say things, e.g., that there is no precise mathematical definition of the declarative sentences of English. In order to sharpen this contrast, consider these the following sentences:

P1. Kamala Harris is the Democratic Nominee.
P2. 'Kamala Harris' is composed of two uppercase letters and ten lowercase letters.

When we want to talk about Kamala Harris, we *use* her name as in P1. When we want to talk about Kamala Harris' name, we *mention* her name which we do by putting her name in quotation marks as in P2. Similarly, whereas it is true to say that you are learning logic, the expression 'logic' is the name of the subject that you are learning.

In general, when we want to talk about how things are, we *use* expressions in a language. When we want to talk about the expressions of a language, we *mention* those expression. Of course, we need to indicate that we are mentioning expressions rather than using them. To do this, some convention or other is needed. Here is the first convention that we will use:

> Quotes: A quoted expression is the *canonical name* for the expression quoted.

For instance 'ABC' is the name for the expression consisting of the first letter of the alphabet, followed by the second letter of the alphabet, followed by the third letter of the alphabet. Put otherwise, 'ABC' is the result of concatenating the symbols 'A', 'B', and 'C' where speaking loosely, this means that the complex symbol 'ABC' is formed by putting the others together one after the next. Concatenation will play an important role below.

Consider the following sentences:

Q1. 'Kamala Harris' is the Democratic Nominee.
Q2. Kamala Harris is composed of two uppercase letters and ten lowercase letters.

Sentence Q1 says that the expression 'Kamala Harris' is the Democratic Nominee which is false. Rather, it is the *woman* named by the expression 'Kamala Harris' who is the Democratic Nominee, not her *name*. We find a related mistake in P2 which says that the woman Kamala Harris is composed of letters which is also false. Here is another important type of example:

R1. " 'Kamala Harris' " is the name of 'Kamala Harris'.

Whereas on the left we have the name of a name, on the right we have a name. Perhaps this kind of sentence only occurs in logic textbooks, but it is true nonetheless.

It is important to contrast two other uses that quotation marks often have in other contexts: attribution and scare quotes. These uses are connected. For instance, in writing a paper, one might *use* the words of another while nevertheless attributing those words to that author. Here's a concrete example. Say we are discussing Quine's ontology, and we want to say that Quine argues that positing merely possible objects, "offends the aesthetic sense of us who have a taste for desert landscapes." The quoted words belong to Quine, and we want to make this clear to our reader. Nevertheless, we are still *using* Quine's words; we are not merely mentioning them, i.e., naming the string that he wrote.

Along these same lines, one might *use* certain words to make a claim but without wanting to attribute that claim to oneself even though there is no one else to which we might attribute the claim. For instance, one might claim that the song is 'ratchet' to use the slang term but without fully standing by its use. This usage might suggest that others take the song to be ratchet without joining them in doing so. There are many such examples along these and other lines, but this is not the way that we will be using quotation marks in this text.

Although quotes are often helpful in order to indicate that we are talking about an expression rather than using that expression to assert something, quotes are only useful for speaking about the specific expression inside the quotes. In order to speak more effectively about all expressions of $\mathcal{L}^{\text{PL}}$, it will be important to introduce a further device. By way of motivation, the following section will begin to introduce the primitive symbols of $\mathcal{L}^{\text{PL}}$.

1.9.2 Primitive Symbols

In order to define the sentences of $\mathcal{L}^{\text{PL}}$ in a more careful way than we have so far, it will be important to introduce the primitive symbols of $\mathcal{L}^{\text{PL}}$. Some of these we have already seen above. In particular, consider the following definitions for $\mathcal{L}^{\text{PL}}$:

SENTENTIAL OPERATORS: '$\neg$','$\wedge$','$\vee$','$\rightarrow$', and '$\leftrightarrow$'.
PUNCTUATION: '(' and ')'.

Quotes have been used above in order to explicitly indicate the symbols that are taken to be sentential operators and parentheses, respectively. The sentential operators are also commonly called CONNECTIVES since they connect sentences to form new sentences of greater complexity. In our case, the sentential operators above are all truth-functional since the truth-value of a sentence with any of the sentential operators given above as its main operator is determined by the truth-values of its arguments. Accordingly, the sentential operators indicated above are also sometimes called the TRUTH-FUNCTIONAL CONNECTIVES (or TRUTH-FUNCTIONS) as well as EXTENSIONAL OPERATORS. For simplicity, we will refer to the sentential operators above as the OPERATORS of $\mathcal{L}^{\text{PL}}$ since we will not consider non-extensional operators. Additionally, the parentheses indicated above provide the necessary punctuation needed to specify the order of operations in complex sentences in which multiple operators occur.

Given that there are a finite number of operators and parentheses, it is straightforward to indicate them individually using quotation marks. The same cannot be said, however, for the sentence letters which may include infinitely many different subscripts. For instance, consider the following attempt to specify the sentence letters of $\mathcal{L}^{\text{PL}}$:

Attempt 1: 'φ_x' is a SENTENCE LETTER of $\mathcal{L}^{\text{PL}}$ whenever φ is a capital letter of the English alphabet and x is a numeral for a natural number.

Attempt 2: φ_x is a SENTENCE LETTER of $\mathcal{L}^{\text{PL}}$ whenever φ is a capital letter of the English alphabet and x is a numeral for a natural number.

The problem with the first attempt is that even assuming that 'φ' is schematic variable whose values range over all capital letters of the English alphabet and 'x' is a variable ranging over natural numbers, 'φ_x' names one and the same expression every time. In particular, 'φ_x' names the expression which occurs within the quotation marks which consists of a lowercase Greek letter subscripted by a lowercase English letter. By the lights of the first attempt, there is only once sentence letter $\mathcal{L}^{\text{PL}}$, i.e., 'φ_x', contrary to what we intended.

In response to the shortcomings of this first attempt, the second attempt given above removes the quotation marks altogether. Given that φ is any capital letter of the English alphabet and x is any natural number, we may now expect there to be many different sentence letters of $\mathcal{L}^{\text{PL}}$, thereby avoiding the problem that we faced before. In the instance where φ has the first capital letter of the English alphabet 'A' as its value and x has the numeral for the first nonzero natural number '1' as its value, the second attempt asserts that A_1 is a sentence letter of $\mathcal{L}^{\text{PL}}$. Similarly, B_3, H_0, W_{27}, and so on are all taken to be sentence letters by the lights of the second attempt. Although this may seem to cut considerably closer to what we want, the second attempt falls short by *using* the sentence letters that it aims to introduce rather than *mentioning* them as we might otherwise intend.

In order to press the previous point, it is worth considering one more attempt to define all of the sentence letters of $\mathcal{L}^{\text{PL}}$. Rather than using variables, one might provide paradigm cases:

Attempt 3: 'A_0','A_1', ...,'B_0','B_1', ...,'Z_0','Z_1', ... are the SENTENCE LETTERS of $\mathcal{L}^{\text{PL}}$.

All of the sentence letters indicated above are mentioned rather than used, where this is just what we want in order to complete our specification of the primitive symbols of $\mathcal{L}^{\text{PL}}$. Whereas A_1 is not a sentence letter of $\mathcal{L}^{\text{PL}}$, 'A_1' is a sentence letter of $\mathcal{L}^{\text{PL}}$. Although it is reasonably clear how to continue the list of partial lists indicated above, this definition still leaves something to be desired. Besides answering the immediate question of *which* sentence letters $\mathcal{L}^{\text{PL}}$ includes, there is also the methodological question of *how* to specify all of the sentence letters included in $\mathcal{L}^{\text{PL}}$ without relying on our readers ability to extend the partial lists indicated above. Even if the answer to the first question is clear enough, it is the answer to the second question that will have a number of further applications below. It is for this reason that we will improve on the third attempt however pedantic this may seem.

1.9.3 Corner Quotes

Continuing with our previous ambition to specify the sentence letters of $\mathcal{L}^{\text{PL}}$ in an accurate, explicit, and exhaustive way, consider the following somewhat long-winded alternative:

Attempt 4: The symbol to result from subscripting φ by x is a SENTENCE LETTER of $\mathcal{L}^{\text{PL}}$ whenever φ is a capital letter of the English alphabet and x is a numeral for a natural number.

This definition succeeds where the others failed. The only remaining issues that we may raise is just how cumbersome the construction 'The symbol to result from subscripting…' is, lacking the syntactic simplicity and generality that we might otherwise want.

Enter CORNER QUOTES, also called QUINE QUOTES on account of W.V. Quine. Instead of using the long-winded construction given above, we may stipulate the following:

SENTENCE LETTERS: $\ulcorner \varphi_x \urcorner$ for any capital English letter φ and natural numeral x.

Whereas standard quotation marks are used to name the string of symbols that they contain, corner quotes are used to refer to the complex symbol that results when the schematic variables are replaced with explicit values. One way to put the point is to say that $\ulcorner \varphi_x \urcorner$ is the result of concatenating φ with a subscript x. So long as 'φ' and 'x' are both schematic variables which have symbols as values, concatenating φ with a subscript x is also a symbol, e.g., 'A_1'. It is in this way that we may explicitly specify all sentence letters of $\mathcal{L}^{\text{PL}}$.

1.9.4 Expressions

Given the definitions above, we may now define the PRIMITIVE SYMBOLS of $\mathcal{L}^{\text{PL}}$ to include the operators, punctuation, and sentence letters for $\mathcal{L}^{\text{PL}}$ given above, and nothing besides. The EXPRESSIONS of $\mathcal{L}^{\text{PL}}$ may the be defined recursively as follows:

1. Every primitive symbol of $\mathcal{L}^{\text{PL}}$ is an expression of $\mathcal{L}^{\text{PL}}$.
2. For any expressions φ and ψ of $\mathcal{L}^{\text{PL}}$, $\ulcorner \varphi\psi \urcorner$ is an expression of $\mathcal{L}^{\text{PL}}$.
3. Nothing else is an expression of $\mathcal{L}^{\text{PL}}$.

In addition to taking the primitive symbols of $\mathcal{L}^{\text{PL}}$ to be expressions of $\mathcal{L}^{\text{PL}}$, the result of concatenating any two expressions of $\mathcal{L}^{\text{PL}}$ is an expression of $\mathcal{L}^{\text{PL}}$, and nothing else besides. This definition specifies the expressions of $\mathcal{L}^{\text{PL}}$ in an explicit and exhaustive way.

1.9.5 Well-Formed Sentences

Since any sequence of symbols is an expression, many expressions of $\mathcal{L}^{\text{PL}}$ will fail to be candidates for interpretation. That is, not only do they fail to mean something on a particular interpretation, there is no good way to interpret them at all. In particular, there is no good way to assign them truth-values. For example, consider the following expressions:

S1. $\neg\neg\neg\neg$
S2. B_3A_0
S3. $))\leftrightarrow$
S4. $A_4 \vee$

In order to interpret $\mathcal{L}^{\text{PL}}$, we need to say which expressions are candidates for interpretation, where we may expect the expressions above to be excluded. Put otherwise, we may ask which expressions of $\mathcal{L}^{\text{PL}}$ are grammatical, or as we will soon say, *well-formed sentences*. For ease of exposition, we will use the acronym 'wfs' throughout what follows where the plural is 'wfss'.

Sentence letters like 'A_1' and 'G_{13}' are certainly wfss. We can form further wfss out of these by using the various operators. Using negation, we can get '$\neg A_1$' and '$\neg G_{13}$'. Using conjunction, we can get '$A_1 \wedge G_{13}$', '$G_{13} \wedge A_1$', '$A_1 \wedge A_1$', and '$G_{13} \wedge G_{13}$'. We could also apply negation repeatedly to get '$\neg\neg A_1$' or apply negation along with conjunction to get wfss like '$\neg(A_1 \wedge G_{13})$' and '$\neg(G_{13} \wedge \neg G_{13})$'. The possible combinations are endless, even starting with just these two sentence letters rather than infinitely many sentence letters as above.

Although there is no point in trying to list all the wfss, we can define all WELL-FORMED SENTENCES OF $\mathcal{L}^{\text{PL}}$ by way of the following recursive clauses:

1. Every sentence letter of $\mathcal{L}^{\text{PL}}$ is a wfs of $\mathcal{L}^{\text{PL}}$.
2. For any expressions φ and ψ of $\mathcal{L}^{\text{PL}}$, if φ and ψ are wfss of $\mathcal{L}^{\text{PL}}$, then:
   (a) $\ulcorner\neg\varphi\urcorner$ is a wfs of $\mathcal{L}^{\text{PL}}$;
   (b) $\ulcorner(\varphi \wedge \psi)\urcorner$ is a wfs of $\mathcal{L}^{\text{PL}}$;
   (c) $\ulcorner(\varphi \vee \psi)\urcorner$ is a wfs of $\mathcal{L}^{\text{PL}}$;
   (d) $\ulcorner(\varphi \rightarrow \psi)\urcorner$ is a wfs of $\mathcal{L}^{\text{PL}}$; and
   (e) $\ulcorner(\varphi \leftrightarrow \psi)\urcorner$ is a wfs of $\mathcal{L}^{\text{PL}}$.
3. Nothing else is a wfs of $\mathcal{L}^{\text{PL}}$.

As in the definition of the expressions of $\mathcal{L}^{\text{PL}}$, the definition of the wfss of $\mathcal{L}^{\text{PL}}$ is recursive on account of calling on the wfss of $\mathcal{L}^{\text{PL}}$ in (2) in order to define further wfss of $\mathcal{L}^{\text{PL}}$. If you have not seen recursive definitions before, you might worry that this is all rather circular.

In order assuage any doubts that the recursive definition of the wfss of $\mathcal{L}^{\text{PL}}$ given above is in good standing it will help to think of building the set of all wfss of $\mathcal{L}^{\text{PL}}$ as follows. In stage 0, we add all the sentence letters, calling this set Λ_0. Then in stage 1, we take any wfss from Λ_0 and substitute them for φ and ψ in the composition rules given in condition (2) above, adding the results to a new set Λ_0'. We do this in all the ways that we can, adding as much to Λ_0' as possible while limiting ourselves to the ingredients included in Λ_0. Once we stop getting anything new, we may take $\Lambda_1 = \Lambda_0 \cup \Lambda_0'$ which contains all and only the wfss which belong to either Λ_0 or Λ_0'. We then repeat the process to build Λ_2 from Λ_1 in the same way. More generally, given any Λ_n we may build Λ_{n+1} by the same procedure. Finally we consider the union which gathers together the members from each Λ_n for all $n \in \mathbb{N}$.

$$\Lambda = \bigcup_{n \in \mathbb{N}} \Lambda_n.$$

A simpler way to describe the same thing is to take the set Λ of wfs of $\mathcal{L}^{\text{PL}}$ to be the smallest set— this corresponds to condition (3) above— to satisfy conditions (1) and (2) above. By imposing condition (3), we are making sure that nothing else ends up a wfs of $\mathcal{L}^{\text{PL}}$ like the number 2 or the Eiffel tower, thereby specifying a unique set of wfs of $\mathcal{L}^{\text{PL}}$.

Note that the definition of the wfss of $\mathcal{L}^{\text{PL}}$ is purely *syntactic*. Each composition rule specifies which expressions of $\mathcal{L}^{\text{PL}}$ are to be considered wfs of $\mathcal{L}^{\text{PL}}$. These rules will matter a great deal in the semantic and metalogical portions of this text. The resulting definition provides exactly what linguists have given up attempting to provide for English: a complete specification once and for all of which syntactic constructions are grammatical sentences— i.e., the wfss— of $\mathcal{L}^{\text{PL}}$. It is important to stress that the definition of the wfss $\mathcal{L}^{\text{PL}}$ does not tell us what the sentences of $\mathcal{L}^{\text{PL}}$ *mean* or what truth-values they have. To do so, we will provide a semantics for $\mathcal{L}^{\text{PL}}$ in Chapter 2. For now, our concern is limited to the rules for writing sequences of symbols in $\mathcal{L}^{\text{PL}}$ which count as well-formed sentences and nothing more.

1.9.6 Main Operator

It is worth computing whether an expression is a wfss of $\mathcal{L}^{\text{PL}}$ or not. For instance, suppose that we want to know whether or not '$\neg\neg\neg D$' is a wfs of $\mathcal{L}^{\text{PL}}$. Looking at the second clause of the definition, we know that '$\neg\neg\neg D$' is a wfs *if* '$\neg\neg D$' is a wfs. So now we need to ask whether or not '$\neg\neg D$' is a wfs. Again looking at the second clause of the definition, '$\neg\neg D$' is a wfs *if* '$\neg D$' is. Again, '$\neg D$' is a wfs *if* 'D' is a wfs. Now 'D' is a sentence letter, so we know that 'D' is a wfs by the first clause of the definition. Thus '$\neg\neg\neg D$' is in fact a wfs.

The operator that you look to first in decomposing a sentence is called the main operator of that sentence. For example, the main operator of '$\neg(E \vee (F \to G))$' is negation, and the main operator of '$(\neg E \vee (F \to G))$' is disjunction. Conversely, if you're building up a wfs from simpler sentences, the operator introduced by the last composition rule you apply is the main operator of the resulting sentence. It is the operator that governs the interpretation of the entire sentence. Being able to identify the main operator of sometimes convoluted sentences in $\mathcal{L}^{\text{PL}}$ is going to be an essential skill in your logical tool kit.

1.10 Metalinguistic Abbreviation

So far we have been careful to be precise. This is sort of precision is important when introducing a formal theory of syntax for the first time. However, maintaining this level of precision is both tedious and can often be hard to read, and so it is common to introduce certain simplifications as a convenient shorthand. For instance, we will often suppress the subscript '$_0$', writing 'A' and 'B' in place of 'A_0' and 'B_0', and so on for the other twenty-four sentence letters in which '$_0$' is the subscript. In this way, subscripts are rarely required, avoiding quite a bit of trouble by simply using different capital letters.

However simple, suppressing the subscript '$_0$' in sentence letters provides a paradigm case of the kinds of simplifications that we will introduce in this section. It is important to specify that these notational conventions do not change the official definitions. For instance, just because we have started writing 'A' instead of 'A_0' does not mean that 'A' is officially a wfss of $\mathcal{L}^{\text{PL}}$ and 'A_0' is not. Rather, we have simply provided a shorthand which can be done away with, recovering the official sentence letters of $\mathcal{L}^{\text{PL}}$. More generally, our notational conventions provide a convenient shorthand that eases the expression of the sentences of $\mathcal{L}^{\text{PL}}$ given a clear understanding of the official definitions. Accordingly, these notational conventions are called METALINGUISTIC ABBREVIATIONS since they do not happen in our object language, but rather in the metalanguage that we use for talking *about* elements of the object language in a natural and abbreviated way. It is nevertheless important to be clear about which conventions we will permit throughout what follows to avoid introducing confusion.

Before indulging in the simplifications indicated below, it is important to ensure that: (1) you understand the official definitions in both spirit and letter; (2) the simplifications that you employ have been specifically permitted in this text; and (3) you do not make any simplifications that lead to ambiguities that cannot be removed or else permitted without harm (more on this soon). For instance, the following section will explain how to drop parentheses. However, until you are confident about the use of parentheses in the definition of the wfss of $\mathcal{L}^{\text{PL}}$, it is good advice to stick to the official definition of the wfss of $\mathcal{L}^{\text{PL}}$. The notational conventions that we provide are a way to skip steps when writing things down, but they will not help you master the official definitions. If you're unsure about whether it's OK to take a certain shortcut, the safest thing is to continue to use the official definitions. With this health warning in place, we may proceed with caution.

1.10.1 Parentheses

Given the definition of the wfs of $\mathcal{L}^{\text{PL}}$, we are now in a position to observe that neither of the following are wfss of $\mathcal{L}^{\text{PL}}$, and for good reason:

T1. $Q \wedge R$.
T2. $C \vee D \wedge E$.

The reason that T1 is not a wfs is the boring but crucial reason that it lacks outermost parentheses. Officially, a wfs like '$Q \wedge R$' must have outermost parentheses because we might want to use this sentence to construct further, more complicated sentences which have this sentence as a part. For instance, if we negate '$(Q \wedge R)$', we get '$\neg(Q \wedge R)$'. If we just had '$Q \wedge R$' without the parentheses and put a negation in front of it, we would have '$\neg Q \wedge R$' which it would be hard to distinguish from '$(\neg Q \wedge R)$', something very different than '$\neg(Q \wedge R)$'. The sentence '$\neg(Q \wedge R)$' means that it is not the case that both 'Q' and 'R' are true; 'Q' might be false or 'R' might be false, but the sentence does not tell us which. The sentence '$(\neg Q \wedge R)$' means specifically that 'Q' is false and that 'R' is true. This highlights the critical role that parentheses will play when we go on to interpret the wfss of $\mathcal{L}^{\text{PL}}$.

Officially, a conjunction is a sentence of the form $\ulcorner(\varphi \wedge \psi)\urcorner$ for any sentences φ and ψ, not a sentence of the form $\ulcorner\varphi \wedge \psi\urcorner$. Nevertheless, dropping the outermost parentheses is both permissible and common when it does not lead to any ambiguities, i.e., when there is a unique wfs $\mathcal{L}^{\text{PL}}$ which is easy to recover by adding an outermost pair, or else when the ambiguity that results does not make any substantial difference (more on this in the next chapter).

For example, we may recover the following wfs from T1 without any ambiguity whatsoever:

U1. $(Q \wedge R)$.

The same cannot be said for T2. Rather, we must choose between the following candidates:

V1. $((C \vee D) \wedge E)$.
V2. $(C \vee (D \wedge E))$.

Apart from anything to do with their meaning, the two wfss above are different. Whereas neither T1 nor T2 are officially wfss of $\mathcal{L}^{\text{PL}}$, both V1 and V2 are wfss of $\mathcal{L}^{\text{PL}}$ in the most official sense. Of course, we could have defined the wfss not to include outermost parentheses, so what's the reason for this stipulation? As it will turn out, V1 and V2 will have different truth-conditions, providing good reason to distinguish between them syntactically.

Without including any parentheses at all, our syntax would fail to keep track of the order in which a sentence has been constructed, and as a result, would run together sentences like V1 and V2 above, turning both of them back into the sentence T2. Although it will be convenient to sometimes drop parentheses to make things easier for ourselves, it is important to do this carefully so that we do not lose important information that we need in order to get back to the original form of the sentence. We will do this in several ways.

First, we understand that '$Q \wedge R$' is short for '$(Q \wedge R)$'. As a matter of convention, we can leave off parentheses that occur *around the entire sentence*. Even though we don't always write out the outermost parentheses, we know that they really should be there. It is important to stress that this is only possible when '$Q \wedge R$' occurs by itself, and not as a part of some more complex sentence. If we were able to drop parentheses even when some sentence occurs as a part of a bigger sentence, we would be able to turn both V1 and V2 into T2, and that is not what we want because then there would be no way to determine the main operator.

Second, it can sometimes be confusing to look at long sentences with nested sets of parentheses. We adopt the convention of using square brackets '[' and ']' in place of parenthesis. There is no logical difference between '$(P \vee Q)$' and '$[P \vee Q]$', for example. The unwieldy sentence '$(((H \rightarrow I) \vee (I \rightarrow H)) \wedge (J \vee K))$' could be written in the following way, omitting the outermost parentheses and using square brackets to make the structure of the sentence clear:

$$\big[(H \rightarrow I) \vee (I \rightarrow H)\big] \wedge (J \vee K).$$

Note that *Carnap* will always add the outer parentheses which can make things a bit harder to parse. So be careful when using that system.

Third, we will sometimes want to regiment the conjunction or disjunction of three or more sentences. For instance, consider the following sentence and symbolization key:

W1. Alice, Bob, and Candice all went to the party.
W2. Either Alice, Bob, and Candice went to the party.

A: Alice went to the party.
B: Bob went to the party.
C: Candice went to the party.

The composition rules only allow us to form a conjunction out of two sentences, so we can regiment W1 as either '$(A \wedge B) \wedge C$' or '$A \wedge (B \wedge C)$'. However, there is no reason to distinguish between these regimentation since the two are logically equivalent, or put otherwise, the two are true in exactly the same interpretations. So we might as well just write '$A \wedge B \wedge C$'. As a matter of convention, we can leave out parentheses when we conjoin three or more sentences.

A similar situation arises in disjoining multiple sentences. For instance, W2 can be regimented as either '$(A \vee B) \vee C$' or '$A \vee (B \vee C)$'. Since these two regimentations are logically equivalent, we may write '$A \vee B \vee C$'. These two conventions only apply to multiple conjunctions or multiple disjunctions, not any combination of conjunctions and disjunctions. If a series of operators includes both disjunctions and conjunctions, then the parentheses are essential, as with '$(A \wedge B) \vee C$' and '$A \wedge (B \vee C)$'. The parentheses are also required if there is a series of conditionals or biconditionals, as with '$(A \rightarrow B) \rightarrow C$' and '$A \leftrightarrow (B \leftrightarrow C)$'.

If we had given a different definition of the wfss of $\mathcal{L}^{\text{PL}}$, strings of conjunctions or disjunctions could have counted as wfss of $\mathcal{L}^{\text{PL}}$, obviating the need for the conventions indicated above. For instance, we might have permitted $\ulcorner(\varphi \wedge \ldots \wedge \psi)\urcorner$ to be a wfs of $\mathcal{L}^{\text{PL}}$ whenever $\varphi, \ldots, \psi$ are all wfss of $\mathcal{L}^{\text{PL}}$. This would have made it a little easier to regiment some English sentences, but it would have also come at the cost of making our the language $\mathcal{L}^{\text{PL}}$ much more complicated. In particular, we would have to keep this more complex definition in mind when we provide a semantics and proof system for $\mathcal{L}^{\text{PL}}$. Rather, we want $\mathcal{L}^{\text{PL}}$ to be as simple as possible without reducing its expressive power. Since no expressive power is added by permitted $\ulcorner(\varphi \wedge \ldots \wedge \psi)\urcorner$ to be a wfs of $\mathcal{L}^{\text{PL}}$, there is little reason to indulge in this complication, keeping the semantics and proof theory as simple as possible. Nevertheless, insisting that conjunctions and disjunctions have exactly two conjuncts or disjuncts on the official definition does raise certain syntactic redundancies that are not motivated by corresponding semantic differences, e.g., between the equivalent regimentations of W1 and W2. Adopting notational conventions is a compromise between the competing desires to avoid complicating the semantics and proof theory of $\mathcal{L}^{\text{PL}}$ on the one hand and to avoid needless syntactic redundancies on the other.

Strictly speaking, '$A \vee B \vee C$' and '$A \wedge B \wedge C$' are not sentences of $\mathcal{L}^{\text{PL}}$. We write them this way for the sake of convenience, but really these sentences are ambiguous. The only reason we can get away with this is that the ambiguities do not amount to any logical differences since any way of adding parentheses will amount to sentences with the same truth-conditions.

1.10.2 Dropping Quotes

We have taken subscripted uppercase letters in English to be sentence letters of $\mathcal{L}^{\text{PL}}$:

$$A, B, C, Z, A_1, B_4, A_{25}, J_{375}, \ldots$$

Although we could have added quotes around each of the letters above, it is clear in this context that we mean to be mentioning and not using these sentences. After all, this text is written in the metalanguage of mathematical English and the sentences above are sentences of the object language $\mathcal{L}^{\text{PL}}$ that we intend to be discussing. Consider the following examples:

X1. D is a sentence letter of $\mathcal{L}^{\text{PL}}$.
X2. 'D' is a sentence letter of $\mathcal{L}^{\text{PL}}$.

Since 'D' is a sentence of $\mathcal{L}^{\text{PL}}$ and not our metalanguage, X1 is nonsense. Put flatly, the sentences of $\mathcal{L}^{\text{PL}}$ do not belong to our metalanguage mathematical English. Rather, it is the canonical names for the sentences of $\mathcal{L}^{\text{PL}}$ which belong to our metalanguage, and for this we must use quotes as before. Thus whereas X1 is gibberish, X2 is true.

We may then compare:

Y1. Schnee ist weiß is a German sentence.
Y2. 'Schnee ist weiß' is a German sentence.

Whereas Y1 is again gibberish belonging to no single natural language, Y2 is a perfectly intelligible sentence of English which happens to mention a sentence of German.

Although quotes are officially required for Y2 to be true, it is nevertheless pretty clear what this sentence intends. Matters are even clearer in the case of X1 and X2 above. Accordingly, it is permissible to drop the quotes in X2, using X1 in its place for ease. Since we will be talking about the sentences of $\mathcal{L}^{\text{PL}}$ a lot and the quotes can get cumbersome, we will often drop the quotes, relying on context and the reader's competence to know where they belong.

In addition to a convenience that we will indulge in throughout this course, dropping quotes is common practice throughout logic, and so it is good to get some practice with these conventions. This applies as much to corner quotes as it does to standard quotes. For instance, in defining the wfss of a language, it is common practice to do so *without* using corner quotes by relying on the reader's competence to know where they belong. Since this is an introductory text, the definitions above have been made explicit. Nevertheless, when we go on to setup further languages such as $\mathcal{L}^{\text{FOL}}$ below, we will have occasion to indulge in certain simplifications, leaving off the corner quotes.

In making such simplifications, what matters is that the conveniences that we indulge do not lead to any real ambiguities. Additionally, we should maintain quotes when clarity is improved. After all, the goal is to make things to read, not harder, so bear this in mind if you choose to drop quotes. Certainly in some contexts using quotes is invaluable.

1.10.3 Conventions for Arguments

One of the main purposes of using $\mathcal{L}^{\text{PL}}$ is to study arguments. In English, the premises of an argument are often expressed by individual sentences, and the conclusion by a further sentence. Since we can regiment English sentences in $\mathcal{L}^{\text{PL}}$, we can also regiment English arguments using $\mathcal{L}^{\text{PL}}$ by regimenting each of the sentences used in an English argument. Sometimes English arguments run all the sentences together where clarity is helped by dividing things up when we go about giving our regimentation. Even so, $\mathcal{L}^{\text{PL}}$ itself has no way to flag some sentence as the conclusion and the other sentences as the premises. By contrast, English uses words like 'so', 'therefore', etc., to mark which sentence is the conclusion.

In Chapter 0, we specified that we will arrange arguments so that the conclusion is the final sentence, separating the premises with a horizontal line. Although we will maintain this convention below, it is worth emphasizing that $\mathcal{L}^{\text{PL}}$ does not include any such horizontal lines. Rather, these presentational details are taking place in the metalanguage in order to help us specify which sentences are the premises and conclusion. Nevertheless, this convention has the downside that it makes use of vertical space, and so cannot be present in line. Thus it will occasionally be helpful to use the symbol '$\therefore$' in order to indicate that what follows is the conclusion. For instance, suppose we want to regiment an argument with the premises and conclusion with ψ. We may write '$\varphi_1, \ldots, \varphi_n \therefore \psi$' without breaking the paragraph.

Since the symbol '$\therefore$' belongs to the metalanguage not $\mathcal{L}^{\text{PL}}$, one might think that we would need to put quotes around the $\mathcal{L}^{\text{PL}}$-sentences which flank it, referring to the result as an argument with $\mathcal{L}^{\text{PL}}$ sentences. That is a sensible thought, but adding these quotes would only make things more cumbersome and rather than easier to read. While we are stipulating conventions, we can simply stipulate that quotes around arguments are unnecessary. For instance, we may say that $A, A \rightarrow B \therefore B$ is an argument in $\mathcal{L}^{\text{PL}}$ whose premises are A and $A \rightarrow B$ and whose conclusion is B, leaving the quotes off for ease.

Chapter 2

Logical Consequence

Whereas the previous chapter introduced truth tables, this chapter will present the truth table method which provides a decidable procedure for evaluating the validity of arguments in $\mathcal{L}^{\text{PL}}$. Given the (albeit limited) expressive power of $\mathcal{L}^{\text{PL}}$, this amounts to a mechanical way of evaluating the natural language arguments that admit of reasonably faithful regimentations in $\mathcal{L}^{\text{PL}}$. After reviewing both the advantages and disadvantages of this procedure, the second half of the chapter will present a more versatile alternative.

2.1 Truth-Functional Operators

Any wfs of $\mathcal{L}^{\text{PL}}$ that is not a sentence letter is composed of sentence letters together with the sentential operators. In Chapter 1, we offered truth tables for each operator. The fact that it is possible to give truth tables like this is very significant. It means that our operators are TRUTH-FUNCTIONAL. That is to say, the only thing that matters for determining the truth-value of a given wfs of $\mathcal{L}^{\text{PL}}$ is the truth-values of its constituent. For instance, to determine the truth-value of a sentence $\neg A$, the only thing that matters is the truth-value of A. Given any interpretation, the truth-value of a negation on that interpretation is a *function* of the truth-value of its negand on that interpretation, and likewise for the other operators.

We are using the same notion of a function that you have probably encountered in mathematics. First we may define the CARTESIAN PRODUCT $X \times Y$ of the sets X and Y to be the set of all ordered pairs $\langle x, y\rangle$ where $x \in X$ and $y \in Y$ which we may write in set-builder notation as $X \times Y \coloneqq \{\langle x, y\rangle : x \in X,\ y \in Y\}$.[1] A RELATION from X to Y is any subset $A \subseteq X \times Y$. A FUNCTION $f : D \to R$ from the DOMAIN D to the RANGE R is any relation $f \subseteq D \times R$ from D to R where $f(x) = f(y)$ for any $x, y \in D$ where $x = y$. Intuitively, a function from one set to another associates each member of the first set (the domain) with exactly one member of the second set (the range). Once the first element is fixed, the function uniquely selects an

[1] The '$\coloneqq$' symbol signifies that a definition is being provided.

element of the second set. Instead of always writing $f(x, y)$, it makes sense to write $f(x) = y$ for the unique element in the range y to which the element x in the domain is mapped. For instance, given any numerical value of x, we may unambiguously determine the value of x^2, and so $f(x) = x^2$ is a function. In the same way, the truth-value of A on an interpretation will unambiguously determine the truth-value of $\neg A$ on that interpretation.

Truth-functionality is not inevitable. The syntax of English, for example, permits one to make a new declarative English sentence by prefixing the phrase 'Ted Cruz doesn't care whether' in front of any declarative English sentence. In this respect, 'Ted Cruz doesn't care whether' is syntactically similar to '$\neg$' in $\mathcal{L}^{\text{PL}}$: it is a sentential operator, producing new sentences from old. Nevertheless, it is impossible to give a truth-functional characterization of the operator 'Ted Cruz doesn't care whether' in English that respects its intuitive meaning in English. If you want to know whether Ted Cruz cares about fixing Texas's electrical grid, it's not enough to know whether anyone is fixing Texas's electrical grid. If it is being fixed, he might care or he might not. If it is not be fixed, again he might care or might not. Thus 'Ted Cruz doesn't care whether' is not truth-functional: it operates on more than just the truth-value of its argument. By contrast, the sentential operators included in $\mathcal{L}^{\text{PL}}$ are truth-functional, and it is for this reason that we are able to construct truth tables.

2.2 Complete Truth Tables

A truth table for a sentence may be constructed by writing the sentence in question at the top right of a table, and each of the distinct sentence letters immediately to the left on the top row. We then add 2^n rows below the top row where n is the number of distinct sentence letters. For instance, if there are only two sentence letters, we will need four rows of truth-values. Beginning with the sentence letter furthest to the left, we fill out the column with $2^{(n-1)}$ copies of 1 followed by $2^{(n-1)}$ copies of 0. Moving to the next sentence letter, we fill out the column with $2^{(n-2)}$ copies of 1 followed by $2^{(n-2)}$ copies of 0. We then proceed to the next sentence letter (if there is one), following the same pattern as before but with $2^{(n-3)}$ copies of 1 and 0, respectively. Continue this process until all sentence letters in the table have truth-values below them. This completes the truth table setup.

In order to assign truth-values to complex sentences, consider the following CHARACTERISTIC TRUTH TABLES where instead of particular sentence letters, we will use schematic variables:

φ	$\neg\varphi$
1	0
0	1

φ	ψ	$\varphi \wedge \psi$	$\varphi \vee \psi$	$\varphi \to \psi$	$\varphi \leftrightarrow \psi$
1	1	1	1	1	1
1	0	0	1	0	0
0	1	0	1	1	0
0	0	0	0	1	1

Table 2.1: The characteristic truth tables for the operators of $\mathcal{L}^{\text{PL}}$.

The table above provides a general recipe for calculating truth-values for any sentences φ and ψ, however complex. The characteristic truth table for conjunction, for example, gives the truth conditions for any sentence of the form $\varphi \wedge \psi$. Even if the conjuncts φ and ψ are complex sentences, the conjunction is true if and only if both φ and ψ are true.

Let's construct a truth table for the complex sentence $(H \wedge I) \rightarrow H$. We consider all possible combinations of 1 and 0 for H and I, which gives us four rows. We then copy the truth-values for the sentence letters and write them underneath the letters in the sentence.

H	I	$(H$	$\wedge$	$I)$	$\rightarrow$	H
1	1	**1**		**1**		**1**
1	0	**1**		**0**		**1**
0	1	**0**		**1**		**0**
0	0	**0**		**0**		**0**

So far, all we have done is duplicate the first two columns. We have written the H column twice— once under each H— and the I column once under the I.

Now consider the subsentence $H \wedge I$ which is a conjunction. Since H and I are both true on the first row and a conjunction is true when both conjuncts are true, we write a 1 underneath the conjunction symbol. In the other three rows, at least one of the conjuncts is false, so the conjunction $H \wedge I$ is false. So we write 0s under the conjunction symbol on those rows:

H	I	$(H$	$\wedge$	$I)$	$\rightarrow$	H
1	1	1	**1**	1		1
1	0	1	**0**	0		1
0	1	0	**0**	1		0
0	0	0	**0**	0		0

The entire sentence $(H \wedge I) \rightarrow H$ is a conditional. On the second row, for example, $H \wedge I$ is false and H is true. Since a conditional is true when the antecedent is false, we write a 1 in the second row underneath the conditional symbol. Using the truth table for the material conditional where $H \wedge I$ is the antecedent and H as the consequent, we may derive the following values for the material conditional claim as a whole:

H	I	$(H$	$\wedge$	$I)$	$\rightarrow$	H
1	1		1		**1**	1
1	0		0		**1**	1
0	1		0		**1**	0
0	0		0		**1**	0

In computing the value for the conditional (the column under $\rightarrow$), it is only important to look at the values for its antecedent (the column under $\wedge$) and the value of its consequent (the column under H on the right). The column of 1s underneath the conditional tells us that the sentence $(H \wedge I) \rightarrow H$ is true regardless of the truth-values of H and I. They can be true or false in any combination, and the compound sentence still comes out true.

It is crucial that we have considered all of the possible combinations. If we only had a two-line truth table, we could not be sure that the sentence was not false for some other combination of truth-values. Since each row of the truth table represents a different way of interpreting the relevant sentence letters H and I, each row corresponds to a distinct interpretation of the sentence letters in question. Moreover, every possible combination of truth-values for H and I have been included. Since the truth-values of all other sentence letters do not effect the truth-value of the sentence in question, we may conclude that $(H \wedge I) \rightarrow H$ is *true in all interpretations* whatsoever. In other words, $(H \wedge I) \rightarrow H$ is a *tautology.*

In this example, we have not repeated all of the entries in every successive table, so that it's easier for you to see which parts are new. When actually writing truth tables on paper, however, it is impractical to erase whole columns or rewrite the whole table for every step. Although it is more crowded, the truth table can be written in this way:

H	I	$(H$	$\wedge$	$I)$	$\rightarrow$	H
1	1	1	1	1	**1**	1
1	0	1	0	0	**1**	1
0	1	0	0	1	**1**	0
0	0	0	0	0	**1**	0

Most of the columns underneath the sentence are only there for bookkeeping purposes. When you become more adept with truth tables, you will probably no longer need to copy over the columns for each of the sentence letters. In any case, the truth-values for the original sentence is given by the column underneath the main logical operator of the sentence which in this case is the column underneath the conditional, marked in **bold** for clarity.

A COMPLETE TRUTH TABLE has a row for all possible combinations of 1 and 0 for the sentence letters and the characteristic truth tables have been used to write truth-values below all the operators. The size of the complete truth table depends on the number of different sentence letters in the table. A sentence that contains only one sentence letter requires only two rows, as in the characteristic truth table for negation. This is true even if the same letter is repeated many times, as in the sentence $[(C \leftrightarrow C) \rightarrow C] \wedge \neg(C \rightarrow C)$. The complete truth table requires only two lines because there are only two possible interpretations of C: either it is true or it is false. A single sentence letter can never be marked both 1 and 0 on the same row. The truth table for this sentence looks like this:

C	$[($	C	$\leftrightarrow$	C	$)$	$\rightarrow$	C	$]$	$\wedge$	$\neg$	$($	C	$\rightarrow$	C	$)$
1		1	1	1		1	1		**0**	0		1	1	1	
0		0	1	0		0	0		**0**	0		0	1	0	

Looking at the column underneath the main operator, we see that the sentence is false on both rows of the table, and so it is false regardless of whether C is true or false. Since the rows of the truth table correspond to the different possible interpretations of the relevant sentence letters, the sentence above is false on all interpretations, and so it is a *contradiction*.

A sentence that contains two sentence letters requires 2^2 lines for a complete truth table as in the other characteristic truth tables and the table for $(H \wedge I) \rightarrow I$ above. A sentence that contains three sentence letters requires 2^3 lines. For example:

M	N	P	M	$\wedge$	$(N$	$\vee$	$P)$
1	1	1	1	**1**	1	1	1
1	1	0	1	**1**	1	1	0
1	0	1	1	**1**	0	1	1
1	0	0	1	**0**	0	0	0
0	1	1	0	**0**	1	1	1
0	1	0	0	**0**	1	1	0
0	0	1	0	**0**	0	1	1
0	0	0	0	**0**	0	0	0

This table shows that $M \wedge (N \vee P)$ is true on some interpretations and false on others depending on the truth-values of M, N, and P, and so it is *contingent*.

A complete truth table for a sentence that contains four different sentence letters requires 2^4 lines. This means that the truth table method becomes syntactically unmanageable very quickly. This is a significant limitation of this method.

2.3 Truth Table Definitions

Recall from §0.5.1 that an English sentence was said to be a tautology just in case it is true on all interpretations, a contradiction just in case it is false on all interpretations, and logically contingent just in case it is true on some interpretations and false on others. Similarly, an English sentence logically entails another just in case the latter is true on any interpretation on which the former is true, and two English sentences are logically equivalent just in case they logically entail each other and so are true on exactly the same interpretations. Even in restricting consideration to truth-values in interpreting the sentences of English, there is no well-defined set of English sentences, and so no corresponding definition of an interpretation as an assignment of every English sentence to a unique truth-value. It is for this reason that we introduced $\mathcal{L}^{\text{PL}}$ in Chapter 1, carefully defining the wfss of $\mathcal{L}^{\text{PL}}$.

Given the definition of the wfs of $\mathcal{L}^{\text{PL}}$ together with the characteristic truth tables for the sentential operators of $\mathcal{L}^{\text{PL}}$, we are now in a position to define the interpretations of $\mathcal{L}^{\text{PL}}$ in a way that we could not do so for English. In particular, an INTERPRETATION of $\mathcal{L}^{\text{PL}}$ is any function from the set of wfs of $\mathcal{L}^{\text{PL}}$ (the domain) to the set of truth-values $\{0, 1\}$ (the range) which satisfies the characteristic truth tables given above. For instance, any interpretation that assigns A to 0 will assign $\neg A$ to 1. More generally, an interpretation assigns $\neg\varphi$ to 1 just in case it assigns φ to 0 since this is what is required to satisfy the characteristic truth table for negation. Similarly, an interpretation assigns $\varphi \wedge \psi$ to 1 just in case it assigns both φ and ψ to 1 as specified by the characteristic truth table for conjunction. Given any wfs of $\mathcal{L}^{\text{PL}}$, the characteristic truth table for the main operator of that sentence determines the truth-value of that sentence as a function of the truth-value(s) for its arguments.

Rather than worrying about the truth-value of all wfs of $\mathcal{L}^{\text{PL}}$ whatsoever, constructing a complete truth tables provides a way to interpret a sentence while limiting consideration to its parts. This approach depends on the truth-functionality of the sentential operators. We may then put this method to work to define a TAUTOLOGY of $\mathcal{L}^{\text{PL}}$ to be any sentence of $\mathcal{L}^{\text{PL}}$ whose truth table only has 1s under its main operator. Accordingly, a tautology is a sentence of $\mathcal{L}^{\text{PL}}$ whose truth does not depend on the particular truth-values of the sentence letters from which it was constructed, but rather follows from the *logical form* of that sentence. Similarly, a sentence is a CONTRADICTION in $\mathcal{L}^{\text{PL}}$ just in case the column under its main operator is 0 on every row of its complete truth table. Instead of being true in virtue of its logical form, a contradiction is false in virtue of its logical form. A sentence is LOGICALLY CONTINGENT in $\mathcal{L}^{\text{PL}}$ just in case the column under its main operator includes both 1s and 0s.

2.3.1 Logical Entailment and Equivalence

A sentence in English was said to logically entail another just in case the latter is true on any interpretation on which the former is true. Two sentences were then said to be logically equivalent in English just in case they logically entail each other, and so have the same truth-value on all interpretations. We can now say with greater precision that a sentence of $\mathcal{L}^{\text{PL}}$ LOGICALLY ENTAILS another just in case the complete truth table for these two sentences is such that on any row, the latter has a 1 under its main operator whenever the former has a 1 on that row under its main operator. Consider the following example:

A	B	$\neg$	A	$\neg$	$(A$	$\wedge$	$B)$
1	1	**0**	1	**0**	1	1	1
1	0	**0**	1	**1**	1	0	0
0	1	**1**	0	**1**	0	0	1
0	0	**1**	0	**1**	0	0	0

Whereas before we only considered truth tables for one wfs at a time, the truth table above interprets two wfs of $\mathcal{L}^{\text{PL}}$ at once. By including a column for all sentence letters contained in either wfs, the complete truth table exhausts all possible combinations of truth-values for the sentence letters upon which the truth of the wfss depend. We may the observe that every row in which $\neg A$ has a 1 under its main operator is also a row in which $\neg(A \wedge B)$ has a 1 under its main operator, and so $\neg A$ logically entails $\neg(A \wedge B)$. By contrast, given the second row of truth-values, $\neg(A \wedge B)$ does not logically entail $\neg A$.

In the special case where two wfs of $\mathcal{L}^{\text{PL}}$ logically entail each other, we may say that those sentences are LOGICALLY EQUIVALENT. It follows that any two wfs of $\mathcal{L}^{\text{PL}}$ are logically equivalent just in case the truth-values under their main operators are the same on every row of their complete truth table. For instance, consider the sentences $\neg(A \vee B)$ and $\neg A \wedge \neg B$. In order to find out if they are logically equivalent we may construct their complete truth table:

A	B	$\neg$	$(A$	$\vee$	$B)$	$\neg$	A	$\wedge$	$\neg$	B
1	1	**0**	1	1	1	0	1	**0**	0	1
1	0	**0**	1	1	0	0	1	**0**	1	0
0	1	**0**	0	1	1	1	0	**0**	0	1
0	0	**1**	0	0	0	1	0	**1**	1	0

The columns under the main operators for the two wfs on the right are identical. Since the rows of a complete truth table exhaust the different interpretations of the relevant sentence letters, this amounts to requiring the two wfs of $\mathcal{L}^{\text{PL}}$ to have the same truth-value on every interpretation. It is for this reason that the two wfs of $\mathcal{L}^{\text{PL}}$ are logically equivalent.

2.3.2 Satisfiability

In the previous example, a truth table was constructed for two wfss of $\mathcal{L}^{\text{PL}}$ at once. More generally, we may take a COMPLETE TRUTH TABLE for a set Γ of wfss of $\mathcal{L}^{\text{PL}}$ to be the result of listing each wfs in Γ side-by-side on the top right of a table and then completing the table for each wfs in a similar manner to what was described above.

Recall that a set of sentences in English is satisfiable just in case there is an interpretation which makes them all true. Analogously, we may wish to say that a set Γ of wfss in $\mathcal{L}^{\text{PL}}$ is SATISFIABLE just in case there is a row of a complete truth table including every wfs in the set where the main operator under every sentence is 1, and UNSATISFIABLE otherwise. For instance, look again at the truth table above. We see that $\{\neg(A \vee B), \neg A \wedge \neg B\}$ is satisfiable, because there is at least one row where both sentences have 1 under their main operators.

To take another example, we may ask if the set $\{A, \neg(A \vee \neg B)\}$ is satisfiable. Here the answer is 'No' as the following truth table shows:

A	B	$\neg$	$(A$	$\vee$	$\neg$	$B)$
1	1	**0**	1	1	0	1
1	0	**0**	1	1	1	0
0	1	**1**	0	0	0	1
0	0	**0**	0	1	1	0

Since there is no row in which both of the sentences in the set $\{A, \neg(A \vee \neg B)\}$ have a 1 under their main operator, we may conclude that the set is unsatisfiable.

2.3.3 Logical Consequence and Validity

Whereas Chapter 0 provided an intuitive definition of logical consequence for the sentences of English, we may now appeal to complete truth tables in order to provide a correlate for $\mathcal{L}^{\text{PL}}$. In particular, a wfs φ of $\mathcal{L}^{\text{PL}}$ is a LOGICAL CONSEQUENCE of a set of wfss Γ of $\mathcal{L}^{\text{PL}}$ just in case every row of a complete truth table for the wfs in Γ together with φ is such that φ has a 1 under its main operator whenever every sentence in Γ has a 1 under its main operator. We may then say that an argument in $\mathcal{L}^{\text{PL}}$ with premises Γ and conclusion φ is VALID just in case the conclusion is a logical consequence of the premises.

Instead of saying that an English argument is valid by appealing to the interpretations of English, we may say that an argument in English has a $\mathcal{L}^{\text{PL}}$ regimentation that is valid. Regimenting an English argument in $\mathcal{L}^{\text{PL}}$ and constructing its complete truth table provides a way to identify the logical features which explain why the argument in English is valid. Although there is a precise mathematical definition of validity in $\mathcal{L}^{\text{PL}}$, the same cannot be said for what does and does not count as a faithful regimentation of the argument. Rather, this much remains intuitive, relying on and individual's best judgment. In many cases, there may not be a uniquely best regimentation, or indeed any good regimentation at all.

Given any argument in $\mathcal{L}^{\text{PL}}$ whose premises are unsatisfiable, it follows that the argument is *vacuously valid* since there is no row in a complete truth table for the premises with a 1 under the main operator of every premise, and so vacuously, every row in a complete truth table in which there is a 1 under the main operator of every premises also has a 1 under the main operator for the conclusion. For instance, consider the following argument:

A1. $\neg A \wedge B$
A2. $\underline{\neg B \quad\;}$
A3. B

Given the truth table method presented above, it is straightforward to show that the argument is indeed valid. In particular, consider the following complete truth table:

A	B	$\neg$	A	$\wedge$	B	$\neg$	B	B
1	1	0	1	**0**	1	**0**	1	**1**
1	0	0	1	**0**	0	**1**	0	**0**
0	1	1	0	**1**	1	**0**	1	**1**
0	0	1	0	**0**	0	**1**	0	**0**

It is easy to see that there is no row in which the premises all have 1 under their main operators, and so trivially, there is no rows in which the premises all have a 1 under their main operators and the conclusion has a 0 under its main operator. Put otherwise, every row in which the premises all have 1 under their main operators— all zero of them— is such that the conclusion has a 1 under its main operator. Thus the argument is valid.

It is worth contrasting the example above with the following argument:

B1. $\neg L \rightarrow (J \vee L)$
B2. $\underline{\neg L \quad}$
B3. J

It is good practice to consult your intuitions about whether an argument is valid before beginning to complete its truth table. For instance, without looking any further, consider whether the argument above is valid. Once you have a guess we may construct the following:

J	L	$\neg$	L	$\rightarrow$	$(J$	$\vee$	$L)$	$\neg$	L	J
1	1	0	1	**1**	1	1	1	**0**	1	**1**
1	0	1	0	**1**	1	1	0	**1**	0	**1**
0	1	0	1	**1**	0	1	1	**0**	1	**0**
0	0	1	0	**0**	0	0	0	**1**	0	**0**

To determine whether the argument is valid, check to see whether there are any rows on which both premises have a 1 under their main operators, but the conclusion has a 0 under its main operator. Notice that unlike the previous argument, there is a row in which both premises have a 1 under their main operators. Nevertheless, the only row in which both the premises have a 1 under their main operators is the second row, and in that row the conclusion also has a 1 under its main operator. It follows that the argument is valid in $\mathcal{L}^{\text{PL}}$, though it is not vacuously valid as before. Rather, vacuous validity is a special case.

Here is another example. Is the following argument valid? Try to check intuitively first.

C1. $P \rightarrow Q$
C2. $\underline{\neg P \quad}$
C3. $\neg Q$

P	Q	P	$\rightarrow$	Q	$\neg$	P	$\neg$	Q
1	1	1	**1**	1	**0**	1	**0**	1
1	0	1	**0**	0	**0**	1	**1**	0
0	1	0	**1**	1	**1**	0	**0**	1
0	0	0	**1**	0	**1**	0	**1**	0

On the third row, each premise has a 1 under its main operator but the conclusion does not, and so the argument is invalid: the conclusion is not a logical consequence of its premises.

2.4 Decidability

Evaluating wfss in $\mathcal{L}^{\text{PL}}$ by constructing complete truth tables provides a simple mechanical procedure that is straightforward to systematically employ. Moreover, since every wfs of $\mathcal{L}^{\text{PL}}$ is of finite length and contains a finite number of sentence letters, constructing a complete truth table for a wfs of $\mathcal{L}^{\text{PL}}$ provides a finite procedure which determines whether that sentence is a tautology, contradiction, or logical contingent. Put otherwise, constructing a complete truth table for a wfs of $\mathcal{L}^{\text{PL}}$ provides an EFFECTIVE PROCEDURE for determining whether that wfs is a tautology or not, and similarly for the other logical properties that sentence may or may not have. Since there is an effective procedure for determining whether a wfs of $\mathcal{L}^{\text{PL}}$ is a tautology, we may say that it is DECIDABLE whether a wfs of $\mathcal{L}^{\text{PL}}$ is a tautology. It is similarly decidable whether a wfs of $\mathcal{L}^{\text{PL}}$ is a contradiction or logically contingent.

Given a finite set Γ of wfss of $\mathcal{L}^{\text{PL}}$, constructing a complete truth table for those wfss provides an effective procedure for determining whether Γ is satisfiable or not, and so the question of whether Γ is satisfiable is decidable. However, the same cannot be said for infinite sets of wfss of $\mathcal{L}^{\text{PL}}$. Even though each wfs of $\mathcal{L}^{\text{PL}}$ is of finite length with finitely many sentence letters, an infinite set of wfss of $\mathcal{L}^{\text{PL}}$ may contain infinitely many sentence letters, requiring an infinitely large truth table. Although there is nothing to prevent us from *defining* infinitely large truth tables mathematically, there is of course little hope of *using* such an infinite truth table to determine whether an infinite set of sentences is satisfiable. It is for this reason that the truth table method does not provide an effective procedure for deciding whether a set of wfs of $\mathcal{L}^{\text{PL}}$ is satisfiable. Of course, there could be another effective procedure for determining whether a set of $\mathcal{L}^{\text{PL}}$ sentences is satisfiable, and so we cannot claim that it is UNDECIDABLE whether a set of sentences is satisfiable just because one method cannot be used.

Although it is beyond the scope of this course, it is in fact undecidable whether an infinite set of wfs of $\mathcal{L}^{\text{PL}}$ is satisfiable or not. That is, there is no effective procedure that we could hope to use to determine whether any infinite set of wfs of $\mathcal{L}^{\text{PL}}$ is satisfiable. However, even in restricting consideration to finite sets, we may observe that it is often infeasible to construct truth tables for a wfs or set of wfss of $\mathcal{L}^{\text{PL}}$ which include too many sentence letters. For instance, a wfs or set of wfss of $\mathcal{L}^{\text{PL}}$ which includes 5 sentence letters would require 32 lines, and double that again for 6 sentence letters. At least for humans using pen and paper, this is at about the limit for what it is possible to use without making mistakes.

Here one might be tempted to respond by appealing to computers. Instead of attempting to write out truth tables by hand, perhaps the method is best developed with computational assistance to avoid making mistakes. This leads into the *Boolean satisfiability problem* in computer science, and also falls outside the scope of this course. Rather, we will be concerned with methods for working out reasoning on paper in a finite amount of time. In later chapters, we will have reason to require these methods to simulate certain natural patterns of reasoning. However, before then it will be important to clean up a few loose ends in order to set the stage for these developments. We will begin by presenting a different procedure.

2.5 Partial Truth Tables

To show that a wfs of $\mathcal{L}^{\text{PL}}$ is a tautology, we need to show that 1 occurs below its main operator on every row of its complete truth table. So we need a complete truth table. However, to show that a wfs is *not* a tautology we only need to complete a row in which 0 is beneath its main operator. Therefore, in order to show that a wfs is not a tautology, it is enough to provide a *partial truth table* regardless of how many sentence letters the wfs might include.

For example, consider $(U \wedge T) \rightarrow (S \wedge W)$. We want to show that it is *not* a tautology by providing a partial truth table. To do so, we begin by writing 0 under the main operator which is a material conditional. In order for the conditional to be false, there must be a 1 under the antecedent and a 0 under the consequent. We fill these in as follows:

S	T	U	W	$(U$	$\wedge$	$T)$	$\rightarrow$	$(S$	$\wedge$	$W)$
					1		**0**		0	

In order for a 1 to occur under $U \wedge T$, a 1 must also occur under both U and T as follows:

S	T	U	W	$(U$	$\wedge$	$T)$	$\rightarrow$	$(S$	$\wedge$	$W)$
	1	1		1	1	1	**0**		0	

Remember that each instance of a given sentence letter must have the same truth-value in a given row of a truth table. You can't have 1 occur under one instance of U and 0 occur under another instance of U in the same row. Thus we put a 1 under each instance of U and T.

Now we just need to work out what follows from the 0 under $S \wedge W$. In particular, a 0 must occur under either S or W, or both. Making an arbitrary decision, we may finish the table:

S	T	U	W	$(U$	$\wedge$	$T)$	$\rightarrow$	$(S$	$\wedge$	$W)$
0	1	1	0	1	1	1	**0**	0	0	0

Although showing that a wfs of $\mathcal{L}^{\text{PL}}$ is a tautology requires a complete truth table, showing that a wfs of $\mathcal{L}^{\text{PL}}$ is not a tautology only requires a partial truth table with a single row where 0 occurs below the main operator of that wfs. That's what we've just done. In just the same way, to show that a wfs of $\mathcal{L}^{\text{PL}}$ is not a contradiction, you only need to construct a single row of a truth table where 1 occurs below the main operator of that wfs. By contrast, to show that a wfs of $\mathcal{L}^{\text{PL}}$ is a contradiction, you must show that a 0 occurs below the main operator on *every* row of a complete truth table, and so you need a complete truth table.

A wfs of $\mathcal{L}^{\text{PL}}$ is contingent just in case its complete truth table has a row in which 1 occurs below the main operator and another row in which 0 occurs below the main operator. Thus to show that a wfs of $\mathcal{L}^{\text{PL}}$ is contingent requires a partial truth table with just two rows. For example, we can show that the sentence above is contingent as follows:

S	T	U	W	$(U$	$\wedge$	$T)$	$\to$	$(S$	$\wedge$	$W)$
0	1	1	0	1	1	1	**0**	0	0	0
0	1	0	0	0	0	1	**1**	0	0	0

Just as there happens to be more than one combination of truth-values which makes the sentence false, there are even more ways to make the sentence true. However, this is not always the case. For instance, given a sentence letter A, there is exactly two lines in its complete truth table, one in which it is true, and the other in which it is false.

Showing that a wfs of $\mathcal{L}^{\text{PL}}$ is not contingent requires providing a complete truth table. In particular, one must either show that the complete truth table for the wfs has a 1 under its main operator on all rows, or show that the wfs has a 0 under its main operator on all rows. If you do not know whether a particular sentence is contingent or not, then you do not know whether you will need a complete or partial truth table. One way to proceed is to start working on a complete truth table, stopping as soon as you complete rows that show the sentence is contingent. If not, then you must complete the truth table in full.

Showing that two wfss of $\mathcal{L}^{\text{PL}}$ are logically equivalent requires providing a complete truth table. By contrast, showing that two wfss of $\mathcal{L}^{\text{PL}}$ are not logically equivalent only requires a partial truth table with one row in which a 1 occurs below the main operator of one of the wfss and a 0 occurs below the main operator of the other wfs.

Showing that a set of wfss of $\mathcal{L}^{\text{PL}}$ is satisfiable requires a single row of a truth table in which a 1 occurs below the main operator of every wfs in the set. However, to show that a set of wfss of $\mathcal{L}^{\text{PL}}$ is unsatisfiable requires a complete truth table since you must show that on every row of a complete truth table for the set, there is a 0 below the main operator of at least one of the wfss in the set. Of course, we could only hope to succeed if the set of wfss is finite.

Showing that an argument is valid requires a complete truth table. Showing that an argument is invalid only requires providing a single row of a partial truth table in which a 1 occurs below the main operator of every premise and a 0 occurs below the conclusion. Thus we have:

	YES	NO
Tautology	complete truth table	one-line partial truth table
Contradiction	complete truth table	one-line partial truth table
Contingent	two-line partial truth table	complete truth table
Entailment	complete truth table	one-line partial truth table
Equivalent	complete truth table	one-line partial truth table
Satisfiable	one-line partial truth table	complete truth table
Valid	complete truth table	one-line partial truth table

The table above summarizes when a complete truth table is required and when a partial truth table will suffice. If you are trying to remember whether you need a complete truth table or not, the general rule is, if you're looking to establish a claim about *every* interpretation, you need a complete table. Otherwise, a one-line or perhaps two-line truth table may do instead.

2.6 Semantics

Although partial truth tables might help to avoid doing some amount of work, invariably you will end up needing to construct complete truth tables for sometimes long wfs of $\mathcal{L}^{\text{PL}}$ or else for arguments or sets which include many wfss of $\mathcal{L}^{\text{PL}}$. In addition to being extremely tedious and time consuming, constructing large truth tables is also highly prone to human error. These provide some initial reasons that one might hope to devise an alternative.

Complete truth tables also played an important role in defining what it is for a wfs of $\mathcal{L}^{\text{PL}}$ to be a tautology, where something similar may be said for the definitions of a contradiction as well as a logically contingent wfs of $\mathcal{L}^{\text{PL}}$. Similarly, the definitions for logically entailment, logical equivalence, satisfiability, logical consequence, and the validity of arguments given above all appealed to complete truth tables. Whereas the corresponding definitions for the sentences of English could not be made precise given that there is no well-defined sense of what counts as a grammatical sentence of English, the sections above appealed to complete truth tables for the wfss of $\mathcal{L}^{\text{PL}}$ in order to avoid this problem. Nevertheless, the truth table definitions of the logical notions considered above still leave something to be desired.

So far we have identified the rows of a truth table for a relevant wfs or set of wfss of $\mathcal{L}^{\text{PL}}$ with the relevant range of interpretations for that wfs or set of wfss. Of course, the truth tables that we can write down are are finite in size and so cannot specify truth-values for every sentence letter of $\mathcal{L}^{\text{PL}}$. Rather, the rows of a truth table provide PARTIAL INTERPRETATIONS of $\mathcal{L}^{\text{PL}}$ by specifying all combinations of truth-values for some finite set of sentence letters. Insofar as a truth table includes all of the sentence letters that occur in the wfs(s) in question, the truth-values for all other sentence letters do not make a difference, and so may be safely ignored. Nevertheless, we cannot define the interpretations of $\mathcal{L}^{\text{PL}}$ as the rows of any finite truth table since no finite truth table specifies truth-values for every sentence letter. This provides further motivation to present a more general approach.

Next we may consider the definition of a complete truth table itself. Rather than providing a formal definition, we appealed to the table that results from writing the wfs(s) of $\mathcal{L}^{\text{PL}}$ in question at the top right of the table with all of the sentence letters it contains at the top left. We then provided a procedure for adding the appropriate number of rows depending on how many sentence letters were involved and distributing truth-values accordingly. The rest of the values in the table were then to be added by appealing to the characteristic truth tables as a rubric. This left open certain ambiguities like the order of the sentence letters as well as the order of the wfss in a complete truth table for a set of wfss of $\mathcal{L}^{\text{PL}}$. Additionally, at least given what we have said so far, one must rely on an intuitive grasp of the main operator for each subsentence, as well as how to correctly apply the characteristic truth tables.

Although with some ingenuity we could tighten up all of these details by either eliminating ambiguities or else establishing that the ambiguities do not make a difference, the definitions themselves are bound to become even more cumbersome to state precisely. Rather, the truth table definitions given above are best understood as intuitive approximations of the precise definitions to which we will soon turn. Despite their imprecision, the truth table definitions provide a natural and accessible account of the logical notions that we are after, and so may be preserved as helpful heuristics in contemplating the abstract definitions to follow.

Rather than relying on diagrams, officially we will take an INTERPRETATION of $\mathcal{L}^{\text{PL}}$ to be any function $\mathcal{I}$ from the set of sentence letters for $\mathcal{L}^{\text{PL}}$ to the set of truth-values $\{1, 0\}$, thereby assigning every sentence letter to exactly one truth-value. Although interpretations only specify the truth-values of sentence letters, we may draw on any interpretation $\mathcal{I}$ of $\mathcal{L}^{\text{PL}}$ in order to define another function which assigns every wfs of $\mathcal{L}^{\text{PL}}$ to exactly one truth-value in accordance with the characteristic truth tables but without appealing to the characteristic truth tables. More precisely, given any interpretation $\mathcal{I}$ of $\mathcal{L}^{\text{PL}}$, we may recursively define the VALUATION FUNCTION $\mathcal{V}_{\mathcal{I}}$ over the domain of wfss for $\mathcal{L}^{\text{PL}}$ by way of the following semantics:

Valuation Function: For any wfss φ and ψ of $\mathcal{L}^{\text{PL}}$:

$\mathcal{V}_{\mathcal{I}}(\varphi) = \mathcal{I}(\varphi)$ *if* φ is a sentence letter of $\mathcal{L}^{\text{PL}}$.

$\mathcal{V}_{\mathcal{I}}(\neg\varphi) = 1$ *iff* $\mathcal{V}_{\mathcal{I}}(\varphi) = 0$.

$\mathcal{V}_{\mathcal{I}}(\varphi \vee \psi) = 1$ *iff* $\mathcal{V}_{\mathcal{I}}(\varphi) = 1$ or $\mathcal{V}_{\mathcal{I}}(\psi) = 1$ (or both).

$\mathcal{V}_{\mathcal{I}}(\varphi \wedge \psi) = 1$ *iff* $\mathcal{V}_{\mathcal{I}}(\varphi) = 1$ and $\mathcal{V}_{\mathcal{I}}(\psi) = 1$.

$\mathcal{V}_{\mathcal{I}}(\varphi \rightarrow \psi) = 1$ *iff* $\mathcal{V}_{\mathcal{I}}(\varphi) = 0$ or $\mathcal{V}_{\mathcal{I}}(\psi) = 1$ (or both).

$\mathcal{V}_{\mathcal{I}}(\varphi \leftrightarrow \psi) = 1$ *iff* $\mathcal{V}_{\mathcal{I}}(\varphi) = \mathcal{V}_{\mathcal{I}}(\psi)$.

The clauses above hold for all sentences φ and ψ of $\mathcal{L}^{\text{PL}}$, thereby extending any interpretation $\mathcal{I}$ of $\mathcal{L}^{\text{PL}}$ to $\mathcal{V}_{\mathcal{I}}$ in order to specify a unique truth-value for every wfs of $\mathcal{L}^{\text{PL}}$. Whereas the characteristic truth tables for the operators specify the truth-values for complex sentences visually, the semantic clauses above specify the same information functionally.

In Chapter 1, we specified the primitive symbols for $\mathcal{L}^{\text{PL}}$. These included the sentence letters, punctuation, and sentential operators. When interpreting $\mathcal{L}^{\text{PL}}$, you are not allowed to change the meaning of the sentential operators. For instance, you cannot take the '$\neg$' symbol to mean what '$\wedge$' usually does. Rather, the '$\neg$' symbol will always have the same semantic clause, and so will always express the same truth-function for negation. Since the meanings for the sentential operators are fixed by the semantic clauses given in the definition of the valuation function, it is common to refer to the sentential operators as LOGICAL CONSTANTS.

The sentence letters are sometimes referred to as the NON-LOGICAL VOCABULARY and are interpreted by assigning them to either 1 or 0, where it is the combination of assignments which may differ between interpretations of $\mathcal{L}^{\text{PL}}$. Accordingly, when we translate an argument from English into $\mathcal{L}^{\text{PL}}$, the sentence letter 'M' does not have its meaning fixed as a result. Rather, we rely on interpretations to assign truth-values to sentence letter such as 'M', where the truth-values provide a maximally course-grained way to model what those sentence letters mean, i.e., whether they express a proposition that obtains or does not obtain.

2.7 Formal Definitions

Having provided a definition of the interpretations of $\mathcal{L}^{\text{PL}}$ that is both mathematically precise and simple to state, we may now put this definition to work to redefine the logical properties and relations discussed above. Letting φ and ψ be wfs of $\mathcal{L}^{\text{PL}}$, Γ be a set of wfss of $\mathcal{L}^{\text{PL}}$, and $\mathcal{I}$ and $\mathcal{J}$ be interpretations of $\mathcal{L}^{\text{PL}}$, we may present the following official definitions:

TAUTOLOGY: φ is a *tautology iff* $\mathcal{V}_{\mathcal{I}}(\varphi) = 1$ for all $\mathcal{I}$.

CONTRADICTION: φ is a *contradiction iff* $\mathcal{V}_{\mathcal{I}}(\varphi) = 0$ for all $\mathcal{I}$.

CONTINGENT: φ is *logically contingent iff* $\mathcal{V}_{\mathcal{I}}(\varphi) \neq \mathcal{V}_{\mathcal{J}}(\varphi)$ for some $\mathcal{I}$ and $\mathcal{J}$.

ENTAILMENT: φ *logically entails* ψ *iff* $\mathcal{V}_{\mathcal{I}}(\varphi) \leqslant \mathcal{V}_{\mathcal{I}}(\psi)$ for all $\mathcal{I}$.

EQUIVALENCE: φ is *logically equivalent* to ψ *iff* $\mathcal{V}_{\mathcal{I}}(\varphi) = \mathcal{V}_{\mathcal{I}}(\psi)$ for all $\mathcal{I}$.

SATISFIABLE: Γ is *satisfiable iff* there is some $\mathcal{I}$ where $\mathcal{V}_{\mathcal{I}}(\gamma) = 1$ for all $\gamma \in \Gamma$.

CONSEQUENCE: $\Gamma \vDash \varphi$ *iff* $\mathcal{V}_{\mathcal{I}}(\varphi) = 1$ for all $\mathcal{I}$ where $\mathcal{V}_{\mathcal{I}}(\gamma) = 1$ for all $\gamma \in \Gamma$.

Note that the double turnstile '$\vDash$' has been introduced for the logical consequence relation which— like the schematic variable 'φ'— is part of the *metalanguage* that we are using to discuss $\mathcal{L}^{\text{PL}}$, and not a part of $\mathcal{L}^{\text{PL}}$ itself. Although officially $\vDash$ takes a set of $\mathcal{L}^{\text{PL}}$ wfss on the left together with a single $\mathcal{L}^{\text{PL}}$ wfs on the right, it is both common and convenient to drop the set notation, writing '$\varphi_1, \ldots, \varphi_n \vDash \psi$' instead of '$\{\varphi_1, \ldots, \varphi_n\} \vDash \psi$'.

As before, we may say that an argument in $\mathcal{L}^{\text{PL}}$ is VALID just in case it's conclusion is a logical consequence of its set of premises. Recall that an argument in $\mathcal{L}^{\text{PL}}$ is a sequence of $\mathcal{L}^{\text{PL}}$ wfs, and so a completely different type of thing than a set of wfs of $\mathcal{L}^{\text{PL}}$. After all, sets do not specify the order of their members. Accordingly, we cannot say that for any set of $\mathcal{L}^{\text{PL}}$ wfss Γ and wfs φ, if $\Gamma \vDash \varphi$, then *the* argument whose premises are the wfss in Γ and whose conclusion is φ is valid. This is because there may fail to be a unique argument that we can construct from a set of $\mathcal{L}^{\text{PL}}$ wfss Γ and a further $\mathcal{L}^{\text{PL}}$ wfs φ. For instance, assuming Γ includes at least two wfss, we can construct different arguments by reversing the order of the premises. Instead, we may claim something more general: $\Gamma \vDash \varphi$ just in case *every* argument where the wfss in Γ are the premises (in some order or other) and φ is the conclusion is valid.

Logical consequence is the most important semantic concept that we will study in this course. In the following chapter, we will introduce a proof theoretic analogue for derivability which we will represent with the single turnstile '$\vdash$'. As we will then go on to show in the metalogical portions of this book, these two relations have the same extension, providing two radically different perspectives on one and the same thing, i.e., formal reasoning. We will then repeat this same methodology for languages with greater expressive power. For the time being, we only pause to indicate the central role that logical consequence will play throughout this text.

2.8 Semantic Proofs

The formal definitions given above are certainly much more elegant than the truth table definitions for the corresponding terms, avoiding all the ambiguities that we mentioned but did not take the time to fully resolve above. Although these new definitions do not provide effective procedures for deciding logical questions in the same way as the truth table definitions, they will nevertheless allow us to prove things about the wfss of $\mathcal{L}^{\text{PL}}$ and their various logical properties and relationships. For instance, consider the following set of wfss:

$$\Gamma = \{A \wedge C,\ B \leftrightarrow \neg A,\ C \rightarrow (B \wedge D),\ E\}$$

It turns out that this set is unsatisfiable. However, to show this using a truth table would requiring constructing a complete truth table with five sentence letters and thirty-two rows. In addition to the tedium, the chances of making a mistake cannot be overlooked. Instead of attempting this, we can prove that Γ is unsatisfiable by assuming that it is satisfiable and appealing to the formal definitions in order to derive a contradiction:

Proof: Assume for contradiction that the set Γ is satisfiable. By the definition of satisfiability, $\mathcal{V}_{\mathcal{I}}(A \wedge C) = \mathcal{V}_{\mathcal{I}}(B \leftrightarrow \neg A) = \mathcal{V}_{\mathcal{I}}(C \rightarrow (B \wedge D)) = \mathcal{V}_{\mathcal{I}}(E) = 1$ for some interpretation $\mathcal{I}$ of $\mathcal{L}^{\text{PL}}$. It follows that $\mathcal{V}_{\mathcal{I}}(A) = \mathcal{V}_{\mathcal{I}}(C) = 1$ by the semantics for conjunction, and so $\mathcal{V}_{\mathcal{I}}(\neg A) = 0$ given the semantics for negation. Since $\mathcal{V}_{\mathcal{I}}(B) = \mathcal{V}_{\mathcal{I}}(\neg A)$ by the semantics for the biconditional, $\mathcal{V}_{\mathcal{I}}(B) = 0$.

Since $\mathcal{V}_{\mathcal{I}}(C \rightarrow (B \wedge D)) = 1$, we know by the semantics for the conditional that either $\mathcal{V}_{\mathcal{I}}(C) = 0$ or $\mathcal{V}_{\mathcal{I}}(B \wedge D) = 1$. Having already shown that $\mathcal{V}_{\mathcal{I}}(C) \neq 0$, we may conclude that $\mathcal{V}_{\mathcal{I}}(B \wedge D) = 1$, and so $\mathcal{V}_{\mathcal{I}}(B) = 1$ and $\mathcal{V}_{\mathcal{I}}(D) = 1$ by the semantics for conjunction. Thus $\mathcal{V}_{\mathcal{I}}(B) \neq 0$, contradicting the above. □

This proof has been made fully explicit so as to make it easy to follow. A proof that is hard to follow is not a very good proof. However, using too many words is not a good thing either, cluttering what might be easier to see otherwise. Writing clear and readable proofs is a skill that requires judgment and practice. If you are new to proof writing, it is best to begin by making everything explicit before tightening things up to write with concision.

Note that the proof stopped once we got a contradiction. Although we might have continued by saying that our assumption must be false given the contradiction that we derived, this much is automatic given the introductory clause "Assume for contradiction…" which sets up this expectation. This is a good example of the kinds of writing strategies that you can use to write informal proofs. By contrast to the formal proofs that we will go on to write in the following chapter, informal proofs are written in the metalanguage mathematical English and are often about wfss of our object language $\mathcal{L}^{\text{PL}}$. In the case above, we established that Γ is unsatisfiable. In order to give you a sense of some of the other methods and phrasing for writing clear and concise informal proofs, it will help to consider a few more examples.

D1. Either the butler is the murderer or the gardener isn't who he says he is.
D2. The gardener is who he says he is.
D3. The butler is the murderer.

In order to determine its validity, let's translate this argument into $\mathcal{L}^{\text{PL}}$ and evaluate the resulting formal argument for validity with a truth table. In particular:

B: The butler is the murderer.
G: The gardener is who he says he is.

E1. $B \vee \neg G$
E2. G
E3. B

It should be pretty clear that this is a valid argument, but to show this we may now write a semantic proof which establishes its validity. Whereas the proof given above proceeded by *reductio ad absurdum*, deriving a contradiction from the negation of what we wanted to prove, it is perhaps easiest to write a direct proof that the argument above is valid.

> *Proof:* Let $\mathcal{I}$ be an arbitrary interpretation of $\mathcal{L}^{\text{PL}}$ for which both of the premises are true, i.e., $\mathcal{V}_{\mathcal{I}}(B \vee \neg G) = \mathcal{V}_{\mathcal{I}}(G) = 1$. We know by the semantics for negation that $\mathcal{V}_{\mathcal{I}}(\neg G) = 0$, where either $\mathcal{V}_{\mathcal{I}}(B) = 1$ or $\mathcal{V}_{\mathcal{I}}(\neg G) = 1$ by the semantics for disjunction, and so may conclude that $\mathcal{V}_{\mathcal{I}}(B) = 1$. Since $\mathcal{I}$ was arbitrary, we may conclude more generally that $\mathcal{V}_{\mathcal{I}}(B) = 1$ for any $\mathcal{I}$ where $\mathcal{V}_{\mathcal{I}}(B \vee \neg G) = \mathcal{V}_{\mathcal{I}}(G) = 1$, and so the argument is indeed valid. $\square$

It is typical to leave this final sentence off assuming that it is sufficiently clear what you are setting out to prove and how you are intended to do so. Nevertheless, it doesn't hurt to include this extra line to make it especially clear in case you are uncertain whether your proof is easy to interpret. In the case above, the key signpost that we used was the construction '*Let $\mathcal{I}$ be an arbitrary interpretation of $\mathcal{L}^{\text{PL}}$ ...*' since this signals that we will establish something general about $\mathcal{L}^{\text{PL}}$ interpretations. We then restrict consideration to those interpretations $\mathcal{I}$ of $\mathcal{L}^{\text{PL}}$ which make the premises true with 'for which both of the premises are true', checking to see that such an arbitrary interpretation makes the conclusion true.

Whereas the first semantic argument that we gave allowed us to show that Γ is unsatisfiable without having to draw a truth table with thirty-two rows, the winnings in the case above are somewhat less substantial. In particular, the truth table only requires that we include four rows since there are just two sentence letters. Thus we have:

B	G	$(B$	$\vee$	$\neg$	$G)$	G	B
1	1	1	**1**	0	1	**1**	**1**
1	0	1	**1**	1	0	**0**	**1**
0	1	0	**0**	0	1	**1**	**0**
0	0	0	**1**	1	0	**0**	**0**

If the argument were invalid, there would be a row on which the first two bold values are 1 but the third is 0. There is no such row, so the argument is valid. Put otherwise, every row in which the premises have a 1 under their main operators is a row in which the conclusion has a 1 under its main operator. Thus the argument in $\mathcal{L}^{\text{PL}}$ is valid.

In this case, both the semantic proof and truth table methods require a similar amount of work. Given either method, we may conclude that the argument in $\mathcal{L}^{\text{PL}}$ is valid. We may then go on to explain that the English argument is valid by appealing to the fact that the English argument can be regimented by a valid argument in $\mathcal{L}^{\text{PL}}$.

Before pressing on, it will be important to consider one more type of semantic proof which appeals to cases. Consider the following argument:

F1. Either Kat or Lu will with the race.
F2. If Kat wins, then she will celebrate.
F3. If Lu wins, then she will celebrate.
F4. Either Kat or Lu will celebrate.

This argument can be regimented as follows:

K: Kat will win.
L: Lu will win.
C_1: Kat will celebrate.
C_2: Lu will celebrate.

G1. $K \vee L$
G2. $K \rightarrow C_1$
G3. $L \rightarrow C_2$
G4. $C_1 \vee C_2$

This argument may seem plain enough since this is just the kind of day-to-day reasoning that we are all accustomed to doing. Nevertheless, providing a truth table would require sixteen rows. Although the semantic proof is easier than attempting to fill out so many rows without making any mistakes, there is no way to avoid a *proof by cases*, at least insofar as we are to provide a direct proof that the $\mathcal{L}^{\text{PL}}$ argument is valid.

Writing with slightly more concision than before, we may reason as follows:

Proof: Let $\mathcal{I}$ be an arbitrary interpretation of $\mathcal{L}^{\text{PL}}$ where:

$$\mathcal{V}_{\mathcal{I}}(K \vee L) = \mathcal{V}_{\mathcal{I}}(K \to C_1) = \mathcal{V}_{\mathcal{I}}(L \to C_2) = 1.$$

By the semantics for disjunction, either $\mathcal{V}_{\mathcal{I}}(K) = 1$ or $\mathcal{V}_{\mathcal{I}}(L) = 1$. Consider:

Case 1: Assume $\mathcal{V}_{\mathcal{I}}(K) = 1$. By the semantics for the conditional, it follows from the above that either $\mathcal{V}_{\mathcal{I}}(K) = 0$ or $\mathcal{V}_{\mathcal{I}}(C_1) = 1$, and so $\mathcal{V}_{\mathcal{I}}(C_1) = 1$. Thus $\mathcal{V}_{\mathcal{I}}(C_1 \vee C_2) = 1$ follows from the semantics for disjunction.

Case 2: Assume $\mathcal{V}_{\mathcal{I}}(L) = 1$. By the semantics for the conditional, it follows from the above that either $\mathcal{V}_{\mathcal{I}}(L) = 0$ or $\mathcal{V}_{\mathcal{I}}(C_2) = 1$, and so $\mathcal{V}_{\mathcal{I}}(C_1) = 1$. Thus $\mathcal{V}_{\mathcal{I}}(C_1 \vee C_2) = 1$ follows from the semantics for disjunction.

Thus $\mathcal{V}_{\mathcal{I}}(C_1 \vee C_2) = 1$ in both cases. Since $\mathcal{I}$ was arbitrary, the conclusion is true on any interpretation where the premises true, and so the argument is valid. □

Although the method of proof by cases is an essential tool to have in your toolkit, these proofs are often harder to read and can also be harder to write clearly. Accordingly, proofs by cases are to be avoided whenever possible. In the case above, we could have avoided introducing cases by using a *reductio* argument (short for *reductio ad absurdum*) instead.

Proof: Assume for contradiction that the argument is invalid. Thus the conclusion is not a logical consequence of the premises, and so there is some interpretation $\mathcal{I}$ of $\mathcal{L}^{\text{PL}}$ where $\mathcal{V}_{\mathcal{I}}(K \vee L) = \mathcal{V}_{\mathcal{I}}(K \to C_1) = \mathcal{V}_{\mathcal{I}}(L \to C_2) = 1$ and $\mathcal{V}_{\mathcal{I}}(C_1 \vee C_2) = 0$. By the semantics for disjunction, both $\mathcal{V}_{\mathcal{I}}(C_1) = \mathcal{V}_{\mathcal{I}}(C_2) = 0$.

By the semantics for the material conditional, either $\mathcal{V}_{\mathcal{I}}(K) = 0$ or $\mathcal{V}_{\mathcal{I}}(C_1) = 1$, and similarly, either $\mathcal{V}_{\mathcal{I}}(L) = 0$ or $\mathcal{V}_{\mathcal{I}}(C_2) = 1$. Given the above, $\mathcal{V}_{\mathcal{I}}(K) = \mathcal{V}_{\mathcal{I}}(L) = 0$, and so $\mathcal{V}_{\mathcal{I}}(K \vee L) = 0$ by the semantics for disjunction, contradicting the above. □

This proof doesn't worry about cases and in that way is a bit easier to follow. Nevertheless, direct proofs are often preferable to *reductio* style proofs since deriving a contradiction is not quite as explanatory as what a direct proof shows. Which proof strategy to use is a judgment call and it can take some practice to know which approach is best to use when. Even once you have chosen a basic strategy, the way that you order the steps of your proof can also have a significant effect on the clarity of your resulting proof. These are points that are worth considering as you practice writing clear and concise proofs in this course.

So far, we have only provided semantic proofs for claims that would have required a complete truth table. This is no accident. For instance, in the case where a set of $\mathcal{L}^{\text{PL}}$ wfss is satisfiable, or an argument $\mathcal{L}^{\text{PL}}$ is valid, all we need to do is find a particular interpretation of the relevant sentence letters of $\mathcal{L}^{\text{PL}}$ in order to draw the desired conclusion. Here we may observe that providing a partial truth table does just that, and so is often to be preferred. We may then go on to record a particular row of a truth table by using the notation $\mathcal{I}(\varphi)$ to specify a truth value for each sentence letter φ that occurs in the truth table.

2.9 Tautologies and Weakening

What should we make of the following claim:

$$P \wedge Q \vDash A \leftrightarrow \neg\neg A.$$

Notice that the sentence letters on the left-hand-side are unrelated to the sentence letters on the right-hand-side. So there is a straightforward sense in which the two sides of the logical consequence above have *nothing to do with one another*. Nevertheless, the claim above is true: every interpretation for which the wfs on the left is true is also an interpretation on which the wfs on the right is true for the simple reason that $A \leftrightarrow \neg\neg A$ is true on every interpretation whatsoever. As a result, we could have dropped the wfs on the left entirely, writing:

$$\vDash A \leftrightarrow \neg\neg A.$$

The claim above is short for $\varnothing \vDash A \leftrightarrow \neg\neg A$ where $\varnothing$ is a convenient notation for the empty set $\{\}$, i.e., the set of no wfs of $\mathcal{L}^{\text{PL}}$. Since we said above that we are often going to drop the set notation when stating logical consequences, we do not need to write the empty set at all when a wfs is a logical consequence of the empty set of wfss as above.

It is easy to show that a wfs φ of $\mathcal{L}^{\text{PL}}$ is a tautology just in case $\vDash \varphi$. Since there are no wfss in $\varnothing$, every interpretation $\mathcal{I}$ of $\mathcal{L}^{\text{PL}}$ vacuously makes all of the wfs in $\varnothing$ true, and so $\vDash \varphi$ just in case $\mathcal{V}_{\mathcal{I}}(\varphi) = 1$ for every interpretation $\mathcal{I}$ of $\mathcal{L}^{\text{PL}}$, i.e., φ is a tautology. Given this equivalence, it is common to define the $\mathcal{L}^{\text{PL}}$ tautologies in terms of logical consequence. In particular, we could have said that a wfs φ of $\mathcal{L}^{\text{PL}}$ is a TAUTOLOGY just in case $\vDash \varphi$.

In order to get another perspective on why φ is a tautology just in case $\vDash \varphi$, it will help to consider the set of interpretations that make a wfs φ of $\mathcal{L}^{\text{PL}}$ true. For brevity, we may define the INTERPRETATION SET $|\varphi| \coloneqq \{\mathcal{I} : \mathcal{V}_{\mathcal{I}}(\varphi) = 1\}$ to be the set of all and only those $\mathcal{L}^{\text{PL}}$ interpretations $\mathcal{I}$ for which $\mathcal{V}_{\mathcal{I}}(\varphi) = 1$. That is, given the set of all interpretations, each wfs φ of $\mathcal{L}^{\text{PL}}$ corresponds to a unique subset of interpretations in which φ is true, i.e., $\mathcal{V}_{\mathcal{I}}(\varphi) = 1$. Or to put it another way, each wfs φ of $\mathcal{L}^{\text{PL}}$ amounts to a *constraint* on the interpretations of $\mathcal{L}^{\text{PL}}$ where only those interpretations $\mathcal{I}$ for which $\mathcal{V}_{\mathcal{I}}(\varphi) = 1$ satisfy the constraint.

Given the definition of an interpretation sets for the wfss of $\mathcal{L}^{\text{PL}}$, we may provide a slightly more abstract characterization of logical consequence. By the official definition, $\Gamma \vDash \varphi$ asserts that every $\mathcal{L}^{\text{PL}}$ interpretation $\mathcal{I}$ which makes every $\gamma \in \Gamma$ true also makes φ true. Another way to say this is that the intersection of all of the interpretation sets $|\gamma|$ for $\gamma \in \Gamma$ is a subset of the interpretation set for the conclusion $|\varphi|$. Formally, we may write this as follows:

$$\bigcap\{|\gamma| : \gamma \in \Gamma\} \subseteq |\varphi|.$$

What this says is that every interpretation $\mathcal{I}$ which belongs to every interpretation set $|\gamma|$ for $\gamma \in \Gamma$ is also in the interpretation set $|\varphi|$ for the conclusion. Since an interpretation $\mathcal{I}$ belongs to an interpretation set $|\psi|$ just in case $\mathcal{V}_{\mathcal{I}}(\psi) = 1$, this is equivalent to requiring every interpretation $\mathcal{I}$ where $\mathcal{V}_{\mathcal{I}}(\gamma) = 1$ for all $\gamma \in \Gamma$ to be such that $\mathcal{V}_{\mathcal{I}}(\varphi) = 1$. But this is exactly what is asserted by $\Gamma \vDash \varphi$, only in the language of set theory.

Despite the equivalence, we will maintain our notation and definition for logical consequence $\Gamma \vDash \varphi$ rather than the set theoretic analogue given above. Nevertheless, it can help to take more than one perspective on the same thing, especially for a concept that is as important as logical consequence. The set theoretic analogue given above provides a vivid account of the way that each $\gamma \in \Gamma$ may constrain the interpretations of $\mathcal{L}^{\text{PL}}$ in which φ is said to be true. If there are no wfss in Γ at all, i.e., $\Gamma = \varnothing$, this corresponds to imposing no constraints on the interpretations of $\mathcal{L}^{\text{PL}}$ in which φ is said to be true. Put otherwise, all interpretations belong to $|\varphi|$, and so $\mathcal{V}_{\mathcal{I}}(\varphi) = 1$ for all interpretations $\mathcal{I}$ of $\mathcal{L}^{\text{PL}}$, i.e., φ is a tautology.

In order to make this more concrete, consider the following claims:

H1. $\vDash P \vee \neg P$.
H2. $\vDash P \leftrightarrow (P \vee (P \wedge Q))$.
H3. $\vDash (P \wedge \neg P) \to (A \vee B)$.

Although for different reasons, each of the claims above is true, and so the wfss on the right are all tautologies. It follows that the following claims are also true:

I1. $A, B \ \vDash P \vee \neg P$.
I2. $\neg P \ \vDash P \leftrightarrow (P \vee (P \wedge Q))$.
I3. $C \vee \neg C \ \vDash (P \wedge \neg P) \to (A \vee B)$.

Given that the wfss on the right are all tautologies, we can add whatever wfss that we like on the left. More generally, it is easy to prove the following principle:

Lemma 2.1 (Weakening) If $\Gamma \vDash \varphi$, then $\Gamma \cup \Sigma \vDash \varphi$.

This principle says that whenever φ is a logical consequence of Γ, then φ is also a logical consequence of $\Gamma \cup \Sigma$ for any set of wfss Σ of $\mathcal{L}^{\text{PL}}$ whatsoever where $\Gamma \cup \Sigma$ is the union set including all of the wfss in either Γ or Σ and nothing besides. In the special case where $\Gamma = \varnothing$ is empty, we may conclude that if φ is a tautology (i.e., $\vDash \varphi$), then φ is also a logical consequence of every set Σ of wfss of $\mathcal{L}^{\text{PL}}$ (i.e., $\Sigma \vDash \varphi$). It is an inference of this kind which is what justified drawing I1 – I3 as conclusions from H1 – H3 above.

The reason that weakening holds is easy to see given the set theoretic perspective on logical consequence presented before. Given that a logical consequence $\Gamma \vDash \varphi$ holds, adding further conditions beyond just those included in Γ will only further restrict the interpretations of $\mathcal{L}^{\text{PL}}$ for which φ is said to be true. Since we already know that all interpretations that make every $\gamma \in \Gamma$ true also make φ true given that $\Gamma \vDash \varphi$, then any subset of the interpretations that make every $\gamma \in \Gamma$ true is sure to also make φ true, and so $\Gamma \cup \Sigma \vDash \varphi$ for any set of wfss Σ. To put the point set theoretically, $\bigcap\{|\gamma| : \gamma \in \Gamma \cup \Sigma\} \subseteq \bigcap\{|\gamma| : \gamma \in \Gamma\}$ since $\Gamma \subseteq \Gamma \cup \Sigma$.

2.10 Contradictions and Unsatisfiability

What if a given set Γ of wfss $\mathcal{L}^{\text{PL}}$ imposes so many constraints that there is no interpretation $\mathcal{I}$ of $\mathcal{L}^{\text{PL}}$ which makes all the wfss $\gamma \in \Gamma$ true, i.e., $\mathcal{V}_{\mathcal{I}}(\gamma) = 1$ for all $\gamma \in \Gamma$? Put otherwise, what if Γ is unsatisfiable? In such a case, $\bigcap\{|\gamma| : \gamma \in \Gamma\} = \varnothing$, and since the empty set $\varnothing$ is a subset of every set, we know that $\varnothing \subseteq |\varphi|$ for any wfs φ of $\mathcal{L}^{\text{PL}}$. That is if there are no interpretations that make all of the wfss $\gamma \in \Gamma$ true, then it follows vacuously that all zero of those interpretations are guaranteed to make φ true. More succinctly, every wfs φ of $\mathcal{L}^{\text{PL}}$ is a logical consequence of any unsatisfiable set Γ of wfss of $\mathcal{L}^{\text{PL}}$ whatsoever.

Consider the following examples:

J1. $Q, \neg Q \vDash P$.
J2. $R \to (Q \wedge \neg Q), R \vDash \neg R$.
J3. $\neg(A \to A) \vDash (P \wedge \neg P) \to (A \vee B)$.

Some of the above are more obvious than others. For instance, it is easy to see that there is no interpretation of $\mathcal{L}^{\text{PL}}$ in which both $\mathcal{V}_{\mathcal{I}}(Q) = 1$ and $\mathcal{V}_{\mathcal{I}}(\neg Q) = 1$ since a contradiction follows immediately from the semantics for negation were we to assume otherwise. As a result, J1 is valid for the vacuous reason that no interpretation satisfies $\{Q, \neg Q\}$. Although slightly more work is required, J2 is also valid since $\{R \to (Q \wedge \neg Q), R\}$ is also unsatisfiable.

By similar reasoning, we can show that J3 is valid since $\{\neg(A \to A)\}$ is also unsatisfiable. However, unlike the previous cases, this set has only one member, namely the wfs $\neg(A \to A)$. Since $\neg(A \to A)$ can be shown to be a contradiction, it follows that any set of wfss to which it belongs is unsatisfiable. In particular, the singleton set $\{\neg(A \to A)\}$ is unsatisfiable, and so every wfs φ of $\mathcal{L}^{\text{PL}}$ is a logical consequence of $\{\neg(A \to A)\}$. More generally, any set containing a contradiction is unsatisfiable for the simple reason that there is no interpretation which makes a contradiction true. As a result, every wfs of $\mathcal{L}^{\text{PL}}$ is a logical consequence of a set of wfss of $\mathcal{L}^{\text{PL}}$ which includes a contradiction.

Having defined what it is for a wfs of $\mathcal{L}^{\text{PL}}$ to be a tautology in terms of logical consequence, it is natural to consider how to use the notion of logical consequence to define what it is for a wfs of $\mathcal{L}^{\text{PL}}$ to be a contradiction. It will help to consider the following claim:

$$P \wedge \neg P \vDash Q.$$

This statement is true. It says that Q is true in every $\mathcal{L}^{\text{PL}}$ interpretation in which $P \wedge \neg P$ is true. This follows vacuously since $P \wedge \neg P$ is not true in any $\mathcal{L}^{\text{PL}}$ interpretations, and so Q is true in every interpretation in which $P \wedge \neg P$ is true (all zero of them). Or to take another approach, think about what it would take for the logical consequence above to be false: there would have to be a $\mathcal{L}^{\text{PL}}$ interpretation in which $P \wedge \neg P$ is true and Q is false. But there are no $\mathcal{L}^{\text{PL}}$ interpretations in which $P \wedge \neg P$ is true, and so Q is a logical consequence.

The logical consequence above has nothing to do with the wfs Q. In particular, we do not need to appeal to anything about the logical form of Q in order to explain why $P \wedge \neg P \vDash Q$. Thus the same considerations would demonstrate that $P \wedge \neg P \vDash \varphi$ for any wfs φ of $\mathcal{L}^{\text{PL}}$. Moreover, the same conclusion holds were we to replace $P \wedge \neg P$ with any other $\mathcal{L}^{\text{PL}}$ wfs that is not true on any interpretation. For instance, consider:

K1. $A \wedge \neg A \vDash \neg Q \to R$.
K2. $\neg(P \vee \neg P) \vDash (A_1 \vee A_2) \to \neg(A_3 \leftrightarrow (A_4 \wedge \neg A_2))$.

Since there are no $\mathcal{L}^{\text{PL}}$ interpretations in which the wfss on the left are true, you do not need to examine the wfss on the right to confirm that the logical consequences above are true. Given that the wfss on the right do not matter, it would be nice to have a way to represent that any wfs φ of $\mathcal{L}^{\text{PL}}$ is a logical consequence of the wfss on the left. There are two common ways to do this, but both make use of the same notation: $\Gamma \vDash \bot$. One way to interpret the logical consequence above is as a universal claim that Γ entails every wfs φ of $\mathcal{L}^{\text{PL}}$ whatsoever. Another way to interpret this logical consequence is to take '$\bot$' to abbreviate some particular contradiction of $\mathcal{L}^{\text{PL}}$, though it doesn't matter which. These conventions turn out to amount to the very same thing. For simplicity, we will assume the latter convention where '$\bot$' abbreviates the contradiction '$A \wedge \neg A$' for definiteness.

We have already observed that every wfs φ of $\mathcal{L}^{\text{PL}}$ is a logical consequence of an unsatisfiable set Γ of wfss of $\mathcal{L}^{\text{PL}}$. Given any unsatisfiable set Γ of wfss of $\mathcal{L}^{\text{PL}}$, it follows that $\bot$ in particular is a logical consequence of Γ, i.e., $\Gamma \vDash \bot$. Conversely, we may prove the following:

Lemma 2.2 If $\Gamma \vDash \bot$, then Γ is unsatisfiable.

> *Proof:* Let Γ be an arbitrary set of wfss of $\mathcal{L}^{\text{PL}}$ where $\Gamma \vDash \bot$. Assume that Γ is satisfiable for contradiction. Thus there is a $\mathcal{L}^{\text{PL}}$ interpretation $\mathcal{I}$ where $\mathcal{V}_{\mathcal{I}}(\gamma) = 1$ for all $\gamma \in \Gamma$, and so $\mathcal{V}_{\mathcal{I}}(\bot) = 1$ given the assumption that $\Gamma \vDash \bot$. Given the definition of $\bot$, we may conclude that $\mathcal{V}_{\mathcal{I}}(A \wedge \neg A) = 1$. By the semantics for conjunction $\mathcal{V}_{\mathcal{I}}(A) = \mathcal{V}_{\mathcal{I}}(\neg A) = 1$, and so $\mathcal{V}_{\mathcal{I}}(A) = 0$ by the semantics for negation, resulting in a direct contradiction. Thus Γ is unsatisfiable. □

This shows that a set Γ of wfss of $\mathcal{L}^{\text{PL}}$ is unsatisfiable if and only if $\Gamma \vDash \bot$. In the special case where $\Gamma = \{\varphi\}$, we may say that φ is a contradiction just in case $\varphi \vDash \bot$. This provides a way to characterize contradictions in terms of logical consequence.

We may close by mentioning a connection between logical consequence and unsatisfiability that we will have occasion to return to later in the metalogical portions of this text.

Lemma 2.3 $\Gamma \vDash \varphi$ just in case $\Gamma \cup \{\neg\varphi\}$ is unsatisfiable.

We will make use of this principle in proving things about the proof system for $\mathcal{L}^{\text{PL}}$ that we will now turn to introduce in the following chapter.

Chapter 3

Natural Deduction in PL

This chapter introduces a natural deduction proof system for $\mathcal{L}^{\text{PL}}$ which we will refer to as *Propositional Logic* (PL). This is the logic of sentences which aims to describe formal reasoning which is valid in virtue of the logical form of the sentences involved where the syntactic primitives are all sentences or sentential operators. In a future chapter, we will introduce *First-Order Logic* (FOL) for $\mathcal{L}^{\text{FOL}}$ which includes predicates, constants, variables, and quantifiers. Until then, we will continue to restrict attention to what can be expressed with the resources included in $\mathcal{L}^{\text{PL}}$. Accordingly, PL does not aim to describe *all* of formal reasoning, but rather only the formal reasoning that can be carried out in $\mathcal{L}^{\text{PL}}$.

In Chapter 2, the logical consequence relation $\vDash$ provided a first answer to the question of what logic aims to study. Additionally, we presented the truth table method and semantic proof method for establishing which logical consequences hold and which do not. Nevertheless, these methods leave something to be desired. To begin with, we saw just how poorly the truth table method scales with the number of sentence letters, making this method practically infeasible for more than four sentence letters. Although the semantic proof method did not face this same problem, semantic proofs were often cumbersome to write where their construction was completely unconstrained. That is, we never said, and indeed cannot say, what counts as an adequate semantic proof. Rather, these proofs took place in our metalanguage mathematical English which does not have clear cut boundaries or rules.

Recall the strategies for writing semantic proofs that we began to describe in the previous chapter. For instance, these included proof by contradiction and proof by cases. This raises a question about what are *all* of the strategies that one might employ along these lines. More than strategies, we want to know what are *all* of the moves that we can make when writing a proof, and what inferences are absolutely basic and cannot be subdivided into further steps. Questions of these kinds lead to a completely different approach to our present inquiry into the nature of formal reasoning. Instead of asking what is a logical consequence of what by quantifying over the interpretations of a language, we may seek to describe a collection of basic inferences, chaining these together in order to say what can be inferred from what. This chapter will be concerned to answer this question by providing a proof system for $\mathcal{L}^{\text{PL}}$.

After considering a number of arguments in English in Chapter 0, we observed that natural languages do not have well-defined boundaries, frustrating any attempt to say something completely general about all sentences and arguments in English. Chapter 1 avoided this problem by presenting the artificial language $\mathcal{L}^{\text{PL}}$ which has a well-defined notion of a wfs that we may use to regiment English sentences and arguments. Although regimentation itself remains a matter of judgment with no definite answers, this method nevertheless provided a way to identify the logical forms that explain why certain patterns of reasoning in English are especially compelling. In particular, we defined the interpretations of $\mathcal{L}^{\text{PL}}$ to be functions from the wfss of $\mathcal{L}^{\text{PL}}$ to truth-values, drawing on this definition in order to introduce logical consequence along with a number of other logical properties and relations. However, none of this would have been possible were we to attempt these definitions for English.

In just the same way that it was important to work with an artificial language in order to provide a mathematically precise definition of logical consequence, it will also be important to draw on a well-defined language in order to describe the basic inferences that hold in virtue of their logical forms. Rather than introducing another artificial language, we will continue to work with $\mathcal{L}^{\text{PL}}$ maintaining all of the definitions from before in order to provide the proof system PL, also called a *logic*, for reasoning in $\mathcal{L}^{\text{PL}}$. Functionally, you can think of PL as including rules somewhat akin to the proof strategies and steps that we used in writing informal proofs before, only now they will take a precise mathematical form.

What of our informal proofs from before? Are they to be trusted given that they are written in the vague natural language English together with some mathematical conventions? What of mathematics proofs in general which are also written in this kind of language? Are these really mere approximations whose validity is only to be accounted for by regimenting them in some more precise language? You might be surprised to know that the answer is 'No'.

Rather than encoding some final truth, the logical systems that we will present are better understood to be abstractions from the intuitive bedrock from which we must begin: natural language, and in our case, English. After all, how would you ever hope to learn what the sentential operators (much less the sentence letters) of $\mathcal{L}^{\text{PL}}$ mean? The semantic answer we provided above used mathematical English to do so, and this was no mistake since meanings have to get going somewhere and in this respect $\mathcal{L}^{\text{PL}}$ is no place to begin.

Instead of undercutting the meanings that you understand in English, introducing formal languages provides a way to distill certain elements of meaning that we have reason to care about even if they depart from their correlates in natural language. In analogy, you can think of this like refining the raw materials found in nature into the sorts of materials that are of considerable use to us in constructing the build environment. Rather than the material world, our concern is with the conceptual world, and what we are doing here is a kind of conceptual engineering. Although these are only metaphors, hopefully they will help to shed some light on what we have been doing and will continue to do throughout this course. In particular, it is important to appreciate that English cannot be given up any more than the natural world around us. Accordingly, we will continue to write informal proofs to establish claims about our object language $\mathcal{L}^{\text{PL}}$. Soon we will have an analogue for also proving things *in* $\mathcal{L}^{\text{PL}}$.

Whereas the semantic clauses for $\mathcal{L}^{\text{PL}}$ drew on our grasp of certain elements of mathematical English in order to provide a systematic way to *interpret* the wfss of $\mathcal{L}^{\text{PL}}$, this chapter will also draw on mathematical English in order describe how to *reason* in $\mathcal{L}^{\text{PL}}$. One way to think about our target here is to contemplate the extension of the logical consequence relation $\vDash$. That is, think of the set of ordered pairs which relate any set of wfss of $\mathcal{L}^{\text{PL}}$ to a further wfs of $\mathcal{L}^{\text{PL}}$ where the latter is a logical consequence of the former, or in set notation: $\{\langle \Gamma, \varphi \rangle : \Gamma \vDash \varphi\}$. Needless to say, this is a large, though not unruly space. Although our definition of logical consequence $\vDash$ provides an essential account of this space of logical consequences, it is hard work to check which wfss are logical consequences of which sets of wfss of $\mathcal{L}^{\text{PL}}$. Thus it would be convenient to streamline the process by which we may determine whether $\Gamma \vDash \varphi$.

As with all of our methods, writing formal proofs in $\mathcal{L}^{\text{PL}}$ has its range of natural applications where sometimes it is more trouble than it is worth and a semantic proof would have been better. Nevertheless, convenience is not our only motivation. Rather, what we should like to describe are the most basic inferences that make up the practice of formal reasoning, composing those inferences in order to not only say what follows from what in virtue of logical form, but also how. Of course, not any rational seeming maneuvers ought to be included. In particular, we will require the basic inferences that make up PL to be valid. This is referred to as *soundness*: if φ can be inferred from Γ, then φ is a logical consequence of Γ. However, this is not all of what we want. Rather, we also want our basic inferences to be inherently compelling. Put otherwise, we are looking to find the atoms that make up formal reasoning.

Insofar as the proof systems that we will be concerned with in this course aim to encode the natural patterns of formal reasoning, we will refer to these systems as NATURAL DEDUCTION systems. In particular, PL is a natural deduction system for $\mathcal{L}^{\text{PL}}$ where later chapters will consider a natural deduction system for $\mathcal{L}^{\text{FOL}}$. These systems provide a way to argue from the premises to a conclusion in logically valid ways while resembling natural forms of reasoning. In addition to being familiar, reasoning in this way helps to illustrate the logical connections between various claims in a way that is compelling *on its own terms*. That is, you don't have to take a course in logic or learn the semantics for an artificial language in order to appreciate the patterns of reasoning that we will consider, finding them compelling.

In what follows, we will introduce ten basic derivation rules for the five sentential operators in $\mathcal{L}^{\text{PL}}$. Each operator will have an introduction and an elimination rule which, taken together, describe a certain dimension of the *meaning* of that operator. In particular, the introduction and elimination rules describe how to reason with the operators in $\mathcal{L}^{\text{PL}}$. There is also a rule for introducing assumptions and a trivial rule for reiterating earlier lines of a proof. Given these twelve rules in all, we will be in a position to provide a precise definition of a proof in PL, where it is for this reason that PL is referred to as a proof system, or *logic* for $\mathcal{L}^{\text{PL}}$.

As finite and familiar as all of this will turn out to be, PL may nevertheless be shown to have some remarkable properties. In addition to showing that φ is a logical consequence of Γ whenever φ can be proved from Γ, we will also show that nothing is missing: our proof system is capable of deriving *all* logical consequences whatsoever. That is, PL is *complete* in addition to being *sound*. We will prove these results in later chapters.

3.1 Premises and Assumptions

Before introducing the rules, it will help to get a sense of what PL proofs look like in order to articulate some important constraints on the lines of a proof to which a rule may appeal.

A PL proof begins with a (possibly empty) list of premises, where these will be indicated by writing ':PR' on the right. It is often helpful to include a note of what you intend to derive at this point. For instance, consider the following list of premises:

1	$A \to (B \to C)$	:PR
2	A	:PR

The horizontal line indicates where the premises end and the rest of the derivation begins. For instance, we may apply conditional elimination (discussed below) to derive the following:

1	$A \to (B \to C)$	:PR
2	A	:PR
3	$B \to C$	:1, 2 $\to$E

Note that we appealed to lines 1 and 2 in order to derive line 3, indicating as much in the justification of line 3. If $B \to C$ is all that we wanted to derive, then we would be done. However, suppose that we were to continue by adding a new assumption.

1	$A \to (B \to C)$	:PR
2	A	:PR
3	$B \to C$	:1, 2 $\to$E
4	$\quad B$	:AS

At any point in a proof, we can introduce a new assumption on an indented line and starting a new vertical line. More precisely, consider the *assumption rule* (AS) below:

$\quad \varphi$	:AS

We will refer to the process of adding an assumption as one of OPENING A SUBPROOF. Subproofs are what they sound like: a proof within a proof, starting from a single assumption added anywhere in a proof rather than the premises with which we began the proof. For instance, we might add the following lines to the proof above:

1	$A \to (B \to C)$	:PR
2	A	:PR
3	$B \to C$	:1, 2 →E
4	$\quad B$	:AS
5	$\quad C$	:3, 4 →E
6	$\quad C \wedge A$	:2, 5 ∧I
7	$B \to (C \wedge A)$	:4–6 →I

Line 5 applies conditional elimination (discussed below) on lines 3 and 4, and then line 6 applies conjunction introduction (also discussed below) to lines 2 and 5. We then close the subproof, where this may take place at any point in the subproof by ending the vertical line and stepping back one level of indentation. Once a subproof closes, the lines of that closed subproof are DEAD, and so cannot be appealed to individually. Nevertheless, we may appeal to the subproof in its entirety, where line 7 does just this, using conditional introduction (discussed below) which cites lines 4–6 (note the hyphen in place of the comma).

Every line of a proof that is not dead is referred to as LIVE, where rules that cite individual lines (as opposed to subproofs) can only appeal to lines that are live at that point in the proof. For instance, were we to continue our proof a little further, we could not appeal to lines 4, 5, or 6 since these lines are dead. Thus we stipulate the following restriction:

> CITATION: For a rule to cite a single line, that line must not occur within a subproof that has been closed before the line where the rule is being applied.

Closing a subproof is also called DISCHARGING the assumption of that subproof. Subproofs allow us to think about what we could show if we made a further assumption. Accordingly, we have to be careful to keep track of what assumptions we are making and when it is and is not permitted to appeal to an assumption or the wfs of $\mathcal{L}^{\text{PL}}$ that we can derive from that assumption. Our Fitch-style proof system accomplishes this task graphically by indenting assumptions and drawing vertical lines along the length of the resulting subproof. The details for each rule which makes use of this feature of our proof system will be discussed below, but it is important to have some sense of all of this before introducing the rules.

3.2 Reiteration

The first rule was already eluded to above. Given any wfs of $\mathcal{L}^{\text{PL}}$ on a live line of a proof, the *reiteration rule* (R) allows you to repeat that wfs on a new line.

$\vdots$	$\vdots$	
4	$A \wedge B$	
$\vdots$	$\vdots$	
10	$A \wedge B$	:4 R

Given that we have written '$A \wedge B$' on line 4, we may repeat this wfs at some later line, e.g., line 10. We also add a citation which justifies what we have written. In this case, we write 'R', to indicate that we are using the reiteration rule, and we write '4', to indicate that we have applied it to line 4. Here is the general expression of the reiteration rule R:

m	φ	
	φ	:m R

If φ occurs on any line m within the scope of application, we can reiterate φ, justifying this addition by writing ':m R' to indicate that reiteration was applied to line m. Of course, in an actual proof, the lines are numbered, and so m will take on a numerical value.

Here is an example of three legal applications of rule R followed by an illegal application:

1	P		:PR
2		$\neg Q$	:AS
3		$\neg Q$	:2 R (legal)
4		P	:1 R (legal)
5	$\neg Q \rightarrow P$		:2–4 $\rightarrow$I
6	P		:1 R (legal)
7	$\neg Q$		:2 R, (illegal)

On the second line, we begin a subproof by assuming $\neg Q$. You can reiterate $\neg Q$ within the subproof as in line 3, but not when you leave the subproof as in line 7. On line 4, we reiterate P on line 1, maintaining the indentation of the subproof. We then close the subproof, citing the subproof in line 5. At line 6, we can reiterate line 1 which is live, but at line 7 we cannot appeal to line 2 since this line is now dead. Even if we were to open another subproof, we still could not appeal to line 2. Rather, the lines of a closed subproof are forever dead. Even so, this does not stop us from appealing to the subproof as a whole as we do in line 5.

3.3 Conjunction

Consider the rule for *conjunction introduction* ($\wedge$I):

m	φ	
n	ψ	
	$\varphi \wedge \psi$	$:m, n\ \wedge$I

This rule says that given any wfss φ and ψ of $\mathcal{L}^{\text{PL}}$ on live lines, you may derive their conjunction $\varphi \wedge \psi$. It is worth noting that m and n need not be consecutive lines, nor do they need to appear in the order listed. We require only that each line has been established somewhere in the proof, and that both lines are live at the line in which we are applying the rule.

Whereas conjunction introduction licenses the derivation of a conjunction from any two wfss of $\mathcal{L}^{\text{PL}}$, conjunction elimination lets us do the opposite. Given any live conjunction, we may derive either of its conjuncts. For instance, if $A \wedge (P \vee Q)$ is live, we may derive A, or we may derive $P \vee Q$, but we must choose which. If we wish to derive both, then two applications of the rule is required, though the order does not matter.

Here are the left and right *conjunction elimination* ($\wedge$E) rules:

m	$\varphi \wedge \psi$	
	φ	$:m\ \wedge$E
	ψ	$:m\ \wedge$E

These rules allow us to derive either conjunct. Although we will often end up deriving both conjunts, we need not do so. For instance, this is a perfectly acceptable proof:

1	$A \wedge B$	:PR
2	B	:1 $\wedge$E

Note that the $\wedge$E rule only requires one wfs of $\mathcal{L}^{\text{PL}}$, and so there is only one line number in the justification. In order to see the conjunction rules to work together, consider the argument:

$$\begin{array}{ll} \text{A1.} & \underline{[(A \vee B) \rightarrow (C \vee D)] \wedge [(E \vee F) \rightarrow (G \vee H)]} \\ \text{A2.} & [(E \vee F) \rightarrow (G \vee H)] \wedge [(A \vee B) \rightarrow (C \vee D)] \end{array}$$

The main logical operator in both the premise and conclusion is conjunction. Since conjunction is commutative, the argument is obviously valid since the two conjunctions have the same two conjuncts in the opposite order. In order to provide a proof, we begin by writing down the premise on a numbered line indicating that it is a premise. Since this is the only premise, we draw a horizontal line where everything below this line must be justified by a proof rule.

1	$[(A \vee B) \rightarrow (C \vee D)] \wedge [(E \vee F) \rightarrow (G \vee H)]$	:PR

From the premise, we can separate the conjuncts with $\wedge$E. This yields the following:

1	$[(A \vee B) \rightarrow (C \vee D)] \wedge [(E \vee F) \rightarrow (G \vee H)]$	:PR
2	$[(A \vee B) \rightarrow (C \vee D)]$	:1 $\wedge$E
3	$[(E \vee F) \rightarrow (G \vee H)]$	:1 $\wedge$E

The $\wedge$I rule requires that each of the conjuncts is live somewhere in the proof from the current line, though their order and distance from each other does not matter. By applying the $\wedge$I rule to lines 3 and 2, we may arrive at the desired conclusion.

1	$[(A \vee B) \rightarrow (C \vee D)] \wedge [(E \vee F) \rightarrow (G \vee H)]$	:PR
2	$[(A \vee B) \rightarrow (C \vee D)]$	:1 $\wedge$E
3	$[(E \vee F) \rightarrow (G \vee H)]$	:1 $\wedge$E
4	$[(E \vee F) \rightarrow (G \vee H)] \wedge [(A \vee B) \rightarrow (C \vee D)]$	:3, 2 $\wedge$I

This proof may not look terribly interesting or surprising, but it shows how we can use the proof rules together to demonstrate the validity of an argument. Note that using a truth table to show that this argument is valid would have required a staggering 256 lines, since there are eight sentence letters in the argument. A semantic proof would be less unwieldy, but would not have been as simple or natural of an argument. At the very least, you would have to know a bit about the semantics for our language $\mathcal{L}^{\text{PL}}$. By contrast, the steps in the proof above are already pretty compelling given that '$\wedge$' expresses conjunction.

3.4 Disjunction

Suppose Ludwig is reactionary.[1] Then Ludwig is either reactionary or libertarian. Trivial as this may seem, it speaks to the logic of disjunction. Just as we may derive either conjunct from a conjunction, we may derive a disjunction from either of its disjuncts.

Thus the *disjunction introduction* ($\vee$I) rule may be stated as follows:

m	φ	
	$\varphi \vee \psi$	:m $\vee$I
	$\psi \vee \varphi$	:m $\vee$I

As above, the line m must be live, where we cite this line in the justification of the rule application. Since ψ can be *any* wfs of $\mathcal{L}^{\text{PL}}$, the following is a perfectly acceptable proof:

1	M	:PR
2	$M \vee ([(A \leftrightarrow B) \rightarrow (C \wedge D)] \leftrightarrow [E \wedge F])$	:1 $\vee$I

Using a truth table to show this would have taken 128 lines.

The disjunction elimination rule is slightly trickier. Suppose that either Ludwig is reactionary or he is libertarian. It does not follow that Ludwig is reactionary, for he might be a libertarian. Equally, we cannot conclude that Ludwig is libertarian, since he might be reactionary. Given that we don't know which disjunct is true, it is difficult to deduce anything from a disjunction on its own. The elimination rule for disjunction provides a workaround.

Suppose that we could show that if Ludwig's is reactionary, then he is an Austrian economist. Suppose that we could also show that if Ludwig's is a libertarian, then he is also an Austrian economist. Even though we don't know whether Ludwig is reactionary or a libertarian, it doesn't matter: in either case he is an Austrian economist. This is a natural way to make use of a disjunction even when we don't know which disjunct is true. Indeed, we employed reasoning of this kind in the semantic proof by cases that we gave in Chapter 2. Generalizing on this line of reasoning, consider the following *disjunction elimination* ($\vee$E) rule:

[1] This section has been adapted from the Calgary remix §16.7.

m	$\varphi \vee \psi$	
i	$\quad \varphi$	:AS for $\vee$E
$\vdots$	$\quad \vdots$	
j	$\quad \chi$	
k	$\quad \psi$	:AS for $\vee$E
$\vdots$	$\quad \vdots$	
l	$\quad \chi$	
	χ	:m, i–j, k–l $\vee$E

This rule is somewhat clunkier to write down than our previous rules, but the idea is a natural one. Suppose that we have some disjunction $\varphi \vee \psi$. Suppose that we can also construct two subproofs showing that χ can be derived from the assumption that φ, and that χ can be derived from the assumption that ψ. We can then infer χ from the original disjunction $\varphi \vee \psi$ together with our two subproofs. As usual, there can be as many lines as you like between i and j, and as many lines as you like between k and l. Moreover, the subproofs and the disjunction can come in any order, and do not have to be adjacent to each other as above. Although the lines i–j and k–l belong to closed subproofs and so dead, line m must be live.

Some examples will help illustrate. Consider the following argument:

B1. $(P \wedge Q) \vee (P \wedge R)$

B2. P

A proof might run like this, adding the notes 'for $\vee$E' to improve readability:

1	$(P \wedge Q) \vee (P \wedge R)$	:PR
2	$\quad P \wedge Q$	:AS for $\vee$E
3	$\quad P$	:2 $\wedge$E
4	$\quad P \wedge R$	:AS for $\vee$E
5	$\quad P$	:4 $\wedge$E
6	P	:1, 2–3, 4–5 $\vee$E

Here is a slightly harder example. Consider the following argument:

C1. $\underline{A \wedge (B \vee C)}$
C2. $(A \wedge B) \vee (A \wedge C)$

We may then construct the following proof:

1	$A \wedge (B \vee C)$	:PR
2	A	:1 $\wedge$E
3	$B \vee C$	:1 $\wedge$E
4	$\quad B$	:AS for $\vee$E
5	$\quad A \wedge B$	:2, 4 $\wedge$I
6	$\quad (A \wedge B) \vee (A \wedge C)$	:5 $\vee$I
7	$\quad C$	:AS for $\vee$E
8	$\quad A \wedge C$	:2, 7 $\wedge$I
9	$\quad (A \wedge B) \vee (A \wedge C)$	:8 $\vee$I
10	$(A \wedge B) \vee (A \wedge C)$	:3, 4–6, 7–9 $\vee$E

As natural as the rules may seem in isolation, it is not always obvious how to put them together to get from some premises to a conclusion. Like any skill, the ability to construct PL proofs requires practice. To help, we will cover some strategies for finding proofs at the end of the chapter. Nevertheless, once a natural deduction proof has been constructed, each step is easy to justify, making the derivation in total impervious to doubts. Moreover, this certainty does not stem from any semantic considerations. Rather, the proof rules are directly justified by our intuitive understanding of how to use the sentential operators in our language.

3.5 Conditional Introduction

The rule for conditional introduction has already been used in the examples used to first set up the proof system, and should have felt both compelling an familiar. The idea here is that you help yourself to something that you may not know is true, do some reasoning to arrive at some further claim, then conclude by asserting a conditional claim: if the assumption is true, then the further claim is true. This conclusion follows despite not knowing if the original assumption is true. Here is a concrete example of this type of reasoning:

> Ludwig is reactionary. Therefore if Ludwig is libertarian, then Ludwig is both reactionary and libertarian.

We may regiment this argument as a natural deduction proof by starting with one premise R for ‘Ludwig is reactionary’:

1	R	:PR

We may now make an additional assumption L for ‘Ludwig is libertarian’. In common parlance, we might use the turn of phrase ‘Suppose for the sake of argument that...’, or when writing informal proofs, we might start off with ‘Assume R for conditional proof’. In PL, we will indicate that we are adding an assumption by writing ‘AS’ on the right, where it is often helpful to also include ‘for →Intro’ as a note to yourself or your reader.

1	R	:PR
2	$\quad L$	:AS for →Intro

Note that we are not claiming to have proved L from line 1. Accordingly, we do not write any justification for the additional assumption on line 2. Rather, we have started a new subproof by indenting the wfs L and starting a new vertical line. We have also underlined L since it is playing a role analogous to a premise in our new subproof.

With this extra assumption in place, we are now in a position to use ∧I from before.

1	R	:PR
2	$\quad L$	:AS for →Intro
3	$\quad R \wedge L$	:1, 2 ∧I

Given the assumption L, we have deduced $R \wedge L$. We may now discharge our assumption, closing the subproof and adding an appropriate conditional on the next line.

1	R	:PR
2	$\quad L$	:AS for →Intro
3	$\quad R \wedge L$	:1, 2 ∧I
4	$L \to (R \wedge L)$	:2–3 →I

Whereas the indented subproof carries out reasoning under the assumption of L, line 4 reverts back to our original proof which carries out reasoning under the assumption of our single premise R. Accordingly, we cannot conclude $R \wedge L$ merely under the assumption of R by writing $R \wedge L$ at the original level of indenting. Nevertheless, we can assert the conditional $L \to (R \wedge L)$ as given in 4, justifying this line by referring to the entire subproof rather than to individual lines of our proof. In this case, there are only two lines in the subproof, but in general there may be many more. Even in the case where the subproof only consists of two lines, we must use a hyphen to indicate that we are citing a subproof instead of two lines.

Generalising on this pattern, consider the *conditional introduction* rule ($\to$I):

$$
\begin{array}{r|l|l l}
i & & \underline{\varphi} & :\text{AS} \\
\vdots & & \vdots & \\
k & & \psi & \\
 & \multicolumn{2}{l}{\varphi \to \psi} & :i\text{–}k \ \to\text{I}
\end{array}
$$

By appealing to the subproof as a whole for justification, we may write a conditional in a new line stepping back one level of indentation where the assumption of the subproof occurs as the antecedent of the conditional and the conclusion of the subproof occurs as the consequent. As we will see, knowing what the rule is one thing and knowing when to use it is another.

3.6 Conditional Elimination

Many different arguments demonstrate the classic inference *modus ponens*:

D1. $P \to \neg Q$	E1. $\neg P \to (A \leftrightarrow B)$	F1. $(P \vee Q) \to A$
D2. $\underline{P}$	E2. $\underline{\neg P}$	F2. $\underline{P \vee Q}$
D3. $\neg Q$	E3. $A \leftrightarrow B$	F3. A

The natural deduction system of this chapter will include a rule of inference corresponding to *modus ponens* which goes by the name *conditional elimination* ($\to$E). Here is the rule:

$$
\begin{array}{r|l l}
m & \varphi \to \psi & \\
n & \varphi & \\
 & \psi & :m,\ n \ \to\text{E}
\end{array}
$$

What this rule says is that if you have a conditional $\varphi \to \psi$ on a live line number m, and you also have the antecedent φ of that conditional on a live line n, you can write the consequent ψ on a new line. In order to justify this inference, we will list the line numbers m and n as well as '→E' to specify the rule. Given the conditional elimination rule, we can prove that the arguments given above are valid. Here are proofs of two of them:

1	$P \to \neg Q$	:PR
2	P	:PR
3	$\neg Q$	:1, 2 →E

1	$(P \vee Q) \to A$	:PR
2	$P \vee Q$	:PR
3	A	:1, 2 →E

Notice that these proofs share the same structure. We start by listing the premises followed by a horizontal line, where subsequent lines will need to be derived with the rules. We then apply the conditional elimination rule to get the conclusion, citing the appropriate lines. One can produce more complicated proofs with the same rule.

G1. A
G2. $A \to B$
G3. $B \to C$
G4. $C \to [\neg P \leftrightarrow (Q \vee R)]$
G5. $\neg P \leftrightarrow (Q \vee R)$

We begin by writing our four premises on numbered lines:

1	A	:PR
2	$A \to B$	:PR
3	$B \to C$	:PR
4	$C \to [\neg P \leftrightarrow (Q \vee R)]$	:PR (Want $\neg P \leftrightarrow (Q \vee R)$)

The parenthetical off to the right is optional, but can help to keep track of the conclusion that we are attempting to establish. The proof will be complete once we derive $\neg P \leftrightarrow (Q \vee R)$ by applying the rules to the premises or lines that result from doing so. Since we cannot use conditional elimination to get to our desired conclusion directly from our premises, it is worth considering what we can do. For instance, we can use conditional elimination on lines 1 and 2 to get B on a new line, and then repeat using our new line together with line 3 to get C on yet another new line. Continuing in this manner gives us the following proof:

1	A	:PR
2	$A \to B$	:PR
3	$B \to C$	:PR
4	$C \to [\neg P \leftrightarrow (Q \vee R)]$	:PR want ¬P ↔(Q ∨R)
5	B	:1, 2 →E
6	C	:3, 5 →E
7	$\neg P \leftrightarrow (Q \vee R))$	:4, 6 →E

Having derive line 5 from lines 1 and 2, we may derive 6 from 3 and 5, and then conclude by deriving 7 from 4 and 6. In general, each time that we appeal to earlier lines in a proof in order to apply a rule, we must check to see if those lines are live. However, in this case, we have not introduced any assumptions, and so there is no risk that any lines fail to be live.

In order to see the conditional introduction and elimination rules work together, consider:

H1. $P \to Q$
H2. $\underline{Q \to R}$
H3. $P \to R$

We start by listing the premises— this much is automatic requiring no thinking whatsoever. But now we have to think about where we are going, i.e., we want to conclude with the conditional $P \to R$. A great way to do this is by conditional introduction and so, to use this rule, we must begin by assuming the antecedent P of the conditional we want to conclude.

1	$P \to Q$	:PR
2	$Q \to R$	:PR
3	$\quad P$	:AS for →I

Note that there is nothing preventing us from appealing to P in the course of our subproof since before we have closed the subproof, all of its lines are still live. Thus we have:

1	$P \to Q$	:PR
2	$Q \to R$	:PR
3	$\quad P$	:AS for $\to$I
4	$\quad Q$	:1, 3 $\to$E
5	$\quad R$	:2, 4 $\to$E
6	$P \to R$	:3–5 $\to$I

Whereas line 4 derives Q from lines 1 and 3 by conditional elimination, we may apply the same rule to derive R on line 5 from the lines 2 and 4. Finally, we may close our subproof, and conclude $P \to R$ on line 6 while citing the subproof on lines 3–5. Knowing when exactly to open a subproof can take some practice, but a good rule of thumb is that if you want to establish a conditional either at the end of a proof or along the way, you may well need to assume its antecedent and reason your way to the consequent.

3.7 The Biconditional

The rules for the biconditional will be like double-barrelled versions of the rules for the conditional. In order to prove $F \leftrightarrow G$ you must be able to prove G on the assumption F, and separately, prove F on the assumption G. The *biconditional introduction* rule ($\leftrightarrow$I) therefore requires two subproofs. Schematically, the rule looks like this:

i	$\quad \varphi$	:AS for $\leftrightarrow$I
$\vdots$	$\quad \vdots$	
j	$\quad \psi$	
k	$\quad \psi$	:AS for $\leftrightarrow$I
$\vdots$	$\quad \vdots$	
l	$\quad \varphi$	
	$\varphi \leftrightarrow \psi$	:i–j, k–l $\leftrightarrow$I

There can be as many lines as you like between i and j, and as many lines as you like between k and l. Moreover, the subproofs can come in any order, and the second subproof does not need to come immediately after the first.

The biconditional elimination rule (↔E) lets you do a bit more than the conditional rule. If you have the left-hand subsentence of the biconditional, you can obtain the right-hand subsentence. If you have the right-hand subsentence, you can obtain the left-hand subsentence.

m	$\varphi \leftrightarrow \psi$	
n	φ	
	ψ	:m, n ↔E

Equally, we may work in the reverse direction:

m	$\varphi \leftrightarrow \psi$	
n	ψ	
	φ	:m, n ↔E

Note that in the citation for ↔E, we always cite the biconditional first and either the left or right argument depending as the second argument.

3.8 Negation

Here is a simple mathematical argument:

I1. Assume there is some greatest natural number, call it n.
I2. Now consider its successor $n + 1$ which is also a natural number.
I3. Since $n + 1 > n$, we may conclude that n is not the greatest natural number.
I4. But this contradicts our assumption.
I5. Thus there is no greatest natural number.

We used *reductio* style arguments of this kind in some of the semantic proofs in Chapter 2. The full Latin name *reductio ad absurdum* means "reduction to absurdity." Proofs of this form are also sometimes called *indirect proofs*. A *reductio* argument assumes something which we would like to show is false and aims to derive a contradiction. For instance, we might end up reaching the negation of the *reductio* assumption, or else two wfss of the form ψ and $\neg\psi$. Given such a contradiction, we may assert the negation of the original assumption.

In mathematics, *reductio* arguments often lead to conclusions like $0 = 1$ that contradict something that is already known more generally though the negation $0 \neq 1$ might not show up anywhere in the proof. Whether stated or not, what is going on here is that we really have two contradictory claims: $0 = 1$ and $0 \neq 1$, or to be even more explicit, $\neg(0 = 1)$. Mathematical proofs typically suppress many of the obvious details, and so do not take the form of fully explicit valid arguments of the kind with which we will be concerned.

The negation rules will allow us to write *reductio* style arguments. Like the conditional introduction rule (→I), the negation rules require a new assumption on an indented line, starting a new vertical line. If this assumption can be shown to lead to both a wfs of $\mathcal{L}^{\text{PL}}$ as well as its negation within the course of the subproof, then we may write the negation of the assumption of this subproof on a new line, stepping back one level of indentation. Schematically, this is what the *negation introduction* (¬I) rule looks like:

m	$\vert\ \ \vert\ \varphi$	:AS for $\neg$I
n	$\vert\ \ \vert\ \psi$	
o	$\vert\ \ \vert\ \neg\psi$	
	$\vert\ \neg\varphi$	:m–o $\neg$I

On line m, we assume φ for *reductio.* Our goal is to derive a contradiction, represented by two wfss ψ and $\neg\psi$ of $\mathcal{L}^{\text{PL}}$ on separate lines in any order. Accordingly, it is often convenient to include a note to ourselves and our readers that we are trying to introduce a negation by reaching a contradiction. Observe that ψ could be the same wfs as φ, e.g. both could be P, but this need not always be the case. Once we have derived a contradictory pair of wfss of $\mathcal{L}^{\text{PL}}$, we may close the subproof, moving to the left one level of indentation. We may then write the negation of the assumption in the subproof $\neg\varphi$ on a new line, citing the whole subproof by using a hyphen and indicating the negation introduction rule $\neg$I.

Suppose that we want to derive an instance of the law of non-contradiction: $\neg(G \wedge \neg G)$. A decent rule of thumb is that if you want to conclude a negated wfs of $\mathcal{L}^{\text{PL}}$, it is natural to assume the negand and see if you can reach a contradiction, though this may not always be the best strategy. However, in the case of $\neg(G \wedge \neg G)$, this is just what we will do, starting a subproof by adding the assumption $G \wedge \neg G$ to a proof without any premises.

1	$\quad G \wedge \neg G$	:AS for $\neg$I
2	$\quad G$	:1 $\wedge$E
3	$\quad \neg G$	:1 $\wedge$E
4	$\neg(G \wedge \neg G)$	:1–3 $\neg$I

Although some proofs require some real creativity, this one is pretty obvious once we make the right assumption. After all, the only rule we could apply to our assumption is $\wedge$E, where two applications give us a contradiction. By applying $\neg$I, we may conclude the proof.

The *negation elimination* ($\neg$E) rule works in much the same way. If we assume $\neg\varphi$ and show that it leads to a wfs of $\mathcal{L}^{\text{PL}}$ and its negation, we may conclude φ. So the rule looks like this:

m	$\quad \neg\varphi$	:AS for $\neg$E
n	$\quad \psi$	
o	$\quad \neg\psi$	
	φ	:m–o $\neg$E

As in the case of negation introduction, it is important that the justification of an application of negation elimination cite an entire subproof, indicating as much with a hyphen between the first and last lines. Additionally, it is important that the contradictory pair of wfss of $\mathcal{L}^{\text{PL}}$ occur in the subproof itself rather than elsewhere in the proof. Below is an example which makes an essential appeal to the reiteration rule in order to apply negation elimination:

1	P	:PR
2	$\neg Q \rightarrow \neg P$	:PR want Q
3	$\quad \neg Q$	:AS for $\neg$E
4	$\quad \neg P$	:2, 3 $\rightarrow$E
5	$\quad P$	:1 R
6	Q	:3–5 $\neg$E

Negation elimination requires that one show that some wfs of $\mathcal{L}^{\text{PL}}$ and its negation are derivable given the assumption of a negated wfs of $\mathcal{L}^{\text{PL}}$. In this case, we establish that $\neg P$ follows from the assumption that $\neg Q$ by conditional elimination. Even though P occurs on a live line, we must use the reiteration rule in order to include P in our subproof. Only then may we draw Q as a conclusion by way of negation elimination.

3.9 Proof Strategy

The examples have been relatively simple so far, but perhaps you can already get a sense of the kinds of strategic thinking that natural deduction proofs sometimes require. For instance, although it is always permissible to open a subproof with any assumption, knowing which assumption to introduce and when to do so can require some care. Starting a subproof with any arbitrary assumption may clutter your proof. In order to obtain a conditional by →I, for example, it makes sense to assume the antecedent of the conditional in a subproof and see if you can derive the consequent. This is an example of a good proof strategy.

It is also always permissible to close a subproof, discharging its assumptions. However, it will not be helpful to do so until you have reached something useful. Once the subproof is closed, you can only cite the entire subproof in a justification for a line following that subproof. Those rules that call for a subproof, or multiple subproofs, require that the last line of the subproof is a wfs of $\mathcal{L}^{\text{PL}}$ of some form or other. For instance, you are only allowed to cite a subproof for →I if the line you are justifying is of the form $\varphi \to \psi$ where φ is the assumption of your subproof and ψ is the last line of your subproof. This constrains the strategies that one might hope to employ in attempting to construct proofs in PL.

Getting good at natural deduction will take some practice. The good news is that natural deduction proofs are a lot more interesting to construct than truth tables, and a much more beneficial skill: practicing natural deduction will streamline your reasoning well beyond the scope of this course, where the same cannot be said for filling out arrays with 1s and 0s. Although there are no fail-safe methods, and certainly no substitute for practice, there are some general rules of thumb and strategies that are worth keeping in mind.

Work backwards from what you want. The ultimate goal is to derive the conclusion. Look at the conclusion and ask what the introduction rule is for its main operator. This gives you an idea of what should happen just before the last line of the proof. Then you can treat this line as if it were your goal, asking what you could do to derive this new goal.

For example, if your conclusion is a conditional $\varphi \to \psi$, plan to use the →I rule. This requires starting a subproof in which you assume φ. In the subproof, you want to derive ψ.

Work forwards from what you have. When you are starting a proof, look at the premises and consider what implications they might have, or what you would need to derive in order to make use of the premises. It can help to think about the elimination rules for the main operators of the premises, or the wfss that you have derived so far.

For example, if you have a conditional $\varphi \to \psi$, and you also have φ on a line, →E is a pretty natural move to make. Sometimes it is a lot trickier to know what to do next, but not always.

Repeat as necessary. A long proof is just a number of short proofs linked together, so you can fill the gap by alternately working back from the conclusion and forward from the premises. Once you have decided how you might be able to get to the conclusion, ask what you might be able to do with the premises. Then consider these new targets, asking how you might reach each them, slowly connecting the dots of your proof.

Try a reductio when nothing else works. If you cannot find a way to show something directly, try assuming its negation and see where this leads. Sometimes this can help unlock a proof, perhaps even leading you to a direct line of argument.

Persist. Try different things. If one approach fails, then try something else. In general, there are typically many ways to construct a proof.

Although these guidelines can help if you get stuck, it is worth mentioning some of the mistakes that are easy to make. In particular, we will review some of the common errors that can occur when using subproofs. Consider the following example:

1	A	:PR
2	$\quad \mid B$	:AS for →I
3	$\quad \mid B$	:2 R
4	$B \to B$	:2–3 →I

This is perfectly in keeping with the rules that we have laid down above, and it should not seem particularly surprising. After all, $B \to B$ is a tautology, and so follows from no premises. Thus it is just as easy to derive $B \to B$ from some starting premises.

But now suppose that tried to continue the proof as follows:

1	A	:PR
2	$\quad \mid B$	:AS for →I
3	$\quad \mid B$	:2 R
4	$B \to B$	:2–3 →I
5	B	:3, 4 →E (illegal)

If we were allowed to do this, we could derive any wfs of $\mathcal{L}^{\text{PL}}$ from any other. However, if you tell me that Anne is fast (symbolized by A), we shouldn't be able to conclude that Queen

Boudica stood twenty-feet tall (symbolized by B). Thankfully we are prohibited from making this move since our rules only permit us to draw on live lines.

Once a subproof closes, the wfss of $\mathcal{L}^{\text{PL}}$ in that proof are dead, and so we cannot appeal to them individually at a later point in the proof. This does not mean that we can't appeal to their results, or to the subproof as a whole. For instance, we could appeal to $B \to B$ on a later line since this wfs is live throughout the proof. Indentation has been included in the proof system to help keep track of what we can and cannot appeal to while writing proofs since it is easy to see which subproofs are closed. In particular, once you step back one level of indentation, the indented lines of the subproof above are dead, and so can only be cited by certain rules which appeal to the entire subproof, not any one of its lines.

Once we have started thinking about what we can derive from additional assumptions, nothing stops us from asking what we can derive from adding even more assumptions. Instead of doing this all at once the way that we may begin with many premises, we will do so by opening subproofs within subproofs. For instance, here is a proof of contraposition:

1	$Q \to P$	:PR want $\neg P \to \neg Q$
2	$\quad \neg P$	:AS for $\to$I
3	$\quad\quad Q$	:AS for $\neg$I
4	$\quad\quad P$	:1, 3 $\to$E
5	$\quad\quad \neg P$	:2 R
6	$\quad \neg Q$	:3–5 $\neg$I
7	$\neg P \to \neg Q$	:2–6 $\to$I

Since we can't do anything with a conditional by itself, line 2 introduces the assumption $\neg P$. This is a natural choice given that we want to conclude $\neg P \to \neg Q$. Even so, there is not much more that we can do than before, and so we are forced to introduce yet another assumption Q on line 3. This is also a natural choice given that we would like to conclude $\neg Q$, and we know that we can use $\neg$I to do so if we reach a contradictory pair of wfss of $\mathcal{L}^{\text{PL}}$ from assuming Q. Given our assumptions, we may then derive P in line 4 by appealing to lines 1 and 3, both of which are live. Since line 2 is still live, we may derive $\neg P$ on line 5 by reiteration. Closing the second subproof, we may justify $\neg Q$ on line 6 by citing the lines 3–5. Now can close the first subproof, using these lines to justify $\neg P \to \neg Q$ on line 7.

For contrast, here is a proof where things go awry:

1	A	:PR
2	$\quad B$	:AS for →I
3	$\quad\quad C$	:AS for →I
4	$\quad\quad A \wedge B$	:1, 2 ∧I
5	$\quad C \rightarrow (A \wedge B)$	:3–4 →I
6	$B \rightarrow (C \rightarrow (A \wedge B))$	:2–5 →I
7	$C \rightarrow (A \wedge B)$	:3–4 →I (ILLEGAL)

The problem is that the subproof that began with the assumption C was under the assumption of B on line 2. By lines 6 and 7, we have discharged the assumption B, and so are no longer asking what we could show if we assumed B. Although it was perfectly legitimate to draw this same inference on line 5, by the time we are at line 7 we cannot appeal to lines 2–5.

Here is one further mistake worth watching out for:

1	A	:PR
2	$\quad B$	:AS for →I
3	$\quad\quad C$	:AS for →I
4	$\quad\quad B \wedge C$	:2, 3 ∧I
5	$B \rightarrow (B \wedge C)$	:2–4 →I (ILLEGAL)

Line 5 tries to cite a subproof that begins on line 2 and ends on line 4, but the wfs on line 4 depends not only on the assumption on line 2, but also on another assumption (line 3) which we have not discharged at the end of the subproof. Put otherwise, the subproof which starts by assuming B does not end with a wfs at all, but rather ends with a subproof. Although we can close both subproofs at once, doing so wouldn't be strategic since line 5 cannot then cite lines 2–4 to justify $B \rightarrow (B \wedge C)$ in the manner presented above.

It is worth further stressing the difference between citing a single line and citing a subproof with a further example. In particular, when a rule requires you to cite a subproof, you cannot cite an individual line instead, nor *vice versa*. So for instance, this is incorrect:

1	A	:PR
2	$\quad B$	:AS for $\to$I
3	$\quad\quad C$	:AS for $\to$I
4	$\quad\quad B \wedge C$	:2, 3 $\wedge$I
5	$\quad\quad C$	:4 $\wedge$E
6	$\quad C$	:3–5 R (illegal)
7	$B \to C$	:2–6 $\to$I

Here, we have tried to justify C on line 6 by the reiteration rule, but we have done so by citing the subproof on lines 3–5. Although that subproof could in principle be cited on line 6, the reiteration rule does not permit us to do so. Rather, we could have used $\to$I to derive $C \to C$ while citing that subproof. By contrast, the reiteration rule R requires you to cite an individual line that is live, so citing the entire subproof is not permissible.

However obvious these mistakes may seem, it can be tempting to bend the rules when writing natural deduction proofs. This is like bending the rules while playing chess: you simply are no longer playing chess, but rather moving chess pieces around a boards in a manner that is no longer constrained by the rules of chess, or any other game for that matter. So in writing your own proofs in PL, keep these rules in mind, sticking to them precisely.

3.10 Derivability

Given the natural deduction rules specified above, we may present the following definition:

> A derivation (or proof) of φ from Γ in PL is any finite sequence of wfs of $\mathcal{L}^{\text{PL}}$ ending in φ where every wfs in the sequence is either: (1) a premise in Γ; (2) an assumption which is eventually discharged; or (3) follows from previous lines by a natural deduction rule for PL besides AS.

A wfs φ of $\mathcal{L}^{\text{PL}}$ is derivable (or provable) from Γ in PL, i.e., $\Gamma \vdash \varphi$, just in case there is a natural deduction derivation (proof) of φ from Γ in PL. Whereas logical consequence provides a semantic answer to the question of what follows from what in virtue of logical form by quantifying over interpretations, the derivation relation $\vdash$ provides a purely syntactic answer to this question by specifying which wfs of $\mathcal{L}^{\text{PL}}$ can be written after which in a manner which constitutes a derivation. Perhaps surprisingly, these two relations will be shown to have the same extension, describing formal reasoning in $\mathcal{L}^{\text{PL}}$ in two different ways.

Having defined the derivation relation for PL, we are now in a position to introduce a number of other definitions. Two wfss φ and ψ of $\mathcal{L}^{\text{PL}}$ are INTERDERIVABLE in PL— i.e., $\varphi \dashv\vdash \psi$— just in case both $\varphi \vdash \psi$ and $\psi \vdash \varphi$. Letting BOTTOM (also called the FALSUM) be the arbitrary contradiction $\bot := A \wedge \neg A$, we may take a set Γ of wfss of $\mathcal{L}^{\text{PL}}$ to be INCONSISTENT in PL just in case $\Gamma \vdash \bot$, and CONSISTENT in PL otherwise.

Provability is relative to a proof system. Whereas the meaning of the '$\vdash$' symbol featured in this chapter concerns PL, later we will use the same symbol for the derivation relation in FOL$^{=}$, distinguishing these with subscripts as in '$\vdash_{\text{PL}}$' and '$\vdash_{\text{FOL}^{=}}$' if need be. As with logical consequence, it is often convenient to write '$\varphi_1, \varphi_2, \ldots \vdash \psi$' as a shorthand for $\{\varphi_1, \varphi_2, \ldots\} \vdash \psi$. A wfs φ of $\mathcal{L}^{\text{PL}}$ is a THEOREM of PL just in case $\vdash \varphi$. As with the derivations in PL, it is important to note that the wfss of $\mathcal{L}^{\text{PL}}$ are only theorems relative to a proof system, and so there is no sense in which $A \vee \neg A$ is a theorem full stop. Nevertheless, it is natural to expect $A \vee \neg A$ to be a theorem of most any proof system for $\mathcal{L}^{\text{PL}}$.

In order to show that something is a theorem we have to derive it from no premises. But how could we show that something is *not* a theorem? More generally, how could we show that $\Gamma \nvdash \varphi$? Showing that there is no proof of φ from Γ would seem to require searching the space of all natural deduction proofs, and this is not bound to be easy. For instance, even if you (or a computer program) failed to derive φ from Γ in a thousand different ways, perhaps their is a proof that has not yet been considered. This brings out an important difference between our natural deduction system PL and the truth table method presented above for deciding whether $\Gamma \vDash \varphi$. In particular, PL does not provide an effective procedure for determining whether $\Gamma \vDash \varphi$. However, consider the following:

PL Soundness: If $\Gamma \vdash \varphi$, then $\Gamma \vDash \varphi$.

Given the above, we may show that $\Gamma \nvdash \varphi$ by proving that $\Gamma \nvDash \varphi$ by finding an interpretation $\mathcal{I}$ of $\mathcal{L}^{\text{PL}}$ in which $\mathcal{V}_{\mathcal{I}}(\gamma) = 1$ for all $\gamma \in \Gamma$ but $\mathcal{V}_{\mathcal{I}}(\varphi) = 0$. The reasoning goes by contraposition. After all, if $\Gamma \vdash \varphi$, then $\Gamma \vDash \varphi$ follows by Soundness, and so $\Gamma \vdash \varphi$ cannot be the case so long as we have shown that $\Gamma \nvDash \varphi$. Thus we may conclude that $\Gamma \nvdash \varphi$.

Suppose instead that we want to show that $\Gamma \vdash \varphi$ but cannot seem to find a proof however hard we look. Here is another important principle to which we might appeal:

PL Completeness: If $\Gamma \vDash \varphi$, then $\Gamma \vdash \varphi$.

If we can show that $\Gamma \vDash \varphi$ either by constructing a truth table or writing a semantic proof, Completeness permits us to conclude that $\Gamma \vdash \varphi$, and so we know that there is a derivation of φ from Γ even if we haven't managed to find one. This is different from knowing *how* to derive φ from Γ, but valuable information nonetheless. In order to make use of the principles above, the following two chapters will establish the soundness and completeness of PL.

Chapter 4

The Soundness of PL

Chapter 0 provided an informal account of logic as the study of formal reasoning which was glossed as *what follows from what in virtue of logical form.* Rather than attempting to describe all of what follows from what in English which lacks a precise definition of a grammatical sentence, Chapter 1 introduced an artificial language $\mathcal{L}^{\text{PL}}$ in which we stipulated a definition of the well-formed sentences (wfss) of $\mathcal{L}^{\text{PL}}$. Given the definition of the wfss of $\mathcal{L}^{\text{PL}}$, Chapter 2 defined the interpretations of $\mathcal{L}^{\text{PL}}$ in order to provide a theory of logical consequence $\vDash$ which answered the question of what follows from what in $\mathcal{L}^{\text{PL}}$. Nevertheless, we did not say *how* one wfs follows from a set of wfss in $\mathcal{L}^{\text{PL}}$ since nothing in the theory of logical consequence described how to reason with the wfss of $\mathcal{L}^{\text{PL}}$. Chapter 3 filled this lacuna by identifying a collection of deduction rules which were both basic and natural, allowing us to draw inferences between the wfss of $\mathcal{L}^{\text{PL}}$. By considering any way of chaining together individual applications of these rules into a finite sequence of inferences, we defined what it is to derive a wfs from a set of wfss in $\mathcal{L}^{\text{PL}}$ where the derivation relation $\vdash$ asserts that there is at least one derivation of the wfs on the right from the set of wfss on the left.

Despite providing a completely different account of formal reasoning, Chapter 3 closed by asserting that the derivation relation $\vdash$ and the logical consequence relation $\vDash$ have the same extension. Given the extensional equivalence of these two relations together with the naturalness of the basic rules for PL, we have every right to take PL to provide an adequate natural deduction system for $\mathcal{L}^{\text{PL}}$. In order to establish the extensional equivalence of the derivation and logical consequence relations, this chapter will prove PL Soundness and the next chapter will prove PL Completeness. Although these results may appear to be very similar in form, they differ considerably in significance. For instance, suppose that we had provided a logic X where X Soundness failed to hold. This means that there is some line of reasoning that we could carry out in X, i.e., $\Gamma \vdash_X \varphi$, where the conclusion fails to be a logical consequence of the premises, i.e., $\Gamma \nvDash \varphi$. Assuming that logical consequence provides an accurate guide to *what* follows from what even if it does not say how, we may take a failure of X Soundness to disqualify X from providing an adequate logic for $\mathcal{L}^{\text{PL}}$. Put otherwise, X could not be relied upon to reason in $\mathcal{L}^{\text{PL}}$ since it is possible to begin with certain premises and reason to a conclusion that does not follow as a logical consequence.

Establishing PL Soundness shows that PL can be relied on to reason in $\mathcal{L}^{\text{PL}}$ without ever deriving a conclusion that does not follow as a logical consequence of the premises with which one begins. In order to appreciate the importance of PL Soundness, it is helpful to compare PL Soundness to an analogous property that we may expect any calculator to satisfy. In particular, any calculator must have the property that no matter what arithmetical operations you enter, if it gives you an answer, that answer is guaranteed to be a truth of arithmetic. Otherwise the calculator is not really a calculator at all but rather something more like a magic eight ball, turning up incorrect answers in an unpredictable manner.

Since we should like to be able to rely on PL in order to carry out reasoning in $\mathcal{L}^{\text{PL}}$, it is important to establish PL Soundness. This result belongs to metalogic insofar as it concerns the properties that our logical system PL may be said to possess. Given that our present aim is to show that PL can be relied upon by proving PL Soundness, it does not make sense to to use PL in order to prove PL Soundness since this would beg the question. Put otherwise, we cannot rely on PL to show that PL can be relied upon. Rather, the proofs of metalogic are developed in mathematical English in a similar manner to the semantic proofs that we provided in Chapter 2. That is the proofs in metalogic are informal in contrast to the formal derivations in PL that we presented in Chapter 3.

In order to establish that PL Soundness holds for any set of wfss Γ and wfs φ of $\mathcal{L}^{\text{PL}}$, it is natural to consider an arbitrary Γ and φ for which $\Gamma \vdash \varphi$. It follows from the definition of the derivation relation that there is some PL derivation X where φ is the conclusion and Γ is the set of premises. Despite knowing that there is such a PL derivation as X, we cannot conclude much more than that, and so it is hard to see how we might show that $\Gamma \vDash \varphi$. In particular, we do not know how the derivation X proceeds, and so cannot say which wfss of $\mathcal{L}^{\text{PL}}$ are on which lines of X nor can we appeal to any of their justifications. Although X is finite, we do not know how long X is and so are left contemplating an infinite number of proofs of finite length, any one of which X might be. This is a common predicament.

One thought is to attempt a *reductio* style proof by assuming that $\Gamma \nvDash \varphi$ and attempting to derive a contradiction. Even so, we still do not have much to work with. In particular, we do not know what φ is or what Γ includes, and so the *reductio* assumption is of little help.

In order to overcome these challenges we will employ mathematical induction which uses a recursive strategy for showing that $\Gamma \vDash \varphi$. In particular, we will show that every line of X is a logical consequence of the premises and undischarged assumptions of X at that line. Since the last line cannot have any undischarged assumptions, it follows that the last line of X is a logical consequence of just the premises. Before presenting the details of this proof, the following section will provide a detailed guide for writing clear and concise induction proofs. In addition to helping you to write your own induction proofs, this guide will help you to understand how the proof of PL Soundness works. If you are already familiar with mathematical induction, consider the following section a review.

4.1 Mathematical Induction

Step 1: Whenever a domain of objects has a recursive definition, it is natural to appeal to an induction proof in order to show that every object in that domain has a given property. Accordingly, we must identify the relevant domain of objects and the property which we are attempting to show is had by every object in that domain. In the case of PL SOUNDNESS, we will begin by assuming that $\Gamma \vdash \varphi$, and so there is a PL derivation X of φ from Γ. We will then present an induction argument that each line is a logical consequence of the premises and undischarged assumptions at that line. So the domain of objects in question are the lines of the derivation X of φ from Γ, and the property in question is *being a logical consequence of its premises and undischarged assumptions*. This brings to light what can be one of the trickiest part of an induction proof: not only is it important to accurately identify the relevant domain of objects, the property of interest must also be carefully chosen.

Step 2: We must now provide some way of organizing the domain into a sequence of stages. For instance, if our domain was the set of natural numbers, we might consider their natural ordering where every number in the sequence is followed by its successor. In the case of the PL derivation X, we will consider the sequence of lines that constitute X. Sometimes the ordering is not so obvious, or else one must reconsider the domain of objects such that they may be ordered in an manner which is advantageous.

Step 3: Next we will establish that the first stage has the property in question. This step is often called the *base case* of our induction proof. For instance, we may show that the first line of the derivation X follows from the premises and undischarged assumptions at that first line. Although the base case is often easy— sometimes so obvious it is hard to know what to write— this is not always the true, and so should not be dismissed.

Step 4: We will then help ourselves to an important assumption called the *induction hypothesis*. This assumption can come in both weak and strong varieties. Whereas *weak induction* assumes that the property in question holds for the n-th stage, *strong induction* assumes that the property in question holds for the n-th stage and all previous stages. For instance, below we will make the stronger assumption that every line $k \leqslant n$ is a logical consequence of the premises and undischarged assumptions at k.

Step 5: We will complete the induction proof by showing that the property in question also holds for the $n+1$-th stage. If we can establish this claim, then it follows that the property in question holds for every stage. After all, we have shown that the property in question holds for the first stage, and that if property holds at (or up through) the n-th stage, then it holds for the $n+1$-th stage. In the case of PL SOUNDNESS, we show that the $n+1$-th line of X is a logical consequence of its premises and undischarged assumptions at that line.

This provides the rough outline of proof by induction with some reference to the induction proof for soundness. In actual practice, the hardest part about induction proofs is staying organized and figuring out which properties to focus on, since sometimes you can make things a lot easier by proving something related to what you really want to show.

4.2 Soundness

Although we could attempt to prove PL Soundness in one shot, it is common to break up long proofs into parts by establishing a number of supporting lemmas. In addition to making the over all structure of a proof easier to read, it is common for certain lemmas to be used again and again throughout different parts of a proof, or else in other proofs entirely, thereby reducing redundancy. Were a lemma to have significant and far reaching consequences that are of interest in their own right, we would do better to call it a *proposition* or even a *theorem*. For instance, it would be inappropriate to refer to PL Soundness as a lemma given the significance of this result. Although lemmas can often help to streamline the presentation of a proof, too many lemmas can clutter a proof that would have been better to present all at once. Knowing when to carve off a lemma to establish separately from a proof of primary interest is a skill in its own right, one that takes lots of practice to cultivate.

PL Soundness: Assume that $\Gamma \vdash \varphi$ for an arbitrary set Γ of wfss of $\mathcal{L}^{\text{PL}}$ and wfs φ of $\mathcal{L}^{\text{PL}}$. It follows that there is some PL derivation X of φ from Γ. Letting φ_i be the sentence on the i-th line of the derivation X and Γ_i be the set of premises that occur on any line $j \leqslant i$ of X together with the assumptions that are undischarged at line i, we may prove the following:

Lemma 4.1 (Base Step) $\Gamma_1 \vDash \varphi_1$.

> *Proof:* By the definition of a PL derivation, φ_1 is either a premise, an assumption that is eventually discharged, or follows by one of the natural deduction rules for PL besides AS. Since φ_1 is the first line of the proof, there are no earlier lines to be cited, and so φ_1 is either a premise or an assumption. Either way, $\Gamma_1 = \{\varphi_1\}$ since φ_1 is not discharged at the first line. As a result, $\Gamma_1 \vDash \varphi_1$ is immediate. □

Lemma 4.2 (Induction Step) $\Gamma_{n+1} \vDash \varphi_{n+1}$ if $\Gamma_k \vDash \varphi_k$ for every $k \leqslant n$.

Given the lemmas above, it follows by strong induction that $\Gamma_n \vDash \varphi_n$ for all n. Since every proof is finite in length, there is a last line m of X where $\varphi_m = \varphi$ is the conclusion. By the definition of a PL derivation, we know that every assumption in X is eventually discharged, and so $\Gamma_m = \Gamma$ is the set of premises. Thus we may conclude that $\Gamma \vDash \varphi$. Discharging the assumption that $\Gamma \vdash \varphi$ and generalizing on Γ and φ completes the proof. □

Whereas **Lemma 4.1** is easy to prove, **Lemma 4.2** requires checking that all of the natural deduction rules for PL preserve logical consequence. Since there are twelve rules, this proof will require quite a bit more work. Having presented the over all structure of the proof of PL Soundness, the following section will fill in the missing details by proving **Lemma 4.2**.

4.3 Induction Step

In order to prove **Lemma 4.2**, assume for strong induction that $\Gamma_k \vDash \varphi_k$ for every $k \leqslant n$. It remains to show that $\Gamma_{n+1} \vDash \varphi_{n+1}$. By the definition of a PL derivation, φ_{n+1} is either a premise, assumption that is eventually discharged, or follows by a PL deduction rule besides AS. If φ_{n+1} is a premise, then $\Gamma_{n+1} \vDash \varphi_{n+1}$ for the same reason given in **Lemma 4.1**. Thus it remains to show that $\Gamma_{n+1} \vDash \varphi_{n+1}$ if φ_{n+1} has been justified by a deduction rule for PL. There are twelve rule in all, and so we must check each case. The following subsections will attend to this task, establishing a number of supporting lemmas along the way.

4.3.1 Assumption and Reiteration

Before attending to the introduction and elimination rules for each of the sentential operators included in PL, this section focuses on the assumption and reiteration rules. Whereas the proofs for most of the rules will appeal to the induction hypothesis assumed above, the proof for the assumption rule is an exception, employing the same reasoning given in **Lemma 4.1**.

Rule 1 (AS) $\Gamma_{n+1} \vDash \varphi_{n+1}$ if φ_{n+1} is justified by AS.

Proof: Assume that φ_{n+1} is justified by AS. Since φ_{n+1} is an undischarged assumption at line $n+1$, it follows from the definition of Γ_{n+1} that $\varphi_{n+1} \in \Gamma_{n+1}$, and so $\Gamma_{n+1} \vDash \varphi_{n+1}$ follows immediately. □

The proof above does not require the induction hypothesis or any additional results. By contrast, it will help to establish the reiteration rule by first proving the following lemma.

Lemma 4.3 (Inheritance) If φ_k is live at line $n \geqslant k$ of a PL derivation, then $\Gamma_k \subseteq \Gamma_n$.

Proof: Let X be a PL derivation where Γ_k is the set of premises and undischarged assumptions at line k. Assume there is some $\psi \in \Gamma_k$ where $\psi \notin \Gamma_n$ for $n > k$. It follows that ψ has been discharged at a line $j > k$ where $j \leqslant n$, and so φ_k is dead at n. By contraposition, if φ_k is live at line $n > k$, then $\Gamma_k \subseteq \Gamma_n$ as desired. □

Although the proof is short, the lemma above makes an important observation about how the undischarged assumptions of live lines are inherited. As we will see, this lemma plays a critical role throughout many of the following proofs and so is important to understand. Given this lemma, we may now move to establish the reiteration rule R.

Rule 2 (R) $\Gamma_{n+1} \vDash \varphi_{n+1}$ if φ_{n+1} is justified by R.

Proof: Assume that φ_{n+1} is justified by R. It follows that $\varphi_{n+1} = \varphi_k$ for some $k \leqslant n$, and so $\Gamma_k \vDash \varphi_k$ by hypothesis. Since φ_k is live at line $n+1$, $\Gamma_k \subseteq \Gamma_{n+1}$ by **Lemma 4.3**, and so $\Gamma_{n+1} \vDash \varphi_k$ by **Lemma 2.1**. Thus $\Gamma_{n+1} \vDash \varphi_{n+1}$.

By contrast with the assumption rule, the reiteration makes an essential appeal to the induction hypothesis. We will see something similar in all of the rule proofs given below.

4.3.2 Negation Rules

The negation rules are much more complicated than the assumption and reiteration rules on account of citing subproofs rather than individual lines. Accordingly, it will help to establish two supporting lemmas before presenting the proofs for the negation rules. Whereas the first lemma asserts that a satisfiable set of wfss of $\mathcal{L}^{\text{PL}}$ cannot have both a wfs of $\mathcal{L}^{\text{PL}}$ and its negation logical consequences, the second lemma draws a connection between logical consequence and unsatisfiability. These lemmas work nicely together and will reoccur in a number of rule proofs besides the negation rule proofs given below.

Lemma 4.4 If $\Gamma \vDash \varphi$ and $\Gamma \vDash \neg\varphi$, then Γ is unsatisfiable.

Proof: Assume $\Gamma \vDash \varphi$ and $\Gamma \vDash \neg\varphi$. Assume for contradiction that Γ is satisfiable, and so there is some $\mathcal{L}^{\text{PL}}$ interpretation $\mathcal{I}$ where $\mathcal{V}_{\mathcal{I}}(\gamma) = 1$ for all $\gamma \in \Gamma$. By assumption, it follows that $\mathcal{V}_{\mathcal{I}}(\varphi) = 1$ and $\mathcal{V}_{\mathcal{I}}(\neg\varphi) = 1$. By the semantics for negation, $\mathcal{V}_{\mathcal{I}}(\varphi) \neq 1$, contradicting the above. Thus Γ is unsatisfiable. □

Lemma 4.5 If $\Gamma \cup \{\varphi\}$ is unsatisfiable, then $\Gamma \vDash \neg\varphi$.

Proof: Assume $\Gamma \cup \{\varphi\}$ is unsatisfiable and let $\mathcal{I}$ be an arbitrary $\mathcal{L}^{\text{PL}}$ interpretation where $\mathcal{V}_{\mathcal{I}}(\gamma) = 1$ for all $\gamma \in \Gamma$. Assume for contradiction that $\mathcal{V}_{\mathcal{I}}(\neg\varphi) = 0$. It follows that $\mathcal{V}_{\mathcal{I}}(\varphi) = 1$, and so $\Gamma \cup \{\varphi\}$ is satisfiable contrary to assumption. Thus $\mathcal{V}_{\mathcal{I}}(\neg\varphi) = 1$. Generalizing on $\mathcal{I}$, it follows that $\mathcal{V}_{\mathcal{I}}(\neg\varphi) = 1$ for any $\mathcal{I}$ where $\mathcal{V}_{\mathcal{I}}(\gamma) = 1$ for all $\gamma \in \Gamma$. By definition, $\Gamma \vDash \neg\varphi$. □

Given the lemmas above, we may provide the following negation rule proofs. It will be important to study this proof carefully, observing how all the working parts come together.

> **Rule 3 ($\neg$I)** $\Gamma_{n+1} \vDash \varphi_{n+1}$ if φ_{n+1} is justified by $\neg$I.

Proof: Assume that φ_{n+1} follows by $\neg$I. Thus there is some subproof on lines i-j where $i < j \leqslant n$ and $\varphi_{n+1} = \neg\varphi_i$, $\psi = \varphi_h$, and $\neg\psi = \varphi_k$ for $i \leqslant h \leqslant j$ and $i \leqslant k \leqslant j$. By parity of reasoning, we may assume that $h < k = j$. Thus we may represent the subproof as follows:

$$
\begin{array}{l|l|ll}
i & & \varphi & \text{:AS for } \neg\text{I} \\
\cline{3-3}
h & & \psi & \\
j & & \neg\psi & \\
n+1 & \neg\varphi & & :i\text{–}j\ \neg\text{I}
\end{array}
$$

By hypothesis, $\Gamma_h \vDash \psi$ and $\Gamma_j \vDash \neg\psi$. With the exception of $\varphi_i = \varphi$, every assumption that is undischarged at lines h and j are also undischarged at line $n+1$. It follows that $\Gamma_h, \Gamma_j \subseteq \Gamma_{n+1} \cup \{\varphi_i\}$, and so $\Gamma_{n+1} \cup \{\varphi_i\} \vDash \psi$ and $\Gamma_{n+1} \cup \{\varphi_i\} \vDash \neg\psi$ by **Lemma 2.1**. By **Lemma 4.4**, $\Gamma_{n+1} \cup \{\varphi_i\}$ is unsatisfiable, and so $\Gamma_{n+1} \vDash \neg\varphi_i$ by **Lemma 4.5**. Equivalently, $\Gamma_{n+1} \vDash \varphi_{n+1}$. $\square$

The proof begins by assuming φ_{n+1} follows from Γ_{n+1} by negation introduction $\neg$I and unpacking the consequences. This provides a number of details about the proof that are required for $\neg\varphi$ to be derived on line $n+1$ by $\neg$I. In particular, we know that ψ and $\neg\psi$ must occur on earlier lines in a subproof. After appealing to the induction hypothesis to conclude $\Gamma_h \vDash \psi$ and $\Gamma_j \vDash \neg\psi$, the proof observes that although φ has been discharged by line $n+1$, this is the only difference between the sets of undischarged sentences for lines i–j and line $n+1$. Thus the logical consequences $\Gamma_h \vDash \psi$ and $\Gamma_j \vDash \neg\psi$ may be related to the undischarged assumptions at line $n+1$ together with the assumption φ which has been discharged at $n+1$. The core of the proof follows from the two lemmas given above which show that the undischarged assumptions at $n+1$ together with φ are unsatisfiable, and so $\neg\varphi$ is a logical consequences of those undischarged assumptions.

Before moving on to consider the rest of the rule proofs, it can help to try writing the proof for yourself. You might also give the following proof a try which works in a similar manner. Getting a good understanding of how these proofs work will make reading the rest of the proofs in this chapter a lot easier and more meaningful than they would be otherwise.

> **Rule 4 ($\neg$E)** $\Gamma_{n+1} \vDash \varphi_{n+1}$ if φ_{n+1} is justified by $\neg$E.

Proof: This proof is left as an exercise for the reader. $\square$

4.3.3 Conjunction and Disjunction

Whereas the rule proofs given above for negation drew on two lemmas established for just this purpose, the rule proof for conjunction introduction is straightforward:

Rule 5 (&I) $\Gamma_{n+1} \vDash \varphi_{n+1}$ if φ_{n+1} is justified by $\wedge$I.

Proof: Assume that φ_{n+1} is justified by $\wedge$I. Thus $\varphi_{n+1} = \varphi_i \wedge \varphi_j$ for some lines $i, j \leqslant n$ that are live at line $n+1$. By hypothesis, $\Gamma_i \vDash \varphi_i$ and $\Gamma_j \vDash \varphi_j$ where $\Gamma_i, \Gamma_j \subseteq \Gamma_{n+1}$ by **Lemma 4.3**. Thus $\Gamma_{n+1} \vDash \varphi_i$ and $\Gamma_{n+1} \vDash \varphi_j$ by **Lemma 2.1**. Letting $\mathcal{I}$ be an arbitrary $\mathcal{L}^{\text{PL}}$ interpretation where $\mathcal{V}_{\mathcal{I}}(\gamma) = 1$ for all $\gamma \in \Gamma_{n+1}$, it follows that $\mathcal{V}_{\mathcal{I}}(\varphi_i) = \mathcal{V}_{\mathcal{I}}(\varphi_j) = 1$ and so $\mathcal{V}_{\mathcal{I}}(\varphi_i \wedge \varphi_j) = \mathcal{V}_{\mathcal{I}}(\varphi_{n+1}) = 1$. By generalizing on $\mathcal{I}$, we may conclude that $\Gamma_{n+1} \vDash \varphi_{n+1}$. □

In addition to drawing on the induction hypothesis, the proof above makes an essential appeal to the semantic clause for conjunction. The rule proofs for conjunction elimination and disjunction introduction work by similar reasoning, and so have been left as exercises.

Rule 6 (&E) $\Gamma_{n+1} \vDash \varphi_{n+1}$ if φ_{n+1} is justified by $\wedge$E.

Proof: This proof is left as an exercise for the reader. □

Rule 7 (∨I) $\Gamma_{n+1} \vDash \varphi_{n+1}$ if φ_{n+1} follows from Γ_{n+1} by the rule $\vee$I.

Proof: This proof is left as an exercise for the reader. □

Given the induction hypothesis, the rule proofs above amount to little more than applications of the semantic clauses for conjunction and disjunction respectively. Something similar may be said for the rule proof for disjunction elimination though a little more care is required to keep track of all of the moving parts, and so the details have been provided in full.

Rule 8 (∨E) $\Gamma_{n+1} \vDash \varphi_{n+1}$ if φ_{n+1} follows from Γ_{n+1} by the rule ∨E.

Proof: Assume that φ_{n+1} is justified by ∨I. Thus there is some $\varphi_i = \varphi_j \vee \varphi_h$ which is live at $n+1$ and subproofs on lines j-h and k-l where $i < j, k, h, l \leqslant n$ and $\varphi_k = \varphi_l = \varphi_{n+1}$. By parity of reasoning, we represent the proof as follows:

i	$\varphi \vee \psi$	
j	$\quad \varphi$	:AS for ∨E
k	$\quad \chi$	
h	$\quad \psi$	:AS for ∨E
l	$\quad \chi$	
$n+1$	χ	:i, j–k, h–l ∨E

By hypothesis, $\Gamma_i \vDash \varphi_i$, $\Gamma_k \vDash \varphi_k$, and $\Gamma_l \vDash \varphi_l$. Given **Lemma 4.3**, $\Gamma_i \subseteq \Gamma_{n+1}$, and so $\Gamma_{n+1} \vDash \varphi_i$ by **Lemma 2.1**. With the exception of $\varphi_j = \varphi$, every assumption that is undischarged at line k is also undischarged at line $n+1$, and so $\Gamma_k \subseteq \Gamma_{n+1} \cup \{\varphi_j\}$. By the similar reasoning, we may conclude that $\Gamma_l \subseteq \Gamma_{n+1} \cup \{\varphi_h\}$, and so $\Gamma_{n+1} \cup \{\varphi_j\} \vDash \varphi_k$ and $\Gamma_{n+1} \cup \{\varphi_h\} \vDash \varphi_l$ follows by **Lemma 2.1**.

Letting $\mathcal{I}$ be an arbitrary $\mathcal{L}^{\text{PL}}$ interpretation where $\mathcal{V}_{\mathcal{I}}(\gamma) = 1$ for all $\gamma \in \Gamma_{n+1}$, it follows from the above that $\mathcal{V}_{\mathcal{I}}(\varphi_i) = \mathcal{V}_{\mathcal{I}}(\varphi_j \vee \varphi_h) = 1$, and so either $\mathcal{V}_{\mathcal{I}}(\varphi_j) = 1$ or $\mathcal{V}_{\mathcal{I}}(\varphi_h) = 1$ by the semantics for disjunction.

Case 1: If $\mathcal{V}_{\mathcal{I}}(\varphi_j) = 1$, then $\mathcal{V}_{\mathcal{I}}(\gamma) = 1$ for all $\gamma \in \Gamma_{n+1} \cup \{\varphi_j\}$, and so $\mathcal{V}_{\mathcal{I}}(\varphi_k) = 1$ since $\Gamma_{n+1} \cup \{\varphi_j\} \vDash \varphi_k$. Thus $\mathcal{V}_{\mathcal{I}}(\varphi_{n+1}) = 1$ since $\varphi_k = \varphi_{n+1}$.

Case 2: If $\mathcal{V}_{\mathcal{I}}(\varphi_h) = 1$, then $\mathcal{V}_{\mathcal{I}}(\gamma) = 1$ for all $\gamma \in \Gamma_{n+1} \cup \{\varphi_h\}$, and so $\mathcal{V}_{\mathcal{I}}(\varphi_l) = 1$ since $\Gamma_{n+1} \cup \{\varphi_h\} \vDash \varphi_l$. Thus $\mathcal{V}_{\mathcal{I}}(\varphi_{n+1}) = 1$ since $\varphi_l = \varphi_{n+1}$.

In either case, $\mathcal{V}_{\mathcal{I}}(\varphi_{n+1}) = 1$, and so $\Gamma_{n+1} \vDash \varphi_{n+1}$ by generalizing on $\mathcal{I}$. □

As with the previous two deduction rules for conjunction and disjunction, the proof above turns on little more than an application of the semantics for disjunction given the induction hypothesis. Nevertheless, it is very easy for parts to become tangled and a lot of care is required to write a proof that is both clear and concise for your reader.

4.3.4 Conditional Rules

In order to streamline the rule proof for →I, it will help to prove the following.

Lemma 4.6 If $\Gamma \cup \{\varphi\} \vDash \psi$, then $\Gamma \vDash \varphi \to \psi$.

Proof: Assume $\Gamma \cup \{\varphi\} \vDash \psi$ and let $\mathcal{I}$ be an arbitrary $\mathcal{L}^{\text{PL}}$ interpretation where $\mathcal{V}_{\mathcal{I}}(\gamma) = 1$ for all $\gamma \in \Gamma$. Since $\mathcal{V}_{\mathcal{I}}(\varphi) = 1$ or not, there are two cases to consider.

Case 1: If $\mathcal{V}_{\mathcal{I}}(\varphi) = 1$, then $\mathcal{V}_{\mathcal{I}}(\gamma) = 1$ for all $\gamma \in \Gamma \cup \{\varphi\}$, and so $\mathcal{V}_{\mathcal{I}}(\psi) = 1$ given the starting assumption. Thus $\mathcal{V}_{\mathcal{I}}(\varphi \to \psi) = 1$ by the semantics for the conditional.

Case 2: If $\mathcal{V}_{\mathcal{I}}(\varphi) \neq 1$, then $\mathcal{V}_{\mathcal{I}}(\varphi \to \psi) = 1$ by the semantics for the conditional.

Since $\mathcal{V}_{\mathcal{I}}(\varphi \to \psi) = 1$ in both cases, $\Gamma \vDash \varphi \to \psi$ follows by generalizing on $\mathcal{I}$. □

Rule 9 (→I) $\Gamma_{n+1} \vDash \varphi_{n+1}$ if φ_{n+1} is justified by →I.

Proof: Assume that φ_{n+1} is justified by →I. Thus there is a subproof on lines i-j where $i < j \leqslant n$ and $\varphi_{n+1} = \varphi_i \to \varphi_j$. We may represent the subproof as follows:

i	φ	:AS for →I
j	ψ	
$n+1$	$\varphi \to \psi$	:i–j →I

By hypothesis, we know that $\Gamma_j \vDash \varphi_j$. With the exception of φ_i, every assumption that is undischarged at line j is also undischarged at line $n+1$. It follows that $\Gamma_j \subseteq \Gamma_{n+1} \cup \{\varphi_i\}$, and so $\Gamma_{n+1} \cup \{\varphi_i\} \vDash \varphi_j$ by **Lemma 2.1**. Thus $\Gamma_{n+1} \vDash \varphi_i \to \varphi_j$ by **Lemma 4.6**. Equivalently, $\Gamma_{n+1} \vDash \varphi_{n+1}$. □

Whereas the proof above appealed to **Lemma 4.6**, the following proof proceeds in a similar manner to the proofs given above, and so the details have been left as an exercise.

Rule 10 (→E) $\Gamma_{n+1} \vDash \varphi_{n+1}$ if φ_{n+1} is justified by →E.

Proof: This proof is left as an exercise for the reader. □

Rule 11 (↔I) $\Gamma_{n+1} \vDash \varphi_{n+1}$ if φ_{n+1} is justified by ↔I.

Proof: Assume φ_{n+1} is justified by $\leftrightarrow$E. Thus there are some subproofs on lines i-j and h-k for some $i < j \leqslant n$ and $h < k \leqslant n$ where $\varphi_i = \varphi_k = \varphi$, $\varphi_j = \varphi_h = \psi$, and either $\varphi_{n+1} = \varphi \leftrightarrow \psi$ or $\varphi_{n+1} = \psi \leftrightarrow \varphi$. By parity of reasoning, we may assume that $\varphi_{n+1} = \varphi \leftrightarrow \psi$. Thus we have:

i	$\quad\varphi$	:AS for $\vee$E
j	$\quad\psi$	
h	$\quad\psi$	:AS for $\vee$E
k	$\quad\varphi$	
$n+1$	$\varphi \leftrightarrow \psi$	:i–j, h–k $\leftrightarrow$I

By hypothesis, $\Gamma_j \vDash \varphi_j$, $\Gamma_k \vDash \varphi_k$, and $\Gamma_{n+1} \vDash \varphi_{n+1}$. With the exception of φ_i, every assumption that is undischarged at line j is also undischarged at line $n+1$, and so $\Gamma_j \subseteq \Gamma_{n+1} \cup \{\varphi_i\}$. Similarly, we may conclude that $\Gamma_k \subseteq \Gamma_{n+1} \cup \{\varphi_h\}$, and so $\Gamma_{n+1} \cup \{\varphi_i\} \vDash \varphi_j$ and $\Gamma_{n+1} \cup \{\varphi_h\} \vDash \varphi_k$ by **Lemma 2.1**.

Let $\mathcal{I}$ be an arbitrary $\mathcal{L}^{\text{PL}}$ interpretation where $\mathcal{V}_\mathcal{I}(\gamma) = 1$ for all $\gamma \in \Gamma_{n+1}$. Assuming $\mathcal{V}_\mathcal{I}(\varphi) = 1$, it follows that $\mathcal{V}_\mathcal{I}(\gamma) = 1$ for all $\gamma \in \Gamma_{n+1} \cup \{\varphi_i\}$, and so $\mathcal{V}_\mathcal{I}(\varphi_j) = 1$ given that $\Gamma_{n+1} \cup \{\varphi_i\} \vDash \varphi_j$. Thus $\mathcal{V}_\mathcal{I}(\psi) = 1$. Assuming instead that $\mathcal{V}_\mathcal{I}(\psi) = 1$, it follows that $\mathcal{V}_\mathcal{I}(\gamma) = 1$ for all $\gamma \in \Gamma_{n+1} \cup \{\varphi_h\}$, and so $\mathcal{V}_\mathcal{I}(\varphi_k) = 1$ given that $\Gamma_{n+1} \cup \{\varphi_h\} \vDash \varphi_k$. Thus $\mathcal{V}_\mathcal{I}(\varphi) = 1$. We may then conclude that $\mathcal{V}_\mathcal{I}(\varphi) = 1$ if and only if $\mathcal{V}_\mathcal{I}(\psi) = 1$, and so $\mathcal{V}_\mathcal{I}(\varphi) = \mathcal{V}_\mathcal{I}(\psi)$. By the semantics for the biconditional, $\mathcal{V}_\mathcal{I}(\varphi \leftrightarrow \psi) = 1$, and so $\Gamma_{n+1} \vDash \varphi_{n+1}$ by generalizing on $\mathcal{I}$. □

Rule 12 ($\leftrightarrow$E) $\Gamma_{n+1} \vDash \varphi_{n+1}$ if φ_{n+1} is justified by $\leftrightarrow$E.

Proof: This proof is left as an exercise for the reader. □

Given **Rule 1** – **Rule 12**, it follows that $\Gamma_{n+1} \vDash \varphi_{n+1}$ no matter how φ_{n+1} has been derived, thereby completing the proof of **Lemma 4.2** as well as the proof of PL SOUNDNESS. Accordingly, we may carry out reasoning in PL while remaining confident that the our derivations preserve logical consequence, and so there is no risk of using PL to reason from some premises to a conclusion that does not follow as a logical consequence.

4.4 Derived Rules

Having established PL SOUNDNESS, we may now proceed to put this theorem to work. In particular, we may explore the range of logical consequences without having to write semantic proofs. Rather, we can use PL in order to write derivations where each conclusion follows as a logical consequences from its premises given PL Soundness.

Suppose that we have managed to construct a PL derivation, for instance that $\neg A \vdash \neg(A \wedge B)$. Even though this derivation is written in terms of particular wfss of $\mathcal{L}^{\text{PL}}$, we could have written a similar derivation by substituting any wfss of $\mathcal{L}^{\text{PL}}$ for 'A' and 'B'. Thus instead of merely asserting that $\neg A \vdash \neg(A \wedge B)$, we may wish to assert the schema $\neg\varphi \vdash \neg(\varphi \wedge \psi)$ where φ and ψ are any wfss of $\mathcal{L}^{\text{PL}}$ whatsoever. More generally, given any particular derivation, we may assert a generalization by replacing the sentence letters with schematic variables, referring to the result as a RULE SCHEMA or, what is also often called a DERIVED RULE.

The reason it makes sense to refer to the schematic generalizations of particular derivations as *rules* at all is that although they have not been included as basic rules of the proof system PL, they may be used in much the same way as the basic rules are used. This is because anything that can be proven with a derived rule can also be proven using just the basic rules included in PL. Accordingly, we may think of the derived rules as abbreviating subroutines which only appeal to the basic rules of PL. Derived rules can then be used to shorten proofs, making some proofs easier to write and more intuitive to read.

Given PL Soundness, derived rules may also be used to indicate logical consequences that are of interest, bringing the vast range of logical consequences that there are into better view. Nevertheless, little is to be gained be restating every derived rule of the form $\Gamma \vdash \varphi$ as a logical consequence of the form $\Gamma \vDash \varphi$. Rather, this much is understood given PL Soundness. Moreover, as brought out above, soundness is an absolutely essential property of any proof system of interest, and so it goes without saying that the derivations in a proof system indicate a corresponding range of logical consequences.

With these general points in order, we may now turn to provide a range of derived rules in PL. Despite being derived rather than basic, many of the derived rules will look familiar, capturing standard ways of reasoning. In addition to shedding light on the logical consequence relation for $\mathcal{L}^{\text{PL}}$, these rules will help to write tricky proofs since they may be cited much like the basic rules, vastly simplifying otherwise lengthy derivations within PL.

4.4.1 *Modus Tollens*

Modus tollens is an extremely important and common inference rule in ordinary reasoning. Here is the derived rule for *modus tollens* (MT):

$$
\begin{array}{l|ll}
m & \varphi \to \psi & \\
n & \neg\psi & \\
 & \neg\varphi & :m,\ n\ \text{MT}
\end{array}
$$

If you have a conditional on one numbered line and the negation of its consequent on another line, you may derive the negation of its antecedent on a new line. We abbreviate the justification for this rule as 'MT' for *modus tollens*. For instance, if you know that if Sue found the treasure, then she is happy, and you also know that Sue isn't happy, then you can infer that Sue didn't find the treasure. Inferences of this form should feel familiar.

In order to derive MT from our basic rules, we will construct a derivation in the manner above while using schematic variables instead of wfss of $\mathcal{L}^{\text{PL}}$. Consider the following:

1	φ	
2	$\psi \to \neg\varphi$	(Want: $\neg\psi$)
3	$\quad \mid \psi$	:AS for $\neg$I
4	$\quad \mid \neg\varphi$	:2, 3 $\to$E
5	$\quad \mid \varphi$	:1 R
6	$\neg\psi$	:3–5 $\neg$I

Since φ and ψ are schematic variables, the lines above do not constitute a PL derivation. Rather, what we have above is a DERIVATION SCHEMA which is a kind of recipe for constructing derivations. Given any wfss φ and ψ, the derivation schema for MT returns a PL derivation as an instance. Accordingly, applications of MT can always be replaced with an appropriate instance of the derivation schema for MT which only refers to the basic rules included in PL. Nevertheless, MT is a convenient shortcut and so we will add it to our list of derived rules.

Here is a simple example that would have been much more cumbersome without using MT:

1	$A \to B$	
2	$B \to C$	
3	$C \to D$	
4	$\neg D$	(Want: $\neg A$)
5	$\neg C$	:3, 4 MT
6	$\neg B$	:2, 5 MT
7	$\neg A$	:1, 6 MT

4.4.2 Dilemma

One of the most difficult deduction rules to apply is disjunction elimination, and so it will be convenient to derive deduction rules that streamline arguments from disjunctive sentences. Consider the *dilemma rule* (DL):

m	$\varphi \vee \psi$	
n	$\varphi \to \chi$	
o	$\psi \to \chi$	
	χ	:m, n, o DL

If you know that two conditionals are true, and they have the same consequent, and you also know that one of the two antecedents is true, then the conclusion is true no matter which antecedent is true. We may derive this rule as follows:

1	$\varphi \vee \psi$	
2	$\varphi \to \chi$	
3	$\psi \to \chi$	want χ
4	$\quad \varphi$	:AS
5	$\quad \chi$	:2, 4 $\to$E
6	$\quad \psi$	:AS
7	$\quad \chi$	:3, 6 $\to$E
8	χ	:1, 4–5, 6–7 $\vee$E

Whereas $\vee$E cites subproofs, DL only appeals to live lines in a proof, and so may be easier to apply in certain contexts. For example, suppose you know all of the following:

A1. If it is raining, the car is wet.
A2. If it is snowing, the car is wet.
A3. It is raining or it is snowing.

From these premises, you can definitely establish that the car is wet. This is an example of the argument form that DL captures, nicely describing a common way of reasoning.

As in the case of MT, the DL rule doesn't allow us to prove anything we couldn't prove via basic rules. Anytime you wanted to use the DL rule, you could always include a few extra steps to prove the same result without DL. Nevertheless, DL captures an natural form of reasoning in its own right, and so is well worth including in our stock of derived rules.

4.4.3 Disjunctive Syllogism

Although DL is occasionally useful, there other common forms of reasoning from a disjunction which DL does not capture. In particular, consider the following argument.

B1. $P \vee Q$
B2. $\neg P$
B3. Q

Even small children and non-human animals can engage in reasoning of the form given above. For instance, if a ball is under one of two cups but you don't know which, and then it is revealed that it is not under one of the cups, it is natural to conclude that the ball must be under the other cup. This inference is called *disjunctive syllogism* (DS):

$$
\begin{array}{l|ll}
m & \varphi \vee \psi & \\
n & \neg\psi & \\
 & \varphi & :m,\ n \text{ DS}
\end{array}
\qquad\qquad
\begin{array}{l|ll}
m & \varphi \vee \psi & \\
n & \neg\varphi & \\
 & \psi & :m,\ n \text{ DS}
\end{array}
$$

We represent two different inference patterns here, because the rule allows you to conclude *either* disjunct from the negation of the other. Nevertheless, both go by the same name as is the case for other symmetrical rules like $\wedge$E. The derivations for DS go as follows:

$$
\begin{array}{l|lll}
1 & \varphi \vee \psi & & \\
2 & \neg\varphi & & \\
\hline
3 & \quad \varphi & & :\text{AS for } \vee\text{E} \\
4 & \quad\quad \neg\psi & & :\text{AS for } \neg\text{E} \\
5 & \quad\quad \neg\varphi & & :2 \text{ R} \\
6 & \quad\quad \varphi & & :3 \text{ R} \\
7 & \quad \psi & & :4\text{–}6\ \neg\text{I} \\
8 & \quad \psi & & :\text{AS for } \vee\text{E} \\
9 & \quad \psi & & :8 \text{ R} \\
10 & \psi & & :1,\ 3\text{–}7,\ 8\text{–}9\ \vee\text{E}
\end{array}
\qquad\qquad
\begin{array}{l|lll}
1 & \varphi \vee \psi & & \\
2 & \neg\psi & & \\
\hline
3 & \quad \psi & & :\text{AS for } \vee\text{E} \\
4 & \quad\quad \neg\varphi & & :\text{AS for } \neg\text{E} \\
5 & \quad\quad \neg\psi & & :2 \text{ R} \\
6 & \quad\quad \psi & & :3 \text{ R} \\
7 & \quad \varphi & & :4\text{–}6\ \neg\text{I} \\
8 & \quad \varphi & & :\text{AS for } \vee\text{E} \\
9 & \quad \varphi & & :8 \text{ R} \\
10 & \varphi & & :1,\ 3\text{–}7,\ 8\text{–}9\ \vee\text{E}
\end{array}
$$

Like DL, the derived rule DS makes it easier to write derivations while capturing a natural way of reasoning. In order to put DS to work, consider the following derivation:

1	$\neg L \to (J \vee L)$	
2	$\neg L$	(Want: J)
3	$J \vee L$	:1, 2 $\to$E
4	J	:2, 3 DS

It is easy to see that $J \vee L$ follows by $\to$E from the two premises, but it is difficult to see how the proof will go next were we constrained to the basic rules. However, given DS, it is plain to see that J follows immediately from $J \vee L$ and $\neg L$. So the proof is easy.

4.4.4 Hypothetical Syllogism

We also add *hypothetical syllogism* (HS) as a derived rule:

m	$\varphi \to \psi$	
n	$\psi \to \chi$	
	$\varphi \to \chi$	:m, n HS

Note that HS does not cite any subproofs, and so makes for elegant proofs that are easy to read. The same cannot be said for the derivation schema for HS:

1	$\varphi \to \psi$	
2	$\psi \to \chi$	
3	$\quad \varphi$	:AS
4	$\quad \psi$	:1, 3 $\to$E
5	$\quad \chi$	:2, 4 $\to$E
6	$\varphi \to \chi$	:3–5 $\to$I

4.4.5 Contraposition

Next we may add *contraposition* (CP) as a derived rule:

m	$\varphi \to \psi$	
	$\neg\varphi \to \neg\psi$	:m CP

Not only is this inference natural, it is extremely useful. We have had various occasions to use CP in the informal proofs given above. The derivation in PL goes as follows:

1	$\varphi \to \psi$	
2	$\quad \neg\psi$	:AS
3	$\quad\quad \varphi$	:AS
4	$\quad\quad \psi$	:1, 3 $\to$E
5	$\quad\quad \neg\psi$	:2 R
6	$\quad \neg\varphi$	:3–5 $\neg$I
7	$\neg\psi \to \neg\varphi$	:2–6 $\to$I

Whereas the proof above involves two subproofs, one embedded in the other, applications of CP directly cite live lines of a proof, greatly simplifying the resulting argument.

4.4.6 Negative Biconditionals

Biconditional elimination only works when we have a biconditional together with one of the arguments of the biconditional on live lines. However, it in cases where we have the negation of one of the arguments of a biconditional, it is convenient to make use of the following derived rule for *negative biconditionals* (NB):

m	$\varphi \leftrightarrow \psi$	
n	$\neg\varphi$	
	$\neg\psi$	:m, n NB

m	$\varphi \leftrightarrow \psi$	
n	$\neg\psi$	
	$\neg\varphi$	:m, n NB

The derivations for NB go as follows:

1	$\varphi \leftrightarrow \psi$	
2	$\neg\varphi$	
3	$\quad \psi$	:AS
4	$\quad \varphi$	:1, 3 $\leftrightarrow$E
5	$\quad \neg\varphi$	:2 R
6	$\neg\psi$	:3–5 $\neg$I

1	$\varphi \leftrightarrow \psi$	
2	$\neg\psi$	
3	$\quad \varphi$	:AS
4	$\quad \psi$	:1, 3 $\leftrightarrow$E
5	$\quad \neg\psi$	:2 R
6	$\neg\varphi$	:3–5 $\neg$I

4.4.7 Double Negation

Whereas we have included two similar rules for negation introduction and elimination, some texts only include negation introduction together with the following rule for *double negation elimination* (DN):

m	$\neg\neg\varphi$	
	φ	$:m$ DN

Although some philosophers of logic contest DN, arguing instead for *intuitionistic logics* in which DN is neither basic nor derivable, most take DN to be a useful and extremely natural inference to draw. After all, what is meant by saying that it is not the case that the ball is not round, and yet it fails to be the case that the ball is round? Or to take the converse, what is meant by saying that the ball is round, but it fails to be the case that the ball is not not round. The classical logician may claim that there is no difference at all here by accepting DN. Although, DN is a derived rule in PL rather than basic, this much is only a difference in convention. Here is the derivation of DN in the present system PL:

1	$\neg\neg\varphi$	
3	$\quad\neg\varphi$	:AS for $\neg$E
4	$\quad\neg\neg\varphi$	:1 R
5	φ	:3–4 $\neg$E

As in the other case, our derived rule DN allows us to draw natural inferences with minimal complexity, avoiding the need to open any subproofs.

4.4.8 Ex Falso Quodlibet

From a falsehood anything follows, or in Latin, *ex falso quodlibet.* For instance, if A is false, then $\neg A$ is true, and so if we were to take A to also be true, then together we may derive B from this contradiction. More generally, we have the following rule (EFQ):

m	φ	
n	$\neg\varphi$	
	ψ	$:m$, n EFQ

This inference is occasionally convenient since, given φ and $\neg\varphi$ on live lines we may draw any conclusion that we might happen to want on the next line. Here is the derivation of EFQ.

1	φ	
2	$\neg\varphi$	
3	$\quad \neg\psi$	:AS for $\neg$E
4	$\quad \varphi$	:1 R
5	$\quad \neg\varphi$	:2 R
6	ψ	:3–5 $\neg$E

This puts a syntactic spin on a semantic idea that we considered before: just as every wfs of $\mathcal{L}^{\text{PL}}$ is a logical consequence of an unsatisfiable sets of wfss of $\mathcal{L}^{\text{PL}}$, every wfs of $\mathcal{L}^{\text{PL}}$ can be derived from any wfss φ and $\neg\varphi$ of $\mathcal{L}^{\text{PL}}$, and indeed from any set Γ containing φ and $\neg\varphi$.

The explosion of wfss of $\mathcal{L}^{\text{PL}}$ that can be derived from a set containing φ and $\neg\varphi$ helps to shed light on why a set Γ of wfss of $\mathcal{L}^{\text{PL}}$ was said to be inconsistent in PL just in case $\Gamma \vdash \bot$. Since $\bot \coloneqq A \wedge \neg A$, both $\bot \vdash A$ and $\bot \vdash \neg A$ by $\wedge$E, and so by EFQ, any ψ can be derived from $\bot$. Thus if $\Gamma \vdash \bot$, it follows that $\Gamma \vdash \psi$ for any wfs ψ of $\mathcal{L}^{\text{PL}}$ whatsoever.

4.4.9 Law of Excluded Middle

Recall from Chapter 2 that the $\mathcal{L}^{\text{PL}}$ interpretations assign every sentence letter of $\mathcal{L}^{\text{PL}}$ to exactly one of just two truth-values 1 and 0. It follows that every wfs φ of $\mathcal{L}^{\text{PL}}$ is assigned to either 1 or 0 and not both, i.e., $\mathcal{V}_{\mathcal{I}}(\varphi) \in \{1, 0\}$. Thus $\mathcal{V}_{\mathcal{I}}(\varphi \vee \neg\varphi) = 1$ for any wfs φ and interpretation $\mathcal{I}$ of $\mathcal{L}^{\text{PL}}$, and so every instance of $\varphi \vee \neg\varphi$ is a tautology. The syntactic analogue of this semantic claim asserts that every instance of $\varphi \vee \neg\varphi$ is a theorem of PL which, given its central place within classical logic, is referred to as *the law of excluded middle*:

$\varphi \vee \neg\varphi$	:LEM

1	$\quad \neg(\varphi \vee \neg\varphi)$	:AS for $\neg$E
2	$\quad\quad \varphi$	:AS for $\neg$I
3	$\quad\quad \varphi \vee \neg\varphi$	:2 $\vee$I
4	$\quad\quad \neg(\varphi \vee \neg\varphi)$	:1 R
5	$\quad \neg\varphi$	:2–4 $\neg$I
6	$\quad \varphi \vee \neg\varphi$	:5 $\vee$I
7	$\varphi \vee \neg\varphi$	:1–6 $\neg$E

By contrast with the basic and derived rules given above, theorems do not cite previous lines of the proofs in which they occur, though they are justified all the same. This is because applications of LEM abbreviate proofs of the form given above on the right.

4.4.10 Law of Non-Contradiction

Given that $\mathcal{V}_{\mathcal{I}}(\varphi) \in \{1, 0\}$ for any wfs φ of $\mathcal{L}^{\text{PL}}$, it also follows by the semantics for negation and conjunction that $\mathcal{V}_{\mathcal{I}}(\varphi \wedge \neg\varphi) = 0$ for any wfs φ and interpretation $\mathcal{I}$ of $\mathcal{L}^{\text{PL}}$, and so every instance of $\varphi \wedge \neg\varphi$ is a contradiction. Equivalently, all instances of $\neg(\varphi \wedge \neg\varphi)$ are tautologies, and so we may expect $\neg(\varphi \wedge \neg\varphi)$ to be a theorem for any wfs φ of $\mathcal{L}^{\text{PL}}$. In order to cover all instances, we may provide the following derivation:

$| \; \neg(\varphi \wedge \neg\varphi)$:LNC

1	$\quad\; \varphi \wedge \neg\varphi$	:AS for $\neg$I
2	$\quad\; \varphi$	:1 $\wedge$E
3	$\quad\; \neg\varphi$	:1 $\wedge$E
4	$\neg(\varphi \wedge \neg\varphi)$	:1–3 $\neg$I

Having observed that every instance of $\varphi \vee \neg\varphi$ and $\neg(\varphi \wedge \neg\varphi)$ are tautologies, one might reasonably expect these to be derivable in PL though nothing so far allows us to jump to this conclusion. Rather, the derivations above do important work, indicating that PL is doing what it should do by allowing us to reason our way to the logical consequences of any set of premises where the logical consequences of the empty set are a special case. Nevertheless, we should like to know if there is anything missing. That is, we may ask whether there are logical consequences of $\mathcal{L}^{\text{PL}}$ which PL is unable to derive. It turns out that this is not the case: every logical consequences of $\mathcal{L}^{\text{PL}}$ whatsoever is derivable in PL. In a word, PL is *complete*.

We will turn to prove PL Completeness in the following chapter. For the time being, there is an important consequence of PL Soundness that we are now in a position to draw.

4.5 Consistency

In the previous section we set about deriving a host of rules and theorems. You might begin to wonder just how many derived rules and theorems there are where it might be natural to think that the more the better. Another way to put this point is in terms of the strength of PL as a proof system where this refers to how much we can derive with the basic rules that PL provides. Accordingly, one might be tempted to think that stronger logics are better. After all, what could be bad about being able to derive more rather than less?

Tempting as it may be to think that strength is only a good thing, we have already seen some cases where being able to derive too much is not a good thing. In particular, we saw that everything can be derived from a set containing a wfs of $\mathcal{L}^{\text{PL}}$ and its negation. More generally, all wfss of $\mathcal{L}^{\text{PL}}$ are derivable from an inconsistent set of wfss of $\mathcal{L}^{\text{PL}}$. We may then prove:

Corollary 4.1 If Γ is inconsistent, then Γ is unsatisfiable.

Having established PL Soundness and **Lemma 2.2**, the proof of **Corollary 4.1** follows easily and so has been left as an exercise for the reader. By contrast with lemmas which are used to establish important results, corollaries are the consequences of important results. What the corollary above shows is that any set of wfss of $\mathcal{L}^{\text{PL}}$ that is strong enough to be able to derive all wfss of $\mathcal{L}^{\text{PL}}$ is also unsatisfiable. Thus we also have the immediate consequence:

Corollary 4.2 If Γ is satisfiable, then Γ is consistent.

Proof: Follows immediately from **Corollary 4.1** by contraposition. □

Whereas inconsistency has witnesses, consistency does not. That is, although you might show how to derive $\bot$ from some set Γ of wfss of $\mathcal{L}^{\text{PL}}$ by providing a particular derivation in PL, to claim that Γ is consistent is to say that there is no way to derive $\bot$ from Γ in PL. This might seem like a hard thing to show since how should we expect to survey the entire space of possible derivations in order to claim that there are none in which $\bot$ is the conclusion and Γ is the set of premises? Were one to proceed by *reductio*, it is not clear how to derive a contradiction from the assumption that there is a derivation of $\bot$ from Γ in PL.

Given **Corollary 4.2**, there is a much easier way to show that a set Γ of wfss of $\mathcal{L}^{\text{PL}}$ is consistent: simply show that it is satisfiable. Whereas consistency does not have witnesses on account of asserting something general, satisfiability does have witnesses. That is, given any satisfiable set Γ, there is a least one particular interpretation $\mathcal{I}$ of $\mathcal{L}^{\text{PL}}$ where $\mathcal{V}_{\mathcal{I}}(\gamma) = 1$ for all $\gamma \in \Gamma$. Assuming that we can identify an interpretation $\mathcal{I}$ which witnesses the satisfiability of Γ, we may draw on **Corollary 4.2** in order to conclude that Γ is consistent.

There is a particularly important application of this general procedure. That is, something we should like to know is whether the theorems of PL are consistent, since if the theorems of PL turned out to be inconsistent, then PL would be so strong as to be able to derive anything from nothing. But that is not what we want. Rather, our hope in setting up PL was to describe what follows from what in virtue of logical form where we had previously characterized this by defining logical consequence. If it turns out that everything is derivable from nothing, then all our hard work will have been for nothing since PL will have been shown to massively overshoot its intended target: formal reasoning.

Given our present strategy, all that remains is to find an interpretation of $\mathcal{L}^{\text{PL}}$ that satisfies all of the theorems of PL. But how shall we choose? The answer is that we don't need to: any interpretation at all will do. Since we know by PL Soundness that every theorem of PL is a $\mathcal{L}^{\text{PL}}$ tautology, $\mathcal{V}_{\mathcal{I}}(\varphi) = 1$ for any theorem φ of PL and interpretation $\mathcal{I}$ of $\mathcal{L}^{\text{PL}}$ whatsoever. As our witness, suppose we choose the $\mathcal{L}^{\text{PL}}$ interpretation $\mathcal{I}^+$ where $\mathcal{V}_{\mathcal{I}^+}(\psi) = 1$ for every sentence letter ψ of $\mathcal{L}^{\text{PL}}$. Since, like any $\mathcal{L}^{\text{PL}}$ interpretation, $\mathcal{V}_{\mathcal{I}^+}(\varphi) = 1$ for every theorem φ of PL, it follows that the theorems of PL are indeed satisfiable, and so consistent by **Corollary 4.2** above. Thus we may conclude that despite all of the rules and theorems we derived, PL is not so strong as to be able to derive everything from nothing.

Chapter 5

The Completeness of PL

PL COMPLETENESS is not mandatory in the same way as PL SOUNDNESS. Continuing the analogy above, we may observe that no calculator is complete for the simple reason that every calculator has a finite amount of memory which is exhausted by arithmetical operations with sufficiently large operands. Even so, this does not stop calculators from being of considerable utility. These considerations might lead one to give up any hope of finding a complete logic for $\mathcal{L}^{\text{PL}}$, settling for a logic that is at least sound and so consistent. However, logical systems are not finite mechanisms made up of material elements such as the bits inside a calculator. For this reason, logics don't face the same constraints that a calculator does, and so the analogy breaks down. In this chapter, we will prove PL COMPLETENESS thereby establishing that PL does not leave any room for logical consequences that cannot be derived within PL. This is a beautiful result and a great achievement of twentieth century logic.

5.1 Introduction

Completeness asserts that φ is derivable from Γ whenever φ is a logical consequence of Γ, or more compactly: if $\Gamma \vDash \varphi$, then $\Gamma \vdash \varphi$. Instead of beginning with $\Gamma \vDash \varphi$ as an assumption and arguing to the conclusion $\Gamma \vdash \varphi$, we will focus on establishing a closely related result:

> **Theorem 5.1** If Γ is consistent, Γ is satisfiable.

You might recognize this as the converse of **Corollary 4.2** from Chapter 4. Indeed, **Theorem 5.1** will bear a close connection to PL COMPLETENESS in a way that is related to the connection between **Corollary 4.2** and PL SOUNDNESS. However, instead of proving **Theorem 5.1** from PL COMPLETENESS, the proof will work in the reverse direction so that officially PL COMPLETENESS will be a corollary of **Theorem 5.1**. Before diving into the proof in earnest, this section will present an overview of the proof so that you can find your bearings. If you get lost, think of this section as a map to which you can return.

Recall that a set of $\mathcal{L}^{\text{PL}}$ wfss Γ is inconsistent if $\bot$ is derivable from Γ, and consistent otherwise. If $\Gamma \vDash \varphi$, we know that $\Gamma \cup \{\neg\varphi\}$ is unsatisfiable by **Lemma 2.3**, and so $\Gamma \cup \{\neg\varphi\}$ is inconsistent by **Theorem 5.1**. We will then show that it follows that $\Gamma \vdash \neg\neg\varphi$ by appealing to **Lemma 5.3** below. Given the derived rule DN for double negation elimination, we may conclude that $\Gamma \vdash \varphi$ from the assumption that $\Gamma \vDash \varphi$. This provides a sketch of the completeness proof for PL given the theorem above (see **Corollary 5.1** for the full proof).

It remains to establish **Theorem 5.1**. We begin by assuming that Γ is a consistent set of wfss of $\mathcal{L}^{\text{PL}}$. We will then extend Γ so that it includes every wfs or its negation but not both, calling this *maximal* set Δ_Γ, or often just Δ for short. We will show in §5.2 that Δ is consistent, where it follows that Δ is *deductively closed* insofar as it contains every wfs that is derivable from Δ. Deductive closure is a very important and convenient property which will play a critical role in the later stages of the proof.

Having extended Γ to a much bigger set of wfss Δ that is consistent, maximal, and deductively closed, we will proceed to use this set of wfss to construct an interpretation $\mathcal{I}_\Delta$ of $\mathcal{L}^{\text{PL}}$ that satisfies Δ, and so satisfies Γ as a result. This may sound strange since we are starting with a set Γ of wfss of $\mathcal{L}^{\text{PL}}$ to ultimately interpret the wfss in Γ itself. Strange as this may seem, there is no circularity here: to interpret our language $\mathcal{L}^{\text{PL}}$, we need a systematic way to assign each sentence letter to exactly one truth-value. So long as we achieve this without assuming that such an assignment has already been given, no questions will have been begged.

Given that $\mathcal{I}_\Delta$ satisfies Γ, we may conclude that Γ is satisfiable. Following tradition, we will refer to this cleverly constructed interpretation $\mathcal{I}_\Delta$ as a Henkin interpretation after Leon Henkin who first presented this proof strategy in 1949. In Chapter 12, we will extend this same method to show that the first-order logic FOL that we will provide in Chapter 10 is complete with respect to the theory of logical consequence given in Chapter 8. Not only will we repeat the same methodology that we have followed so far in developing a semantics, logic, and metalogic for $\mathcal{L}^{\text{PL}}$ for our first-order language $\mathcal{L}^{\text{FOL}}$, many of the same results will carry over. For the time being, we will continue to restrict attention to $\mathcal{L}^{\text{PL}}$, it's semantics, and the proof system PL with which we are presently concerned.

This provides a rough overview of the proof strategy that will be deployed below. If you get lost along the way, continuing to slog on in the dark is not advisable. Rather, it is better to keep zooming out so that you can keep track of where you are and where you are headed to next. In addition to returning to this section, you may need to scan back through the proof multiple times, unlocking each piece and slowly watching them come together.

Without further ado, we may make the first assumption indicated above that Γ is an arbitrary set of wfss of $\mathcal{L}^{\text{PL}}$. Although it may help to hold onto Γ throughout the course of the following sections, many of the lemmas that we establish will be completely general in form, holding for any arbitrary set of wfss of $\mathcal{L}^{\text{PL}}$. We will then put these general results together to apply to Γ in order to show that Γ is indeed satisfiable.

5.2 Maximal Consistency

A set of wfss Δ is MAXIMAL in $\mathcal{L}^{\text{PL}}$ just in case as either $\psi \in \Delta$ or $\neg\psi \in \Delta$ for every wfs ψ in $\mathcal{L}^{\text{PL}}$. Having assumed that Γ is consistent in $\mathcal{L}^{\text{PL}}$, we may maximize Γ by adding every wfs of $\mathcal{L}^{\text{PL}}$ that we can without inconsistency. To do so, we will begin enumerating all wfss $\psi_0, \psi_1, \psi_2, \ldots$ in $\mathcal{L}^{\text{PL}}$ whatsoever in order to present the following recursive construction:

$$\begin{aligned}
\Delta_0 &= \Gamma \\
\Delta_{n+1} &= \begin{cases} \Delta_n \cup \{\psi_n\} & \text{if } \Delta_n \cup \{\psi_n\} \text{ is consistent} \\ \Delta_n \cup \{\neg\psi_n\} & \text{otherwise.} \end{cases} \\
\Delta_\Gamma &= \bigcup_{i \in \mathbb{N}} \Delta_n.
\end{aligned}$$

If Γ is consistent, we may show that Δ_Γ is both consistent and maximal. Given these two properties, we may then show that Δ_Γ is deductively closed. Moreover, we may show $\Gamma \subseteq \Delta_\Gamma$. These properties will form the basis upon which we will construct the Henkin interpretation in §5.3. In order to establish these results, we will begin by proving some supporting lemmas. Whereas some of these lemmas are substantial, others are simply convenient, allowing us to streamline the presentation of the proofs of the results to come.

Lemma 5.1 If $\Lambda \cup \{\varphi\}$ is inconsistent, then $\Lambda \vdash \neg\varphi$.

Proof: Assume $\Lambda \cup \{\varphi\}$ is inconsistent. Thus $\Lambda \cup \{\varphi\} \vdash \bot$, and so there is a derivation X of $A \wedge \neg A$ from $\Lambda \cup \{\varphi\}$ given the definition of $\bot$. Let X' be the result of replacing the premise φ with φ as an assumption and adding lines for A and $\neg A$ by $\wedge$E. We may then discharge the assumption of φ by $\neg$I in order to derive $\neg\varphi$ from Λ. Thus we may conclude that $\Lambda \vdash \neg\varphi$. $\square$

Lemma 5.2 If $\Lambda \vdash \varphi$ and $\Lambda \vdash \neg\varphi$, then Λ is inconsistent.

Proof: Assume $\Lambda \vdash \varphi$ and $\Lambda \vdash \neg\varphi$. Thus there is a PL derivation X of φ from Λ as well as a PL derivation Y of $\neg\varphi$ from Λ. Letting Z be the result of concatenating X and Y and renumbering lines. We may then extend the derivation by using EFQ from §4.4.8 to derive $A \wedge \neg A$, observing that Z is a derivation of $A \wedge \neg A$ from Λ. By definition, Λ is inconsistent. $\square$

Lemma 5.3 If $\Lambda \cup \{\varphi\}$ and $\Lambda \cup \{\neg\varphi\}$ are both inconsistent, then Λ is inconsistent.

Proof: Assuming that $\Lambda \cup \{\varphi\}$ and $\Lambda \cup \{\neg\varphi\}$ are both inconsistent, it follows that $\Lambda \vdash \neg\varphi$ and $\Lambda \vdash \neg\neg\varphi$ by **Lemma 5.1**. Given **Lemma 5.2** above, it follows immediately that Λ is inconsistent. $\square$

Lemma 5.4 If Γ is consistent in $\mathcal{L}^{\text{PL}}$, then Δ_Γ is maximal consistent.

Proof: Assume Γ is consistent and let φ be any wfs of $\mathcal{L}^{\text{PL}}$. Thus $\varphi = \psi_i$ for some $i \in \mathbb{N}$ given the enumeration above where either $\psi_i \in \Delta_{i+1}$ or $\neg\psi_i \in \Delta_{i+1}$. Since $\Delta_{i+1} \subseteq \Delta_\Gamma$, either $\varphi \in \Delta_\Gamma$ or $\neg\varphi \in \Delta_\Gamma$, and so Δ_Γ is maximal.

The proof that Δ_Γ is consistent goes by induction on the construction of Δ_Γ, where we know by assumption that $\Gamma = \Delta_0$ is consistent. Assume for induction that Δ_n is consistent. There are two cases to consider.

Case 1: $\Delta_n \cup \{\psi_n\}$ is consistent, and so $\Delta_{n+1} = \Delta_n \cup \{\psi_n\}$ is consistent.

Case 2: $\Delta_n \cup \{\psi_n\}$ is not consistent, and so $\Delta_{n+1} = \Delta_n \cup \{\neg\psi_n\}$. Assume for contradiction that $\Delta_n \cup \{\neg\psi_n\}$ is not consistent. By **Lemma 5.3**, Δ_n is inconsistent, contradicting the hypothesis. Thus Δ_{n+1} is consistent.

Since Δ_{n+1} is consistent, it follows by induction that Δ_k is consistent for all $k \in \mathbb{N}$. Assume for contradiction that Δ_Γ is inconsistent. Thus $\Delta_\Gamma \vdash \bot$, and so there is a proof Y of $A \wedge \neg A$ from Δ_Γ given the definition of $\bot$. Since Y is finite, there is a finite number of premises cited in Y, and so there is some $k \in \mathbb{N}$ where every premise cited in Y belongs to Δ_k. As a result, Y is also a proof of $A \wedge \neg A$ from Δ_k, and so Δ_k is inconsistent, contradicting the above. Thus Δ_Γ is consistent. $\square$

Whereas maximality is relatively cheap— *viz.* the set of all wfss of $\mathcal{L}^{\text{PL}}$ is maximal— showing that a set of wfss is both maximal and consistent is much more difficult and, as we will soon see, much more significant. Intuitively, the Henkin construction given above specifies a way to increase the strength (i.e., deductive power) of a consistent set as much as possible without crossing over into inconsistency. Maximal consistent sets of this kind may be shown to have a very important property which we may now turn to define.

A set Δ of wfss of $\mathcal{L}^{\text{PL}}$ is DEDUCTIVELY CLOSED in PL just in case for any wfs φ of $\mathcal{L}^{\text{PL}}$, if $\Delta \vdash \varphi$, then $\varphi \in \Delta$. Accordingly, deductively closed sets of wfss are identical to the set of wfss which they derive. In order to show that every maximal consistent set of wfss of $\mathcal{L}^{\text{PL}}$ is deductively closed, we begin by establishing the following lemma.

Lemma 5.5 If Δ is maximal consistent, then Δ is deductively closed.

Proof: Assume Δ is maximal consistent. Let φ be a wfs of $\mathcal{L}^{\text{PL}}$ where $\Delta \vdash \varphi$. If $\Delta \vdash \neg\varphi$, then Δ is inconsistent by **Lemma 5.2**, contradicting the assumption. Thus $\Delta \nvdash \neg\varphi$, and so it follows that $\neg\varphi \notin \Delta$ since otherwise $\Delta \vdash \neg\varphi$ follows by the reiteration rule R. Since Δ is maximal, we may conclude that $\varphi \in \Delta$. $\square$

Given these results, we are now ready to draw on Δ_Γ in order define the Henkin interpretation of $\mathcal{L}^{\text{PL}}$. We will then direct our efforts towards showing that this interpretation witnesses the satisfiability of Δ_Γ, and so so of Γ in particular.

5.3 Henkin Interpretation

Having extended the consistent set of wfss Γ in $\mathcal{L}^{\text{PL}}$ to a maximal consistent set of wfss Δ_Γ in $\mathcal{L}^{\text{PL}}$ and showing that Δ_Γ is deductively closed, we may use Δ_Γ to construct a Henkin interpretation that satisfies Δ_Γ, and so also satisfies Γ. For ease of exposition, we will drop the subscripts, assuming $\Delta = \Delta_\Gamma$ throughout what follows.

We may now proceed to draw on the definition of Δ in order to specify an especially natural interpretation of $\mathcal{L}^{\text{PL}}$ which will guarantee that the resulting interpretation satisfies all of the wfss in Δ. In particular, consider the following definitions:

$$\text{For all sentence letters } \varphi \text{ of } \mathcal{L}^{\text{PL}}, \text{ let } \mathcal{I}_\Delta(\varphi) = \begin{cases} 1 & \text{if } \varphi \in \Delta \\ 0 & \text{otherwise.} \end{cases}$$

Since $\mathcal{I}_\Delta$ assigns every sentence letter of $\mathcal{L}^{\text{PL}}$ to a truth-value, $\mathcal{I}_\Delta$ satisfies the definition of a $\mathcal{L}^{\text{PL}}$ interpretation. Since this construction was introduced by Leon Henkin (1949), we will refer to $\mathcal{I}_\Delta$ as the HENKIN INTERPRETATION of $\mathcal{L}^{\text{PL}}$ for Γ (recall that $\Delta = \Delta_\Gamma$).

It remains to show that $\mathcal{I}_\Delta$ satisfies Δ, and so satisfies Γ as a result. To do so, we will begin by proving the following lemmas where the first is a proof theoretic analogue of **Lemma 2.1**.

Lemma 5.6 If $\Lambda \vdash \varphi$, then $\Lambda \cup \Pi \vdash \varphi$.

> *Proof:* Assuming that $\Lambda \vdash \varphi$, there is a derivation X of φ from Λ in PL. Since $\Lambda \subseteq \Lambda \cup \Pi$, it follows that X is also a derivation of φ from $\Lambda \cup \Pi$ in PL. Thus we may conclude that $\Lambda \cup \Pi \vdash \varphi$. □

Lemma 5.7 If Δ is a maximal consistent set of wfss of $\mathcal{L}^{\text{PL}}$, then for every wfs φ of $\mathcal{L}^{\text{PL}}$: $\mathcal{V}_{\mathcal{I}_\Delta}(\varphi) = 1$ just in case $\varphi \in \Delta$.

> *Proof:* Assume Δ is a maximal consistent set of $\mathcal{L}^{\text{PL}}$ wfss. We will show by induction on complexity that for any wfs φ of $\mathcal{L}^{\text{PL}}$, $\mathcal{V}_{\mathcal{I}_\Delta}(\varphi) = 1$ just in case $\varphi \in \Delta$. After the base case, there are five cases in the induction step.
>
> *Base Case:* Let φ be an arbitrary wfs of $\mathcal{L}^{\text{PL}}$ where $\texttt{Comp}(\varphi) = 0$. It follows that φ is a sentence letter of $\mathcal{L}^{\text{PL}}$. We may then consider the following biconditionals:
>
> $$\begin{aligned} \mathcal{V}_{\mathcal{I}_\Delta}(\varphi) = 1 \quad & \textit{iff} \quad \mathcal{I}_\Delta(\varphi) = 1 \\ & \textit{iff} \quad \varphi \in \Delta. \end{aligned}$$
>
> Since φ is a sentence letter, the first biconditional follows by the semantics for $\mathcal{L}^{\text{PL}}$, and the second biconditional follows from the definition of $\mathcal{I}_\Delta$. It follows that for any wfs φ of $\mathcal{L}^{\text{PL}}$ where $\texttt{Comp}(\varphi) = 0$, $\mathcal{V}_{\mathcal{I}_\Delta}(\varphi) = 1$ just in case $\varphi \in \Delta$.

Induction: Assume for induction that for every wfs φ of $\mathcal{L}^{\text{PL}}$, if $\texttt{Comp}(\varphi) \leqslant n$, then $\mathcal{V}_{\mathcal{I}_\Delta}(\varphi) = 1$ just in case $\varphi \in \Delta$. Let φ be a wfs of $\mathcal{L}^{\text{PL}}$ where $\texttt{Comp}(\varphi) = n + 1$.

Case 1: Assume $\varphi = \neg\psi$. Since $\texttt{Comp}(\neg\psi) = \texttt{Comp}(\psi) + 1$ and $\texttt{Comp}(\varphi) = n + 1$, it follows that $\texttt{Comp}(\psi) = n$. We may then reason as follows:

$$\begin{aligned}
\mathcal{V}_{\mathcal{I}_\Delta}(\varphi) = 1 \quad &\textit{iff} \quad \mathcal{V}_{\mathcal{I}_\Delta}(\neg\psi) = 1 \\
&\textit{iff} \quad \mathcal{V}_{\mathcal{I}_\Delta}(\psi) = 0 && (1)\\
&\textit{iff} \quad \psi \notin \Delta && (2)\\
&\textit{iff} \quad \neg\psi \in \Delta && (3)\\
&\textit{iff} \quad \varphi \in \Delta.
\end{aligned}$$

Whereas (1) follows from the semantics for negation and (2) holds by hypothesis, (3) follows from the maximality of Δ. The other biconditionals follow from the case assumption. Thus $\mathcal{V}_{\mathcal{I}_\Delta}(\varphi) = 1$ just in case $\varphi \in \Delta$, completing the case.

Case 2: Assume $\varphi = \psi \wedge \chi$. Since $\texttt{Comp}(\psi \wedge \chi) = \texttt{Comp}(\psi) + \texttt{Comp}(\chi) + 1$ and $\texttt{Comp}(\varphi) = n + 1$, it follows that $\texttt{Comp}(\psi), \texttt{Comp}(\chi) \leqslant n$. Thus we have:

$$\begin{aligned}
\mathcal{V}_{\mathcal{I}_\Delta}(\varphi) = 1 \quad &\textit{iff} \quad \mathcal{V}_{\mathcal{I}_\Delta}(\psi \wedge \chi) = 1 \\
&\textit{iff} \quad \mathcal{V}_{\mathcal{I}_\Delta}(\psi) = \mathcal{V}_{\mathcal{I}_\Delta}(\chi) = 1 && (1)\\
&\textit{iff} \quad \psi, \chi \in \Delta && (2)\\
&\textit{iff} \quad \psi \wedge \chi \in \Delta && (3)\\
&\textit{iff} \quad \varphi \in \Delta.
\end{aligned}$$

Whereas (1) follows from the semantics for conjunction, (2) holds by hypothesis. In order to establish (3), assume that $\psi \wedge \chi \in \Delta$, it follows that $\Delta \vdash \psi$ and $\Delta \vdash \chi$ by $\wedge$E, and so $\psi, \chi \in \Delta$ by **Lemma 5.5**. Assuming instead that $\psi, \chi \in \Delta$, we know that $\Delta \vdash \psi \wedge \chi$ by $\wedge$I, and so $\psi \wedge \chi \in \Delta$ by **Lemma 5.5**. This proves (3) where the other biconditionals follow from the case assumption.

Case 3: Assume $\varphi = \psi \vee \chi$. (Exercise for the reader.)

Case 4: Assume $\varphi = \psi \to \chi$. Since $\texttt{Comp}(\psi \to \chi) = \texttt{Comp}(\psi) + \texttt{Comp}(\chi) + 1$ and $\texttt{Comp}(\varphi) = n + 1$, it follows that $\texttt{Comp}(\psi), \texttt{Comp}(\chi) \leqslant n$. Thus we have:

$$\begin{aligned}
\mathcal{V}_{\mathcal{I}_\Delta}(\varphi) = 1 \quad &\textit{iff} \quad \mathcal{V}_{\mathcal{I}_\Delta}(\psi \to \chi) = 1 \\
&\textit{iff} \quad \mathcal{V}_{\mathcal{I}_\Delta}(\psi) = 0 \text{ or } \mathcal{V}_{\mathcal{I}_\Delta}(\chi) = 1 && (1)\\
&\textit{iff} \quad \psi \notin \Delta \text{ or } \chi \in \Delta && (2)\\
&\textit{iff} \quad \psi \to \chi \in \Delta && (3)\\
&\textit{iff} \quad \varphi \in \Delta.
\end{aligned}$$

Whereas (1) follows from the semantics for conjunction, (2) holds by hypothesis.

In order to establish (3), assume that $\psi \notin \Delta$. Since Δ is maximal, it follows that $\neg\psi \in \Delta$. We may then derive $\neg\psi \vdash \psi \to \chi$ since given $\neg\psi$ as a premise, we may use the rule AS to write ψ on a second line, deriving χ by EFQ from §4.4.8

and using $\to$I to discharge the assumption. Thus $\Delta \vdash \psi \to \chi$ by **Lemma 5.6**, and so $\psi \to \chi \in \Delta$ by **Lemma 5.5**. Next we may assume that $\chi \in \Delta$, we may derive $\chi \vdash \psi \to \chi$ since given χ as a premise, we may use the rule AS to write ψ on a second line. By then using the rule R, we may rewrite the premise χ, discharging our assumption with the rule $\to$I in order to derive $\psi \to \chi$ from χ. Thus $\Delta \vdash \psi \to \chi$ by **Lemma 5.6**, and so $\psi \to \chi \in \Delta$ by **Lemma 5.5**. We may then conclude that $\psi \to \chi \in \Delta$ if either $\psi \notin \Delta$ or $\chi \in \Delta$.

Assume instead that $\psi \to \chi \in \Delta$. If $\psi \notin \Delta$, then $\psi \notin \Delta$ or $\chi \in \Delta$. If $\psi \in \Delta$, then $\Delta \vdash \chi$ by the rule $\to$E, and so $\chi \in \Delta$ by **Lemma 5.5**. Thus $\psi \notin \Delta$ or $\chi \in \Delta$ if $\psi \to \chi \in \Delta$ which, given the above, establishes (3).

The other biconditionals follow from the case assumption.

Case 5: Assume $\varphi = \psi \leftrightarrow \chi$. (Exercise for the reader.)

Conclusion: It follows by induction that for every wfs φ of $\mathcal{L}^{\text{PL}}$ of any complexity, $\mathcal{V}_{\mathcal{I}_\Delta}(\varphi) = 1$ just in case $\varphi \in \Delta$. This completes the proof. □

5.4 Completeness and Compactness

Having constructed a maximal consistent set Δ of wfss of $\mathcal{L}^{\text{PL}}$ from the consistent set Γ of wfs of $\mathcal{L}^{\text{PL}}$ and defined the Henkin interpretation $\mathcal{I}_\Delta$ as above, we are now ready to draw on the lemmas above in order to show that Δ is satisfiable. In order to extend this result to Γ, we may begin with the trivial lemma given below.

Lemma 5.8 $\Gamma \subseteq \Delta_\Gamma$.

Proof: By definition, $\Gamma = \Delta_0$ where $\Delta_0 \subseteq \Delta_\Gamma$. □

This lemma amounts to little more than an observation, but will be convenient to reference below. We may now move to draw the following conclusion:

Theorem 5.1 If Γ is consistent, then Γ is satisfiable.

Proof: Let Γ be a consistent set of wfss of $\mathcal{L}^{\text{PL}}$. By **Lemma 5.4**, Δ_Γ is a maximal consistent set of wfss in $\mathcal{L}^{\text{PL}}$. Letting $\Delta = \Delta_\Gamma$ and $\mathcal{I}_\Delta$ be the Henkin interpretation of $\mathcal{L}^{\text{PL}}$ defined above, **Lemma 5.7** shows that for every wfs φ of $\mathcal{L}^{\text{PL}}$, $\mathcal{V}_{\mathcal{I}}(\varphi) = 1$ just in case $\varphi \in \Delta$. Thus $\mathcal{V}_{\mathcal{I}}(\varphi) = 1$ for all $\varphi \in \Delta$. Since $\Gamma \subseteq \Delta$ by **Lemma 5.8**, $\mathcal{V}_{\mathcal{I}}(\varphi) = 1$ for all $\varphi \in \Gamma$. By definition, Γ is satisfiable. □

Given this result, the completeness of PL over the semantics for $\mathcal{L}^{\text{PL}}$ follows as a corollary.

Corollary 5.1 (Completeness) If $\Gamma \vDash \varphi$, then $\Gamma \vdash \varphi$.

> *Proof:* Assume $\Gamma \vDash \varphi$. By **Lemma 2.3**, $\Gamma \cup \{\neg\varphi\}$ is unsatisfiable, and so $\Gamma \cup \{\neg\varphi\}$ is inconsistent by **Theorem 5.1**. Thus $\Gamma \vdash \neg\neg\varphi$ by **Lemma 5.1**. Thus there is some PL derivation X of $\neg\neg\varphi$ from Γ. Given the rule DN derived in §4.4.7, we may extend X to derive φ from Γ, and so conclude that $\Gamma \vdash \varphi$. □

Completeness may seem like a good property for any proof system to have. In particular, the completeness of PL shows that there is no (extensionally) better proof system which allows us to derive a valid inference that PL leaves out. However, there is another perspective which takes completeness to describe a certain limitation on what sorts of entailments hold between wfss in $\mathcal{L}^{\text{PL}}$, calling the notion of entailment in $\mathcal{L}^{\text{PL}}$ into question. We will close the chapter with an important consequence of completeness.

Corollary 5.2 If $\Gamma \vDash \varphi$, then there is a finite subset $\Lambda \subseteq \Gamma$ where $\Lambda \vDash \varphi$.

> *Proof:* Assume $\Gamma \vDash \varphi$. It follows by completeness that $\Gamma \vdash \varphi$, and so there is a derivation X of φ from Γ. Letting Γ_X be the set of premises which appear in X, it follows that $\Gamma_X \vdash \varphi$, and so $\Gamma_X \vDash \varphi$. Since X is finite, Γ_X is also finite, and so whenever $\Gamma \vDash \varphi$ there is a finite subset $\Lambda \subseteq \Gamma$ where $\Lambda \vDash \varphi$. □

Corollary 5.3 (Compactness) If every finite subset $\Lambda \subseteq \Gamma$ is satisfiable, then Γ is satisfiable.

> *Proof:* Assume for contraposition that Γ is unsatisfiable. It follows vacuously that $\Gamma \vDash \bot$, and so $\Lambda \vDash \bot$ by **Corollary 5.2** for some finite subset $\Lambda \subseteq \Gamma$. Thus there is some finite subset $\Lambda \subseteq \Gamma$ that is unsatisfiable. By contraposition, if every finite subset $\Lambda \subseteq \Gamma$ is satisfiable, then Γ is satisfiable. □

This property is referred to as COMPACTNESS. Recall that arguments were required to be finite sequences of wfss of $\mathcal{L}^{\text{PL}}$ where an argument is valid just in case its conclusion is a logical consequence of its premises. When we defined what it is for φ to be a logical consequence of Γ, we permitted Γ to be any set of wfss of $\mathcal{L}^{\text{PL}}$ including infinite sets. What compactness shows is that this additional permission does not add anything that would have been lost were logical consequence restricted to finite sets of wfss of $\mathcal{L}^{\text{PL}}$.

It is important to emphasize that PL COMPLETENESS does not entail that PL provides an exhaustive description of formal reasoning. Rather, PL only claims to exhaustively describe logical consequence in $\mathcal{L}^{\text{PL}}$ which in turn is constrained by the expressive resources which $\mathcal{L}^{\text{PL}}$ includes. At most, we may think of PL as providing a complete description of formal reasoning in $\mathcal{L}^{\text{PL}}$ given the logical forms that $\mathcal{L}^{\text{PL}}$ is able to capture.

In the following chapter we will extend the expressive resources of $\mathcal{L}^{\text{PL}}$ by adding constants, variables, predicates, and quantifiers, referring to this first-order language as $\mathcal{L}^{\text{FOL}}$. These additions will make it possible to regiment a host of valid arguments that we are unable to capture in $\mathcal{L}^{\text{PL}}$. After stipulating the syntax of $\mathcal{L}^{\text{FOL}}$ in a similar manner to $\mathcal{L}^{\text{PL}}$, we will also provide a semantics and natural deduction system FOL, developing its metalogic by establishing both the soundness and completeness of FOL. Despite the increase in expressive power of $\mathcal{L}^{\text{FOL}}$ and the logical strength of FOL, the methodology that we will follow will be the same as what we have already provided for $\mathcal{L}^{\text{PL}}$.

Chapter 6

Midterm Review

So as to preserve our beautiful correspondence between full semester weeks, problem sets, and book chapters, we hereby introduce this placeholder chapter for Week 7.

We await the day that the Gods of Time will smile upon us, granting the instructor time to typeset his midterm review guide!

Chapter 7

First-Order Logic

This chapter introduces the formal language called $\mathcal{L}^{\text{FOL}}$— pronounced, *the language of first-order logic*— which includes quantifiers like 'for all' ($\forall$) and 'there is' ($\exists$). In contrast to $\mathcal{L}^{\text{PL}}$ whose primitive symbols were limited to sentence letters and sentential operators, $\mathcal{L}^{\text{FOL}}$ includes *predicates* and *singular terms* among its primitive symbols. Whereas one-place predicates are terms for properties, two-place predicates are terms for relations, and n-place predicates are similar but relate n objects. Singular terms include *constants* which name objects and *variables* which may refer to any object. First-order logic is *first-order* because the quantifiers bind variables that are singular term (type e), and so range over a domain of objects rather than properties and relations as in *second-order logic*, or the higher-order entities of *higher-order logic* (e.g., the lambda calculus and simple type theories). For instance, $\mathcal{L}^{\text{FOL}}$ has the expressive power to quantify over people, animals, or things, but not over their various properties such as *being blue*, or relations like *being taller than*, etc.

Despite its limitations, first-order logic is extremely useful, nicely balancing the theoretical virtues of simplicity and intuitive appeal on the one hand with expressive power and logical strength on the other hand. Mastering first-order logic is also an important stepping stone to studying logics for more expressive languages. Although $\mathcal{L}^{\text{FOL}}$ will introduce a number of new expressive resources, all of the primitive symbols which we included before will be present in $\mathcal{L}^{\text{FOL}}$, and so $\mathcal{L}^{\text{FOL}}$ is said to be an *extension* of $\mathcal{L}^{\text{PL}}$. We will also be repeating the same methodology carried out before, and so much should seem familiar. Whereas this chapter develops the syntax for first-order logic by presenting $\mathcal{L}^{\text{FOL}}$ and using this language to regiment sentences and arguments in English, Chapter 8 will provide a semantics and definition of logical consequence for $\mathcal{L}^{\text{FOL}}$. Chapter 9 will repeat these two steps for a language with even more expressive power by including a primitive symbol for identity along with a stock of function symbols. We will then turn to present a proof system for first-order logic called FOL in Chapter 10 before establishing the soundness and completeness of FOL in Chapter 11 and Chapter 12, respectively. As is the case with the syntax and semantics for FOL, the proof system FOL itself will extend PL from before, and so the proofs of the soundness and completeness of FOL will similarly extend the work that we have already done.

7.1 Expressive Limitations

Consider the following argument:

A1. Every human is mortal.
A2. Socrates is human.
A3. Socrates is mortal.

In order to regiment this argument in $\mathcal{L}^{\text{PL}}$, we will need a symbolization key.

E: Every human is mortal.
H: Socrates is human.
M: Socrates is mortal.

Notice that there is no way to break down the sentences above into smaller parts. However, consider the resulting regimentation of the argument in $\mathcal{L}^{\text{PL}}$:

B1. E
B2. H
B3. M

It is easy to show that this argument is invalid. However, the argument in English is intuitively valid. Notice that although the English sentences given above include the same terms, this is not captured by the regimentation in $\mathcal{L}^{\text{PL}}$. Moreover, there is no better regimentation in $\mathcal{L}^{\text{PL}}$.

Here's another common case:

C1. All humans are mammals.
C2. All mammals are multi-celled organisms.
C3. All humans are multi-celled organisms.

This argument is intuitively valid since the conclusion follows from the premises in virtue of the logical form of the argument. However, were we to regiment this argument in $\mathcal{L}^{\text{PL}}$, the best we could do would look something like the following:

H: All humans are mammals.
M: All mammals are multi-celled organisms.
O: All humans are multi-celled organisms.

Again, there is no way to further decompose the sentences given above into smaller parts. However, the resulting argument in $\mathcal{L}^{\text{PL}}$ is invalid for the same reason:

D1. H
D2. $\underline{M}$
D3. O

Since there is nothing wrong with our regimentation when restricted to $\mathcal{L}^{\text{PL}}$, we may take issue with the language $\mathcal{L}^{\text{PL}}$ itself, concluding that we have simply reached the limit of what can be captured with the expressive resources which $\mathcal{L}^{\text{PL}}$ provides. Instead of carving arguments up into *sentences*, we have to look deeper into the sentences themselves in order to capture the validity of the English arguments considered above. In particular, we need to be able to regiment predicates like 'is human', 'is mortal', 'is a mammal', etc., as well as the quantifiers 'every' and 'all', as well as names like 'Socrates'. More generally, we should like to provide a language which can regiment any such argument at this level of logical resolution power. Providing the expressive resources to do so will be the ambition of this chapter.

As we will see, $\mathcal{L}^{\text{FOL}}$ is extremely powerful. Indeed, most of mathematics is developed in a first-order language. Understanding this language and its limits will prove to be an invaluable resource, shedding light on a wide range of theoretical applications as well as extending your powers of logical thinking in day-to-day reasoning. The development of first-order logics and their applications constitute some of the most important and influential theories that human beings have developed so far. Nevertheless, modern logic is still at its very beginning.

7.2 Primitive Expressions in $\mathcal{L}^{\text{fol}}$

Our first key idea will be to include *predicates* in $\mathcal{L}^{\text{FOL}}$. An easy way to think about predicates is as terms for properties: the predicate 'is red' stands for the property of being red. We use predicates to ascribe properties to objects. On their own, predicates are neither true nor false, though they are meaningful nevertheless. For instance, English speakers will know what the predicate 'is red' means and how to use it to make claims about the world. Nevertheless, it doesn't make sense to assign a truth-value to 'is red', 'is loved by John', 'is careful and quite', 'is between Sam and Mary in birth order', or any other predicate, however complex. Rather predicates are used to ascribe properties and relations to objects where those objects may or may not have the properties and relations which are ascribed to them.

In order to refer to objects, we will need another basic type of expression for *names*. For instance, the name 'Christoph' refers to a particular object, in this case a person. Accordingly, we may construct the sentence 'Christoph is Italian', where this may be true or false depending on whether the object that 'Christoph' names has the property that the predicate 'is Italian' expresses. Although names like 'Christoph' are meaningful on their own, they do not have truth-values any more than predicates have truth-values. Rather, it is by combining predicates with singular terms that we will construct *atomic sentences* which do have truth-values. Given the same sentential operators included in $\mathcal{L}^{\text{PL}}$, we may construct complex sentences which inherit their truth-values from their parts. Despite introducing a range of new symbols to the language, $\mathcal{L}^{\text{FOL}}$ will maintain the convention that only sentences have truth-values.

Given that we can construct sentences with nothing more than names and predicates, you might be wondering where the quantifiers fit in. After all, there is no grammatical way to append 'every', 'all', or 'some' to sentences like 'Christoph is Italian'. What is missing are the *variables*. In order to bring this out, consider the following version of H from before:

H': Everything is such that if it is human, then it is a mammal.

However stilted, this claim is perfectly intelligible and express exactly what H expressed before. Three observations are in order. Whereas H includes the plural terms 'humans' and 'mammals', we now have the singular predicates 'is human' and 'is a mammal'. For instance, whereas we may combine 'is human' with a name like 'Christoph' in order to produce the perfectly intelligible declarative sentence 'Christoph is human', the same cannot be said for the plural term 'humans' since 'Christoph humans' is nonsense.

Second, we may consider the role that 'it' plays in H'. Whereas a name like 'Christoph' refers to an object, the term 'it' also appears to indicate an object, but not particular object. Rather, 'it' is playing the role of a variable. Since both constants and variables refer to objects, we will refer to both as *singular terms*. Accordingly, 'it' may be combined with a predicate to form a sentence. In particular, we may combine 'it' with the predicates above to produce 'it is human' and 'it is a mammal'. On their own, 'it is human' and 'it is a mammal' are both atomic sentences though they include the *free variable* 'it', and so are *open sentences*. Nevertheless, these sentences may be combined with a conditional operator to form 'if it is human, then it is mammal' which is also an open sentence but not atomic. Appending 'Everything is such that' to the conditional sentence 'if it is human, then it is mammal' produces H'. Instead of speaking about a particular, H' says of each object that *it* is a mammal if *it* is human. For each object, both occurrences of 'it' refer to that object, at least until we move on to the next object since we are making a general claim about everything. Put otherwise, both occurrences of 'it' are *bound* by the quantifier 'Everything is such that'.

Third, we may observe that the perfectly natural occurrence of 'All' in H has been replaced with the somewhat cumbersome 'Everything is such that'. Whereas 'All' is a *generalized quantifier*, 'Everything is such that' is an attempt to express the first-order *universal quantifier* $\forall$ with the resources of English. These quantifiers differ in a number of important respects. In particular, generalized quantifiers like 'every', 'some', 'all', 'most', 'many', etc., combine with two descriptive terms in order to produce a grammatical sentence. By contrast, we may observe that 'Everything is such that' combines with the open sentence 'if it is human, then it is mammal' to produce H'. Although 'it' is unbound in 'if it is human, then it is mammal' when considered on its own, both occurrences 'it' are bound by 'Everything is such that' in H'. Thus H' is a *closed sentence* since it does not include any unbound variables.

Although first-order quantifiers can be approximated within English, generalized quantifiers are often more natural to use in English. Despite this disparity, generalized quantifiers are much more complicated than first-order quantifiers, making them not as useful for systematic theorizing. Nevertheless, we may use first-order quantifiers to regiment claims in English which are expressed with generalized quantifiers without too much trouble. For these reasons, we will not include formal analogues of generalized quantifiers in $\mathcal{L}^{\text{FOL}}$.

Given this overview of the new expressive resources that we will include in $\mathcal{L}^{\text{FOL}}$, the following sections will provide a much more details introduction to each of the elements that we have touched on above. We will begin again with names and variables.

7.2.1 Singular Terms

In English, a proper name is a term that refers to a person, place, or thing. For instance, 'Christoph' refers to Christoph. In $\mathcal{L}^{\text{FOL}}$, we will take lower-case letters 'a', 'b', 'c', ..., 's', 't' subscripted by a numeral to be the CONSTANTS of $\mathcal{L}^{\text{FOL}}$. For convenience, we will drop the subscript '$_0$' from constants, using subscripts only as needed. Constants are singular terms since they refer to an object and will be used to regiment names in English.

In English, the same name may refer to different objects in different contexts. Whereas 'Willard' might pick out different people in different contexts who have the same name, we will avoid this ambiguity in $\mathcal{L}^{\text{FOL}}$ by insisting that constants refer to at most one individual on any given interpretation. It is for this reason that we have included infinitely many constants (with the help of the subscripts) so that we don't run out of names.

There are other ways to refer to individuals besides using names. In particular, we often use descriptions. Whereas it is indeterminate to say 'the person standing next to the podium' if more than one person is standing next to the podium or nobody is, this expression succeeds in picking out a unique individual if there is just one person standing next to the podium. Or to take another case, we may use 'the tallest person in the room' to refer to a unique individual assuming that there are at least some people in the room and that one of them is taller than all the others. Such terms are referred to as DEFINITE DESCRIPTIONS. Although the definite descriptions that succeed in referring to unique objects are singular terms, we will not need to include primitive terms in our language to express definite descriptions. Instead, we will construct definite descriptions with the resources of $\mathcal{L}^{\text{FOL}}$.

Another important type of singular term are the VARIABLES that we mentioned above. In English, terms such as 'it', 'her', etc., often play this role. In $\mathcal{L}^{\text{FOL}}$, we will use the lower-case letters 'u', 'v', 'w', 'x', 'y', 'z', from the end of the alphabet with optional subscripts as variables. We will use variables in combination with the quantifiers in order to make claims about every object, or about some object, specifying what properties those objects have. However, in order to make claims about objects, we will need more than singular terms.

7.2.2 Predicates

A one-place predicate is used to express a property of individuals. For instance, 'is hungry' is a one-place predicate which, in combination with a singular term, forms an atomic sentence such as 'Kate is hungry' and 'She is hungry'. In claiming that a predicate is one-place, we are indicating that just one singular term is needed to construct an atomic sentence. This corresponds to the idea that the predicate expresses a property that a single object may have or fail to have. For instance, either Kate is hungry or she isn't. Accordingly, the sentence 'Kate is hungry' is true or false, but the truth-value of this sentence only depends on Kate and whether she is hungry or not, and so no other singular terms need to be provided.

A two-place predicate is used to express a relation between individuals. For instance, the two-place predicate 'is taller than' expresses a height relation between objects. Given a two-place predicate together with two singular terms, we may make a claim which is either true or false. For instance, 'Sam is taller than John' is true just in case Sam is taller than John. It is worth noting that it is possible to use a two-place predicate together with two instances of the same singular term. Just as we may say that 'Jay loves Cary', we may say, 'Jay loves Jay', or somewhat more naturally, 'Jay loves himself'. Although there is just one object at issue in such cases, the predicate 'loves' is two-place all the same.

The singular terms that a predicate requires to form a formula are referred to as the arguments of that predicate. Whereas one-place predicates require one argument and two-place predicates require two arguments, in general we may consider n-place predicates which require n arguments. Although it is easy to generalize, it is uncommon to consider more than three-place predicates in English. For instance, we may say that Sue is between Henry and Tom in birth order, where 'between' has three arguments. Even if it is uncommon to do so in English, nothing stops us from including predicates with more than three arguments in $\mathcal{L}^{\text{FOL}}$. It is convenient to refer to the number of arguments that a predicate has as the adicity (also called arity) of that predicate, where this comes from the convention of referring to one-place predicates as monadic, two-place predicates as dyadic, three-place predicates as triadic, an predicates with more than one place as polyadic. We will also include 0-place predicates which we will refer to as sentence letters and are familiar from before.

In order to regiment n-place predicates in $\mathcal{L}^{\text{FOL}}$, we will use capital letters A^n through Z^n where the superscripts indicate the arity of the predicate. As above, we will drop the superscript '0' so that $A_i, B_i, C_i, \ldots$ and the like are the same sentence letters from before. We will also drop superscripts when the arity is indicated by the number of arguments. For instance, it is clear that 'A' is a tryadic predicate in '$Axcy$' and a dyadic predicate in 'Aaz', though we would do well to use different letters if we needed to consider both at once. Symbolization keys may also rely on the number of arguments to indicate the arity and interpretation as follows:

Ax:	x is angry.	T_1xy:	x is at least as tall as y.
Hx:	x is happy.	T_2xy:	x is at least as tough as y.
Sxy:	x is shorter than y.	$Bxyz$:	y is between x and z.

It is worth emphasising that the variables above do no further work than to specify the order of the arguments in a given predicate. Since monadic predicates only have one argument, the variable merely indicates its adicity. By contrast, the order of the variables also matters for dyadic predicates like T_1. For instance, we could have introduced the predicate:

T_3xy: y is at least as tall as x.

Having reversed the order of the variables above, 'T_3' expresses the converse of the relation expressed by 'T_1'. This makes a big difference. However, it would not have mattered if we were to have replaced 'x' with some other variable such as 'z' or 'y_{13}'.

In order to put these predicates to work, we may now turn to combine the primitive symbols that we have included in $\mathcal{L}^{\text{FOL}}$ in order to produce atomic sentences.

7.2.3 Atomic Sentences

Consider the following sentences:

E1. Cordelia is angry.
E2. Cordelia is shorter than Hamlet.
E3. If Cordelia is angry, then so are Hamlet and Macbeth.
E4. Macbeth is at least as tall and tough as Hamlet.

We have already provided the predicates that we will need to regiment the sentences above, but we have not introduced any constants. Consider the following:

c: Cordelia	Ax: x is angry.
h: Hamlet	Sxy: x is shorter than y.
m: Macbeth	T_1xy: x is at least as tall as y.
	T_2xy: x is at least as tough as y.

Given these resources, we may regiment sentence E1 as: Ac. Since the 'x' in the symbolization key entry for 'Ax' is just a placeholder, we have replaced the variable with the constant which names Cordelia, resulting in an atomic sentence. Whereas 'Ac' includes the monadic predicate 'A', atomic sentences may include predicates of any adicity. More generally, an ATOMIC SENTENCE of $\mathcal{L}^{\text{FOL}}$ is an n-place predicate followed by n constants where n is any natural number. Accordingly, 0-place predicates are atomic sentences of $\mathcal{L}^{\text{FOL}}$ which, as above, we will refer to as sentence letters. Or to take another example from above, the sentence E2 can be regimented with the dyadic predicate 'S' as: Sch.

By contrast with sentences E1 and E2, sentence E3 is not atomic on account of including sentential operators. Nevertheless, we may regiment this sentence by identifying its atomic parts: Ac, Ah, and Am. These atomic sentences will play a similar role to the sentence letters included in $\mathcal{L}^{\text{PL}}$. In particular, we may regiment sentence E3 by: $Ad \to (Ag \wedge Am)$.

Sentence E4 is similar though the atomic sentences are a little trickier to identify. We begin by observing that two things are said of Marybeth and Gregor: first, Marybeth is at least as tall as Gregor; and second, Marybeth is at least as tough as Gregor. Thus we can build the atomic sentences: T_1mg and T_2mg. The last step is to put these together: $T_1mg \wedge T_2mg$.

7.2.4 Open Sentences

Consider the following sentence:

F1. Either it is crimson or ruby since she wants to impress him.

In contemplating the predicates that we will need, we may introduce the following:

Cx: x is crimson.
Rx: x is ruby.
Wxy: x wants to impress y.

Since there are no names in the sentence F1 above, we do need to include constants in our symbolization key. Instead, we can get by with variables alone. In particular, we may regiment sentence F1 by first identifying its atomic parts. To do so, we may observe that the object in question is said to be one of two colors, and that the person in question wants to impress someone else. Since the same object is either crimson or ruby, we will use the same variable for each: Cx, Rx. We use different variables for 'she' and 'him', where this yields: Wyz. Putting these pieces together, we get: $(Cx \vee Rx) \wedge Wyz$.

Although Cx, Rx, and Wyz are atomic, these expressions are not sentences since it is hard to assign truth-values to such expressions since there is no telling which objects the variables refer to. Matters would be different if the variables were bound by quantifiers since then we would know that we need only find some object which bears the relevant property or relation, or else that every object must do so. Since these atomic expressions include variables that are not bound by quantifiers, we will be refer to such expressions as *open sentences* and the unbound variables in them as *free variables*. Although the precise definitions will be given shortly, it is important to get a sense of the open sentences and the free variables that then include in order to appreciate the role that the quantifiers play in $\mathcal{L}^{\text{FOL}}$.

In order to get a better sense of open sentences as well as some of the subtleties of regimenting sentences in $\mathcal{L}^{\text{FOL}}$, consider the closely related sentence:

G1. Either it is crimson or ruby since she likes those colors.

Now we will need the following symbolization key:

Hxy: x has the color y.
Lxy: x likes y.
c: Crimson
r: Ruby

Instead of taking 'is crimson' and 'is ruby' to be monadic predicates, we have included constants for each together with a new dyadic predicate for 'has the color'. This is useful since now we can say that she likes each color by using its name. Thus we have the atomic open sentences: Hxc, Hxr, Lyc, and Lyr. We can then build the following regimentation: $(Hxc \vee Hxr) \wedge (Lyc \wedge Lyr)$. Despite how similar the sentences F1 and G1 are to each other, their regimentations are very different, and by no means exhaustive.

We may conclude this subsection with a final example:

H1. If it is a dolphin, then it is a mammal and he won't eat it.

Dx: x is dolphin.
Mx: x is mammal.
Eyx: y will eat x.

As before, we may build the atomic open sentences Hx, Mx, and $\neg Eyx$, drawing on these to construct: $Hx \to (Mx \wedge \neg Eyx)$. By itself, this open sentence appears to refer to some unnamed individual, however, this hardly captures its power. Rather, it is by combining open sentences with quantifiers that we may make general claims of considerable interest. Indeed, such claims are common throughout mathematics, the sciences, and day-to-day life.

7.2.5 Quantifiers

We are now ready to introduce the first-order quantifiers. Unlike singular terms and predicates, quantifiers are a new kind of unary logical operator similar to negation. Consider the sentences:

I1. Someone is happy.	Hx: x is happy.
I2. Nobody is as guilty as Donald.	Gxy: x is as guilty as y.
I3. Everybody loves somebody.	Lxy: x loves y.
	d: Donald.

It might be tempting to regiment sentence I1 as: Hx. After all, if 'It is happy' is true, then surely someone— whoever 'It' refers to— must be happy. Tempting as this may be, these are very different claims. In order to see how these claims differ, we may consider who these claims *concern* or *are about*. Although it may not be clear who exactly, 'It is happy' is about whoever 'It' refers to where this is some particular individual. By contrast, claiming that someone is happy is not about any particular individual at all. For instance, suppose there are many happy people but 'It' happens to be used to refer to a broken table sitting in the corner. Although sentence I2 may be true, 'It is happy' is false.

Instead of asserting anything about a particular, quantifiers make claims about a *domain* of individuals which we will introduce in the following chapter. Depending on the context of assertion, the domain may differ. Nevertheless, quantified claims may be understood to make claims about the domain which includes all of the individuals that there are. By contrast, open sentences make claims about particular individuals. These are important differences, and we will have a lot more to say about them when we introduce the semantics for $\mathcal{L}^{\text{FOL}}$.

Although the sentence 'It is happy' does not say the same thing as sentence I1, their regimentations start off the same. Given the symbolization key above, 'It is happy' is just: Hx. What is needed to regiment I1 is to add that *there is something such that* it is happy. We will accomplish this by appending an EXISTENTIAL QUANTIFIER $\exists x$, where this yields: $\exists x Hx$. As defined below, the quantifier *binds* the variable x in Hx. Since the result does not contain any free variables, $\exists x Hx$ is not an open sentence, but rather a sentence of $\mathcal{L}^{\text{FOL}}$.

In order to regiment sentence I2, we may ask what this sentence says. One way to read this sentence is that it is not the case that someone is as guilty as Donald. Unpacking this claim even further: it is not the case that something is such that *it is as guilty as Donald*. However cumbersome, this reading makes the italicized open sentence easy to regiment: Gxd. Although Gxd contains the constant d, it also contains the free variable x which we may bind with an existential quantifier as before: $\exists x Gxd$. We may then negate the result: $\neg\exists x Gxd$.

Although this provides a fine regimentation of sentence I2, there is another natural reading. Instead of beginning with negation, we may take sentence I2 to express that everyone is not as guilty as Donald. Put otherwise, everything is such that it is not the case that *it is as guilty as Donald*. Here we begin as before with Gxd but then immediately add negation: $\neg Gxd$. Note that this sentence is open since it includes the free variable x. Finally, we add the UNIVERSAL QUANTIFIER $\forall x$, where this yields: $\forall x \neg Gxd$. It turns out that this reading is logically equivalent to the first regimentation of sentence I2 considered above.

In general, $\forall x\varphi$ is logically equivalent to $\neg\exists x\neg\varphi$, and similarly, $\exists x\varphi$ is logically equivalent to $\neg\forall x\neg\varphi$. This means that any sentence which can be regimented with a universal quantifier can be regimented with an existential quantifier together with an appropriate number of negation signs, and *vice versa*. Even though there is no logical difference in regimenting a sentence with one quantifier rather than the other, sometimes one regimentation might seem more natural than the other. Other times, the difference between regimenting with an existential as opposed to a universal quantifier will make no difference at all.

It remains to regiment sentence I3. This says of everything that it loves something. Focusing on 'it loves something', we may take this to mean that there is something such that *it loves that thing*. Thus we have: everything is such that there is something such that it loves that thing. Notice that without the help of variables, it is easy to lose track of which quantifier is binding which variable. It is for this reason that the method that we have employed above of unpacking the sentence in stilted English is of limited utility. Nevertheless, we do need to make sure we know what the claim is saying, and sometimes it helps to write it out a few different ways in English, proceeding in stages as we have above.

Instead of using English to successively unpack I3, we may employ a similar process in $\mathcal{L}^{\text{FOL}}$ itself. For instance, we may begin by taking sentence I3 to say that something is such that it loves somebody which we may partially regiment as $\forall xLSx$ by temporarily introducing 'LSx' for 'x loves somebody'. Next we may take 'LSx' to mean 'there is some y where x loves y' which we may regiment: $\exists yLxy$. Replacing 'LSx' in our first pass regimentation with '$\exists yLxy$' yields the final product: $\forall x\exists yLxy$. In general, sentences with mixed quantifiers are tricky and require a lot of care to get right. Even if some cases you can immediately intuit a correct regimentation, it is important to learn how to systematically decompose complex sentences into simpler parts before putting the pieces back together. This will take some practice.

Having introduced the new primitives symbols included in $\mathcal{L}^{\text{FOL}}$, we may now proceed to define the sentences of $\mathcal{L}^{\text{FOL}}$ a little more rigorously. Nevertheless, the informal remarks that we have made so far should help to provide an intuitive target for our definition to strike.

7.3 The Well-Formed Formulas

Whereas the well-formed sentences (wfss) of $\mathcal{L}^{\text{PL}}$ were defined recursively in terms of the primitive symbols included in $\mathcal{L}^{\text{PL}}$, this approach will not do $\mathcal{L}^{\text{FOL}}$. In order to see why, it is important to appreciate that the wfss of any language are intended to include all and only those expressions which may ultimately be assigned a truth-value by an interpretation. If the wfss of $\mathcal{L}^{\text{FOL}}$ are to be built up in similar manner to the way they were in $\mathcal{L}^{\text{PL}}$, then there will invariably be stages of construction which include the unbound variables that are then bound by quantifiers introduced in later stages of construction. For instance, if we are to construct $\forall x\exists yLxy$ as above, we must start with Lxy, building up to $\exists yLxy$, and only then add the final quantifier. However, as mentioned above, there is no easy way to interpret open sentences such as Lxy and $\exists yLxy$ which includes a free variable.

Rather that admitting sentences which cannot be assigned truth-values by an interpretation of our language, we will arrive at the well-formed sentences of $\mathcal{L}^{\text{FOL}}$ by first defining the more modest notion of a *well-formed formula* of $\mathcal{L}^{\text{FOL}}$, or wff for short. This definition will proceed much as before. Only then will we move to identify those wffs of $\mathcal{L}^{\text{FOL}}$ which do not have free variables and so may be said to be well-formed sentences of $\mathcal{L}^{\text{FOL}}$ whose truth-values will turn out not depend on anything besides the interpretation of $\mathcal{L}^{\text{FOL}}$. As we will see in the following chapter, not all wffs of $\mathcal{L}^{\text{FOL}}$ may claim this virtue.

Collecting the elements that we have introduced so far, the primitive symbols of $\mathcal{L}^{\text{FOL}}$ include:

n-place predicates for $n \geqslant 0$ with subscripts, as needed	$A^n, B^n, C^n, \ldots, Z^n$ $A^n_1, B^n_1, Z^n_1, A^n_2, A^n_{25}, J^n_{375}, \ldots$
constants with subscripts, as needed	$a, b, c, \ldots, t$ $a_1, w_4, h_7, m_{32}, \ldots$
variables with subscripts, as needed	w, x, y, z $x_1, y_1, z_1, x_2, \ldots$
sentential operators	$\neg$, $\wedge$, $\vee$, $\rightarrow$, $\leftrightarrow$
quantifiers	$\forall, \exists$
parentheses	(,)

In a similar manner to the definition of the wfss of $\mathcal{L}^{\text{PL}}$, we will recursively define the WELL-FORMED FORMULAS (wffs) of $\mathcal{L}^{\text{FOL}}$ as follows:

1. $\mathcal{F}^n\alpha_1, \ldots, \alpha_n$ is a wff of $\mathcal{L}^{\text{FOL}}$ if $\mathcal{F}^n$ is an n-place predicate of $\mathcal{L}^{\text{FOL}}$ and $\alpha_1, \ldots, \alpha_n$ are singular terms (i.e., variables or constants) of $\mathcal{L}^{\text{FOL}}$.
2. For any wffs φ and ψ of $\mathcal{L}^{\text{FOL}}$ and variable α of $\mathcal{L}^{\text{FOL}}$:
   (a) $\exists\alpha\varphi$ is a wff of $\mathcal{L}^{\text{FOL}}$;
   (b) $\forall\alpha\varphi$ is a wff of $\mathcal{L}^{\text{FOL}}$;
   (c) $\neg\varphi$ is a wff of $\mathcal{L}^{\text{FOL}}$;
   (d) $(\varphi \wedge \psi)$ is a wff of $\mathcal{L}^{\text{FOL}}$;
   (e) $(\varphi \vee \psi)$ is a wff of $\mathcal{L}^{\text{FOL}}$;
   (f) $(\varphi \rightarrow \psi)$ is a wff of $\mathcal{L}^{\text{FOL}}$; and
   (g) $(\varphi \leftrightarrow \psi)$ is a wff of $\mathcal{L}^{\text{FOL}}$.
3. Nothing else is a wff of $\mathcal{L}^{\text{FOL}}$.

As brought out in Chapter 1, the definitions above are strictly speaking nonsense. This is because the clauses above *use* rather than *mention* the meta-variables φ and ψ for wffs, as well as new meta-variables for n-place predicates $\mathcal{F}^n$ and singular terms $\alpha_1, \ldots, \alpha_n$. Moreover, it would not help to simply add quotes since the meta-variables that we have used above do not belong to $\mathcal{L}^{\text{FOL}}$. This is what motivated the introduction of corner quotes before, and we may reproduce a similar tactic here. However, instead of doing so, we will follow the common convention of relying on the reader to know where the corner quotes are supposed to go. If you do not remember how to do this, look back to §1.9.3 to review before proceeding.

As before, we will refer to the clauses included in the definition of the wffs of $\mathcal{L}^{\text{FOL}}$ as the COMPOSITION RULES for the wffs of $\mathcal{L}^{\text{FOL}}$ since they tell us how to *compose* wffs from the primitive symbols included in $\mathcal{L}^{\text{FOL}}$. In particular, the first clauses allows us to construct ATOMIC WFFS where the other clauses allow us to construct COMPLEX WFFS. Since we have already seen a number of paradigm cases of wffs in $\mathcal{L}^{\text{FOL}}$, it will help to consider some examples which do not follow the composition rules above. For the sake of readability, we will drop subscripts and superscripts below and throughout much of what follows:

J1. $aFbx$.
J2. $\forall aFbx$.
J3. $\forall yFbx$.
J4. $\forall xFbx$.
J5. $\exists y\forall xFbx$.
J6. $GFbx$.
J7. $G \wedge Fbx$.

The expression in J1 is complete nonsense, or to use the definition given above, J1 is *not* a wff of $\mathcal{L}^{\text{FOL}}$. The reason is that a constant a occurs before the predicate F, but there is no way to achieve this given the composition rules provided above. Were we to remove this misplaced constant, the result would be Fbx which is a wff of $\mathcal{L}^{\text{FOL}}$.

The expression in J2 is not a wff of $\mathcal{L}^{\text{FOL}}$ since the constant a follows the quantifier $\forall$. There is no way to achieve this given our composition rules, and for good reason. For instance, consider the English analogue 'for every Sally, Jim loves it'. Although we might be able to contrive a way to make sense of this in English— something natural languages are very good at— our initial point remains: there is no way to construct J2 given the composition rules for $\mathcal{L}^{\text{FOL}}$, and so J2 is not a wff of $\mathcal{L}^{\text{FOL}}$.

The expression in J3 is a wff of $\mathcal{L}^{\text{FOL}}$. Perhaps this comes as a surprise. After all, the binding variable y which occurs after the quantifier differs from what we might expect to bind the variable x. Accordingly, the quantifier does not make any substantive contribution to the sentence. Be this as it may, we may construct *J3* all the same. We may simulate this effect in English by saying: everything is such that Jim loves Raha. Here, the quantifier 'everything is such that' does no substantive work, where something similar may be said for $\forall y$ in J3.

The expression in J4 is also a wff of $\mathcal{L}^{\text{FOL}}$, but this time the quantifier succeeds in binding the variable x. For instance, we might use this wff to regiment the claim 'Everything is such that Jim loves it', or more naturally, 'Jim loves everything'.

The expression in J5 is a wff of $\mathcal{L}^{\text{FOL}}$, but includes an extra quantifier that does no work. This is because the binding variable y does not bind any variables. Perhaps surprisingly, the same may be said were we to replace y with x. We will have more to say about such cases in the following subsection where we will introduce the notion of *scope*.

The expression in J6 is not a wff of $\mathcal{L}^{\text{FOL}}$ since it includes two concatenated predicates F and G. There is no more latitude for this in $\mathcal{L}^{\text{FOL}}$ than there is for concatenating two sentence letters in $\mathcal{L}^{\text{PL}}$. After all, this would be like saying in English: is red Jim loves it.

The expression in J7 is not a wff of $\mathcal{L}^{\text{FOL}}$, but only for the pedantic reason that we have not included parentheses. Leaving such pedantry to the side, we will go on dropping outermost parentheses when no ambiguity results. For instance, were to wish to bind the variable x with a quantifier, parentheses would be required. Although both $(\forall xG \wedge Fbx)$ and $\forall x(G \wedge Fbx)$ are wffs of $\mathcal{L}^{\text{FOL}}$, the quantifier only succeeds in binding the variable x in the latter. This provides some indication of the role that parentheses play in defining the scope of a quantifier.

7.4 Quantifier Scope

Given a quantified wff of $\mathcal{L}^{\text{FOL}}$ of the form $\exists\alpha\varphi$ or $\forall\alpha\varphi$, we may define φ to be the SCOPE of each of these quantifiers. In considering the composition rules above, the scope of a quantifier is the wff to which that quantifier is applied. Although it is easy to identify the scope of a quantifier when the quantifier is the main connective, this is not always the case.

In the sentence $\exists xGx \to Gl$, the scope of the existential quantifier is the expression Gx. Would it make a difference if the scope of the quantifier were the whole sentence as in $\exists x(Gx \to Gl)$? In order to answer this question, consider the following symbolization key:

Gx: x is a guitarist.
l: Lenny.

Given this key above, $\exists xGx \to Gl$ reads: if there is some guitarist, then Lenny is a guitarist. By contrast, $\exists x(Gx \to Gl)$ reads: something is such that if it is a guitarist, then Lenny is a guitarist. Recall that the material conditional is true any time the antecedent is false. Let the constant j denote Jack who we may assume is not a guitarist. It follows that the sentence $Gj \to Gl$ is true because Gj is false. Since Jack is such that if he is a guitarist then Lenny is a guitarist, it follows that $\exists x(Gx \to Gl)$ is true. The sentence is true because there is a non-guitarist, regardless of whether Lenny is a guitarist. This may strike you as strange.

It turns out that $\exists xGx \to Gl$ is logically equivalent to $\forall x(Gx \to Gl)$, and $\exists x(Gx \to Gl)$ is logically equivalent to $\forall xGx \to Gl$. This oddity does not arise with other connectives, nor does it arise if the variable only occurs in the consequent. For example, $\exists xGx \wedge Gl$ is logically equivalent to $\exists x(Gx \wedge Gl)$, and $Gl \to \exists xGx$ is logically equivalent to $\exists x(Gl \to Gx)$. What this brings out is yet another unusual features of the material conditional. Nevertheless, these examples indicate just how important it is to be clear about the scope of a quantifier.

Another type of case to watch out for is when quantifiers occur within each other's scope. For instance, suppose we have an open sentence $\forall x(Lxy \to Px)$ which includes the free variable y. Now suppose we conjoin this sentence with Kxy, wrapping the result in two further quantifiers to produce the wfs $\forall y\exists x(Kxy \wedge \forall x(Lxy \to Px))$. Since $\forall x$ binds the x variables in the innermost parentheses, those variables are not also bound by the $\exists x$ quantifier, though the x in Kxy is bound by this latter existential quantifier. Nevertheless, the existential quantifier has all of $Kxy \wedge \forall x(Lxy \to Px)$ as its scope.

7.4.1 First-Order Sentences

Recall that a declarative sentence (*sentence* for short) is an expression that can either be true or false on an interpretation. In $\mathcal{L}^{\text{PL}}$, every wfs could be assigned a truth-value given an interpretation, thereby justifying the name 'wfs of $\mathcal{L}^{\text{PL}}$'. By contrast, not all wffs of $\mathcal{L}^{\text{FOL}}$ have truth-values given an interpretation alone. For instance, consider the following:

Lxy: x loves y
b: Boris

The expression Lzz is an atomic wff since a two-place predicate is followed by two singular terms, in this case both of which are instances of the variable z. We may then ask what it would mean for Lzz to be true. For instance, perhaps it means that z is self loving in the same way that Lbb means that Boris loves himself. However, since z is a variable, it does not name something the way that a constant like b does. Whereas there is a clear way to interpret constants, there is no equally determinate way to interpret variables.

In order to be able to say which wffs of $\mathcal{L}^{\text{FOL}}$ can be assigned truth-values and which cannot, it will help to provide precise definitions of some of the notions that we made intuitive use of above. To begin with, we may provide the following recursive definition of FREE VARIABLES:

1. α is free in $\mathcal{F}^n\alpha_1, \ldots, \alpha_n$ if $\alpha = \alpha_i$ for some $1 \leqslant i \leqslant n$ where α is a variable, $\mathcal{F}^n$ is an n-place predicate, and $\alpha_1, \ldots, \alpha_n$ are singular terms.
2. If φ and ψ are wffs of $\mathcal{L}^{\text{FOL}}$ and α and β are variables, then:
   (a) α is free in $\exists\beta\varphi$ if α is free in φ and $\alpha \neq \beta$;
   (b) α is free in $\forall\beta\varphi$ if α is free in φ and $\alpha \neq \beta$;
   (c) α is free in $\neg\varphi$ if α is free in φ;
   (d) α is free in $(\varphi \wedge \psi)$ if α is free in φ or α is free in ψ;
   (e) α is free in $(\varphi \vee \psi)$ if α is free in φ or α is free in ψ;
   (f) α is free in $(\varphi \rightarrow \psi)$ if α is free in φ or α is free in ψ;
   (g) α is free in $(\varphi \leftrightarrow \psi)$ if α is free in φ or α is free in ψ;
3. Nothing else is a free variable.

Observe the manner in which the definition above follows the same recursive structure as the composition rules for $\mathcal{L}^{\text{FOL}}$. Given any wffs of the form $\exists\alpha\varphi$ or $\forall\alpha\varphi$, we may refer to α as the BINDING VARIABLE of these quantifiers where every free occurrence of α in φ is BOUND by the quantifiers $\exists\alpha$ and $\forall\alpha$ in $\exists\alpha\varphi$ and $\forall\alpha\varphi$, respectively. Accordingly, quantifiers BIND all free occurrences of their binding variable which occur within their scope.

A wff of $\mathcal{L}^{\text{FOL}}$ is a WELL-FORMED SENTENCE (wfs) of $\mathcal{L}^{\text{FOL}}$ if it does not include any free variables, and an OPEN SENTENCE of $\mathcal{L}^{\text{FOL}}$ otherwise. Since there are wffs of $\mathcal{L}^{\text{FOL}}$ that include free variables, not all wffs are wffs of $\mathcal{L}^{\text{FOL}}$. Consider the examples:

K1. $\forall x \forall x(Ex \vee Dy) \to \exists z(Rzx \to Lzx)$.
K2. $\forall x(\forall x(Ex \vee Dy) \to \exists z(Rzx \to Lzx))$.
K3. $\exists y \forall x(\forall x(Ex \vee Dy) \to \exists z(Rzx \to Lzx))$.

The scope of the first universal quantifier on the left in K1 is $\forall x(Ex \vee Dy)$. Although x occurs in $\forall x(Ex \vee Dy)$, it is a bound occurrence. Rather, y is the only free variable in $\forall x(Ex \vee Dy)$. Since $y \neq x$, we may observe that y remains free in $\forall x \forall x(Ex \vee Dy)$. Moving to the consequent of K1, both occurrences of z are bound, and neither occurrence of x are bound. Accordingly, K1 is a wff of $\mathcal{L}^{\text{FOL}}$, but not a sentence of $\mathcal{L}^{\text{FOL}}$.

We may change the scope of the universal quantifier on the far left in K1 by adding an additional pair of parentheses as given in K2. Now the scope of the left most universal quantifier is $\forall x(Ex \vee Dy) \to \exists z(Rzx \to Lzx)$. Whereas x is bound by the universal quantifier in the antecedent, x is free in the consequent, and so only these latter occurrences of x are bound by the outermost universal quantifier in K2. Nevertheless, y remains free throughout, and so K2 is not a sentence of $\mathcal{L}^{\text{FOL}}$ though it is a wff of $\mathcal{L}^{\text{FOL}}$.

In order to bind the free occurrence of y in K2, we may add a quantifier whose binding variable is y as given by K3. Since K3 does not include any free variables, K3 is a sentence of $\mathcal{L}^{\text{FOL}}$ and so may be assigned a truth-value given an appropriate interpretation of $\mathcal{L}^{\text{FOL}}$.

7.5 First-Order Regimentation

We now have the basic elements of $\mathcal{L}^{\text{FOL}}$ in place. Translating more complicated sentences will only be a matter of knowing the right way to combine predicates, constants, quantifiers, variables, and sentential connectives. Consider these sentences and symbolization key:

L1. Every coin in my pocket is a loonie.
L2. Some coin on the table is a dime.
L3. Not all the coins on the table are loonies.
L4. None of the coins in my pocket are dimes.

Cx: x is a coin.
Px: x is in my pocket.
Tx: x is on the table.
Lx: x is a loonie.
Dx: x is a dime.

Sentence L1 is most naturally translated with a universal quantifier. The universal quantifier says something about everything, not just about coins, or the coins in my pocket. Accordingly, we may take L1 to say that, for anything, *if* it is a coin and in my pocket, *then* it is a loonie. So we can translate it as $\forall x((Cx \wedge Px) \to Lx)$.

Since sentence L1 is about coins that are both in my pocket *and* that are loonies, it might be tempting to translate it using a conjunction. However, the sentence $\forall x((Cx \wedge Px) \wedge Lx)$ would mean that everything is a coin in my pocket and a loonie. It would also be wrong to regiment sentence L1 as $\forall x(Cx \to (Px \wedge Lx))$ since this say that everything that is a coin is in my pocket and a loonie. This is a very strong claim, and is unlikely to be true since there are a lot of coins out there which are neither in my pocket or loonies.

These examples bring out the idea of *restricting* a universal quantifier. Since saying something about everything in an unrestricted way is only very rarely something that we intend to do, universal claims almost always take the following form:

M1. $\forall \alpha(\varphi(\alpha) \to \psi)$.

Here $\varphi(\alpha)$ is a wff of $\mathcal{L}^{\text{FOL}}$ in which α occurs as a free variable, and so α as it occurs in $\varphi(\alpha)$ is bound by the quantifier $\forall \alpha$. Accordingly, the sentence M1 says that ψ holds of everything that satisfies the condition φ. Although we have not required ψ to also include α as a free variable, it is typical that ψ would include free occurrences of α. For instance, in the case above, we compared taking $Cx \wedge Px$ to restrict the quantifier with taking just Cx to restrict the quantifier. Whereas the former allows us to make universal claims about just the coins in my pocket, the latter allows us to make claims about all coins.

Sentence L2 is most naturally translated with an existential quantifier. It says that there is some coin which is both on the table and which is a dime. So we can translate it as $\exists x(Cx \wedge (Tx \wedge Dx))$. Notice that we did not use a conditional to restrict the existential quantifier in the same way as we did with the universal quantifier. Instead, we used conjunction to say that there is at least one thing which is a coin and moreover it is on the table and is a dime. Using conjunction with an existential quantifier is a common pattern.

What would it mean to write $\exists x(Cx \to (Tx \wedge Dx))$? This says that there is something which is a dime on the table *if it is a coin*. Suppose that there are no coins that are dimes on the table, but there is at least one thing which is not a coin, say the planet Jupiter which we will regiment by the constant j. Since the planet Jupiter is not a coin, it follows that $Cj \to (Tj \wedge Dj)$ is true since the antecedent Cj is false. As a result, $\exists x(Cx \to (Tx \wedge Dx))$ is true even though we have assumed that there are no coins that are dimes on the table. More generally, whenever there is something that is not a coin, $\exists x(Cx \to (Tx \wedge Dx))$ will be true, making this an extremely weak claim. Although the conditional is often used to restrict a universal quantifier, a conditional within the scope of an existential quantifier results in extremely week claims which we almost never intend to assert. Thus it's a good rule of thumb that you should avoid putting conditionals in the scope of existential quantifiers.

Sentence L3 can be paraphrased as, 'It is not the case that every coin on the table is a loonie'. So we can translate it as $\neg\forall x((Cx \wedge Tx) \rightarrow Lx)$. Alternatively, we paraphrase sentence L3 as, 'Some coin on the table is not a loonie'. We would then translate sentence L3 as $\exists x((Cx \wedge Tx) \wedge \neg Lx)$. Despite their superficial differences, these two regimentations are logically equivalent. More generally, $\neg\forall x\varphi$ and $\exists x\neg\varphi$ are logically equivalent, as are $\neg(\varphi \rightarrow \psi)$ and $\varphi \wedge \neg\psi$. This is something that we will prove to in the following chapter.

Sentence L4 can be paraphrased as, 'It is not the case that there is a coin in my pocket that is a dime'. This can be regimented by $\neg\exists x(Cx \wedge (Px \wedge Dx))$. It might also be paraphrased as 'Every coin in my pocket is not a dime' which we may regiment by $\forall x((Cx \wedge Px) \rightarrow \neg Dx)$. These two translations are logically equivalent, and so equally regiment sentence L4.

7.6 Paraphrasing Pronouns

When regimenting English sentences in $\mathcal{L}^{\text{FOL}}$, it is often helpful to paraphrase the sentence in English in a manner that exposes the logical features of that sentence. We have already provided a number of examples of this above. When paraphrasing, it is important that you do not accidentally make changes to the logical structure of the sentence since mistakes at this stage will end up resulting in the wrong regimentation.

Paraphrasing often requires departing from the superficial structure of the target sentence. For instance, consider the following examples and symbolization key:

N1. If MIT is in session, then it is full of activity.
N2. If some institute is in session, then it is full of activity.

Sx: x is in session.
Ax: x is full of activity.
m: MIT

Sentence N1 and sentence N2 have the same words in the consequent, but they cannot be regimented in the same way. This is because the 'it' in sentence N1 is not bound by a quantifier but rather indicates the same subject while avoiding repetition. By contrast, both occurrences of 'it' in sentence N2 are bound by the outermost quantifier. Whereas there is nothing to change about sentence N2, we may paraphrase sentence N1 as follows:

O1. If MIT is in session, then MIT is full of activity.

Compare the following regimentations of the sentences N1:

P1. $Sm \to Ax$.
P2. $Sm \to Am$.

Although it might be tempting to try to regiment the 'it' in the original sentence N1 by a variable, this would result in the open sentence P1 in place of the wfs P2. Since the original sentence says something entirely about MIT and nothing about some yet to be bound variable the way 'It is red' does, sentence P2 provides a better regimentation than P1 does.

7.7 Ambiguous Predicates

Suppose we want to regiment this sentence:

Q1. Jeni is a skilled surgeon.

Consider the following symbolization key:

Kx: x is skilled
Rx: x is a surgeon
j: Jeni

This yields the following:

R1. $Kj \wedge Rj$.

In English, this reads: Jeni is skilled and Jeni is a surgeon. Here one may object that being skilled and also being a surgeon is not the same thing as being a skilled surgeon. For instance, perhaps it is possible to be both skilled and a surgeon without being a skilled surgeon. Accordingly, we could have specified the following predicate instead:

Sx: x is a skilled surgeon
j: Jeni

We may then provide the following regimentation:

S1. Sa.

Considering sentence Q1 on its own, it may be unclear whether to go with regimentation R1 or S1. However, in the context of an argument, there may be good reason to go one way rather than the other. For instance, suppose that we want to regiment this argument:

T1. The hospital will only hire a skilled surgeon.
T2. Adina is skilled but not a surgeon.
T3. Billy is a surgeon, but not skilled.
T4. Jeni is a skilled surgeon.
T5. Therefore, the hospital will hire Jeni and not Billy or Adina.

Here we need to distinguish being a *skilled surgeon* from merely being a *surgeon*. Consider this symbolization key:

Hx: The hospital hires x.	a: Adina.
Kx: x is skilled.	b: Billy.
Rx: x is a surgeon.	j: Jeni.

Now the argument can be regimented this way:

U1. $\forall x(Hx \to (Kx \wedge Rx))$
U2. $Ka \wedge \neg Ka$
U3. $Rb \wedge \neg Kb$
U4. $Kj \wedge Rj$
U5. $Hj \wedge \neg(Hb \vee Ha)$

Although we have not yet provided a semantic nor theory of logical consequence for $\mathcal{L}^{\text{FOL}}$, it is worth considering whether the formal argument in $\mathcal{L}^{\text{FOL}}$ is valid. Whereas the argument in English might have seemed somewhat compelling— at least it is the kind of argument that it is common to hear— its formalization may be used to reveal its invalidity. The next chapter will provide resources to prove that this is so by providing a counterexample.

Compare the previous argument to the following:

V1. Carol is a skilled surgeon and a tennis player.
V2. Therefore, Carol is a skilled tennis player.

We may regiment this argument as follows:

W1.	$\underline{(Rc \wedge Kc) \wedge Tc}$	Tx:	x is a tennis player
W2.	$Tc \wedge Kc$	c:	Carol

This argument is valid in $\mathcal{L}^{\text{FOL}}$, but the original argument in English is not. Something has gone wrong with our regimentation. The problem is that there is a difference between being *skilled as a surgeon* and *skilled as a tennis player*. Regimenting this argument correctly requires two separate predicates, one for each type of skill. Thus we may add the following:

K_1x: x is skilled as a tennis player
K_2x: x is skilled as a surgeon

We may now regiment the argument in this way:

X1. $\underline{(K_2c \wedge Rc) \wedge Tc}$
X2. $K_1c \wedge Tc$

Like the English argument it regiments, this $\mathcal{L}^{\text{FOL}}$ argument is invalid.

Similar problems can arise with predicates like 'is good', 'is big', 'is tall', etc. Just as skilled surgeons and skilled tennis players have different skills, it's obvious that big dogs, big mice, and big problems are all big in different ways. Must we always distinguish between different ways of being skilled, good, big, or tall? Not necessarily. If you are translating an argument that is just about dogs, it is fine to use the predicate 'x is big'. However, if the argument is also about mice, it might be important to let use the predicate 'x is big for a dog' instead. In general, we try to introduce as few predicates as possible while nevertheless capturing the intended meaning of the sentence or argument in question, as well as the intuitive validity or invalidity of the arguments in English that we might consider.

7.8 Multiple Quantifiers

Consider the following sentences and symbolization key:

Dt:	x is a dog.	Y1.	Fifi is a dog.
Fxy:	x is a friend of y.	Y2.	Gerald is a dog owner.
Oxy:	x owns y.	Y3.	Someone is a dog owner.
f:	Fifi	Y4.	All of Gerald's friends are dog owners.
g:	Gerald	Y5.	Dog owners are friends with dog owners.

Sentence Y1 is easy: Df.

Sentence Y2 can be paraphrased as, 'There is a dog that Gerald owns' which can be translated as $\exists x(Dx \land Ogx)$. Also no so difficult.

Sentence Y3 can be paraphrased as, 'There is something such that it is a dog owner'. The subsentence 'it is a dog owner' is just like sentence Y2, except that it is about *it* rather than being about Gerald. We can then regiment the sentence Y3 as $\exists y \exists x(Dx \land Oyx)$, replacing 'is a dog owner' with our previous regimentation.

Sentence Y4 can be paraphrased as, 'Every friend of Gerald is a dog owner'. We can expand this to: 'Everything is such that, if it is a friend of Gerald, then it is a dog owner'. Translating part of this sentence, we get $\forall x(Fxg \to x$ is a dog owner$)$. Again, it is important to recognize that 'x is a dog owner' is structurally just like sentence Y2. Since we already have a quantifier binding x, we will need a different variable for the existential quantifier. Any other variable will do. Using z, sentence Y4 can be translated as $\forall x(Fxg \to \exists z(Dz \land Oxz))$.

Sentence Y5 can be paraphrased as 'For any x that is a dog owner, there is a dog owner who is a friend of x'. Partially translated, this becomes:

$$\forall x\big[x \text{ is a dog owner} \to \exists y(y \text{ is a dog owner} \land Fxy)\big].$$

Completing the translation, sentence Y5 becomes:

$$\forall x\big[\exists z(Dz \land Oxz) \to \exists y\big(\exists z(Dz \land Oyz) \land Fxy\big)\big].$$

Regimenting English sentences in $\mathcal{L}^{\text{FOL}}$ will take some practice, and often there will be more than one way to go. When you come up with a regimentation, it is often worth considering if there are any other regimentations that you could have provided. Often there are, and they are not always logically equivalent. Sometimes what this means is that the original claim is ambiguous. Other times, some regimentations will be more natural than others.

Consider the following sentences and symbolization key:

Z1. Imre likes everyone that Karl likes.
Z2. There is someone who likes everyone who likes everyone that he likes.

Z3. x likes y.
Z4. Imre.
Z5. Karl.

Sentence Z1 can be partially regimented as: $\forall x($Karl likes $x \to$ Imre likes $x)$ which becomes $\forall x(Lkx \to Lix)$. But we might have been able to skip to this final regimentation.

Sentence Z2 is much more complex. There is little hope of writing down the whole regimentation at once, so we will proceed in stages. An initial regimentation might look like:

$$\exists x \text{ everyone who likes everyone that } x \text{ likes is liked by } x$$

The part that remains in English is a universal sentence, so we translate further:

$$\exists x \forall y (y \text{ likes everyone that } x \text{ likes} \rightarrow x \text{ likes } y).$$

The antecedent of the conditional is structurally just like sentence Z1, with y and x in place of Imre and Karl. So sentence Z2 can be completely regimented as follows:

$$\exists x \forall y \big[\forall z (Lxz \rightarrow Lyz) \rightarrow Lxy \big]$$

When symbolizing sentences with multiple quantifiers, it is best to proceed by small steps. Paraphrase the English sentence so that the logical structure is readily regimented in $\mathcal{L}^{\text{FOL}}$. Then regiment piecemeal, replacing the daunting task of regimenting a long or dense sentence with many simple tasks of regimenting shorter sentences.

Chapter 8

A Semantics for First-Order Logic

In this chapter, we will provide a *semantics* for $\mathcal{L}^{\text{FOL}}$ in much the same way that we did for $\mathcal{L}^{\text{PL}}$. Recall that an interpretation of $\mathcal{L}^{\text{PL}}$ assigns exactly one truth-value to each sentence letter. For example, an interpretation $\mathcal{I}$ might have included these assignments:

$$\mathcal{I}(P) = 0$$
$$\mathcal{I}(Q) = 1$$
$$\mathcal{I}(R) = 0$$

Since all the sentential operators in $\mathcal{L}^{\text{PL}}$ are truth-functional, the interpretation $\mathcal{I}$ settles the truth-value of any $\mathcal{L}^{\text{PL}}$ sentence that can be constructed from P, Q, and R given the sentential connectives included in $\mathcal{L}^{\text{PL}}$. Because $\mathcal{L}^{\text{FOL}}$ is much more expressive than $\mathcal{L}^{\text{PL}}$, it requires a richer interpretation than merely assigning truth-values to sentence letters. In order to do so, we will define the *models* of $\mathcal{L}^{\text{FOL}}$ which you can think of as austere representations of what all the terms in $\mathcal{L}^{\text{FOL}}$ mean. Intuitively, a predicate for 'is a dog' might be assigned to the set of all the dogs, though in general we will not limit ourselves to only the intuitive (or *intended*) interpretations in providing a theory of logical consequence for $\mathcal{L}^{\text{FOL}}$. Similarly, constants for names such as 'Fred' will be assigned to individuals, all of which exist within the *domain* of a given model. Rather than also assigning variables to individuals once and for all, we will introduce variable assignment functions to fix the reference of variables so that we may change the reference of variables without changing the model. After all, the model is intended to fix the meanings of the non-logical terms in our language, and we don't want to have to change how we interpret our language in total in order to quantify over the domain of individuals provided by a given model with which we intend to interpret $\mathcal{L}^{\text{FOL}}$.

Whereas open sentences will require both a model and a variable assignment in order to determine their truth-values, the truth-values for the wfs of $\mathcal{L}^{\text{FOL}}$ are entirely determined by each model of the language. By providing semantic clauses for the logical terms in our language, we will define logical consequence in $\mathcal{L}^{\text{FOL}}$ by quantifying over all models of $\mathcal{L}^{\text{FOL}}$. Spelling out these details will be the focus of this chapter.

8.1 Predicate Extensions

Whereas $\mathcal{L}^{\text{PL}}$ was interpreted by assigning sentence letters to truth-values without recourse to any other structure, the same cannot be said for $\mathcal{L}^{\text{FOL}}$. Consider the following sentences:

A1. Casey is at the party and not dancing.
A2. Max loves Casey.
A3. Everyone at the party is dancing.

In order to interpret the sentences A1 – A3, we need to know what the names 'Casey' and 'Max' refer to, who is dancing, who is at the party, and who love who. Were we to merely assign truth-values to these sentences, the fact that the name 'Casey' occurs in both A1 and A2 would be lost. Additionally, there would be no way to detect the fact that A1 and A3 cannot both be true at once. Since it matters what objects there are and which names refer to which objects, each model in $\mathcal{L}^{\text{FOL}}$ will be based on a nonempty set of objects $\mathbb{D}$ which we will refer to as the DOMAIN of that model. You can think of the domain as including everything that there is for the purposes of the interpretation in question. For instance, we might take $\mathbb{D} = \{1, 2, 3\}$ to include just three natural numbers, or we might introduce a domain $\mathbb{D}' = \{Cam, Sara, Kaya, Mel\}$ which includes just four people. Domains can be infinite— e.g., the domain which includes all of the natural numbers, or all of the real numbers— but they cannot be empty. We will come back to contemplate this constraint shortly.

Since it gets cumbersome to write out names, we will use lower-case letters and sometimes numbers for the elements of a domain. It is easy to confuse these with constants, and indeed, it is common to take a constant 'c' to name the element c in the domain. This is permitted so long as we are clear that the letter 'c' is doing double duty.

Given a domain $\mathbb{D}$, we will interpret predicates by assigning them to sets which we will construct from the domain $\mathbb{D}$. For instance, suppose that we symbolize 'is taller than' with the 2-place predicate 'T^2'. To interpret 'T^2', we consider ORDERED PAIRS of elements from $\mathbb{D}$ which we will write $\langle \mathbf{x}, \mathbf{y} \rangle$ for convenience, using bold font to avoid confusion with the variables in $\mathcal{L}^{\text{FOL}}$. Accordingly, we may interpret 'T^2' by assigning it to a set of ordered pairs of elements from $\mathbb{D}$ where the first is taller than the second. We will refer to the set of elements to which a predicate is assigned by an interpretation of $\mathcal{L}^{\text{FOL}}$ as the EXTENSION of that predicate. For example $\langle a, b \rangle$ might belong to the extension of 'T^2' on a given interpretation.

You might be wondering: how do we know which elements in the domain are taller than which? This is analogous to asking: how do we know which sentence letters in $\mathcal{L}^{\text{PL}}$ are true? Instead of relying on some prior interpret of the sentence letters in $\mathcal{L}^{\text{PL}}$, each interpretation stipulates the truth-values of the sentence letters in $\mathcal{L}^{\text{PL}}$. In a similar manner, the interpretations of the predicates in $\mathcal{L}^{\text{FOL}}$ stipulates their extensions in a given model. It is by considering all models that we may define a sentence φ to be a logical consequence of a set of sentences Γ just in case φ is true in every model for which every sentence in Γ is true.

Whereas the extension of a 2-place predicate is a set of ordered pairs of elements from $\mathbb{D}$, how are we to interpret the 0-place and 1-place predicates, not to mention the n-place predicates of $\mathcal{L}^{\text{FOL}}$ for $n > 2$? Moreover, how are we going to do this all at once instead of having to provide separate instructions for the n-place predicates for each value of n.

We begin by drawing on the domain $\mathbb{D}$ to construct Cartesian products of the domain. For instance, if we want to interpret a 2-place predicate 'L^2', we must construct the Cartesian product $\mathbb{D}^2 = \mathbb{D} \times \mathbb{D}$ which includes all ordered pairs of the form $\langle \mathbf{x}, \mathbf{y} \rangle$ where both $\mathbf{x}$ and $\mathbf{y}$ are elements of $\mathbb{D}$. In set builder notation, we may define the following Cartesian product:

$$\mathbb{D}^2 \coloneqq \{\langle \mathbf{x}, \mathbf{y} \rangle : \mathbf{x}, \mathbf{y} \in \mathbb{D}\}.$$

The definition given above reads: $\mathbb{D}^2$ is the set which includes all and only the ordered pair $\langle \mathbf{x}, \mathbf{y} \rangle$ where both $\mathbf{x}$ and $\mathbf{y}$ are members of $\mathbb{D}$. Given this notation, we may require $\mathcal{L}^{\text{FOL}}$ interpretations to assign the 2-place predicates of $\mathcal{L}^{\text{FOL}}$ to subsets of $\mathbb{D}^2$. For instance, given $\mathbb{D} = \{a, b, c, d\}$, we might specify an interpretation where $\mathcal{I}(T^2) = \{\langle a, b \rangle, \langle b, c \rangle\} \subseteq \mathbb{D}^2$.

In order to interpret all n-place predicates of $\mathcal{L}^{\text{FOL}}$, we will generalise on the same pattern. Given any domain $\mathbb{D}$, we will begin by defining the n-ary Cartesian product:

$$\mathbb{D}^n = \{\langle \mathbf{x}_1, \ldots, \mathbf{x}_n \rangle : \mathbf{x}_1, \ldots, \mathbf{x}_n \in \mathbb{D}\}.$$

In place of ordered pairs, now we have ordered n-tuples. Accordingly, n-place predicates may be assigned to subsets of $\mathbb{D}^n$. For instance, a three place predicate will be assigned to a subset of $\mathbb{D}^3$ where its extension will include elements like $\langle a, b, c \rangle$. Following the same pattern takes care of all the extensions of all n-place predicates.

Officially, n-tuples are defined as sets of ordered pairs $\langle \mathbf{x}_1, \ldots, \mathbf{x}_n \rangle = \{\langle 1, \mathbf{x}_1 \rangle, \ldots, \langle n, \mathbf{x}_n \rangle\}$.[1] As a result, $\langle \mathbf{x}_1 \rangle = \{1, \mathbf{x}_1\}$, though for we will maintain the tuple notation $\langle \mathbf{x}_1 \rangle$ for consistency. It follows that $\mathbb{D}^1 = \{\langle \mathbf{x}_1 \rangle : \mathbf{x}_1 \in \mathbb{D}\}$ only consists of 1-tuples containing elements in $\mathbb{D}$. For instance, if $a \in \mathbb{D}$, it follows that $\langle a \rangle \in \mathbb{D}^1$. Accordingly, $\mathbb{D}^1 \neq \mathbb{D}$ where 1-place predicates will be assigned to sets of 1-tuples in $\mathbb{D}^1$. For example, if we take 'H^1' to symbolize the predicate 'is happy', then $\mathcal{I}(H^1) \subseteq \mathbb{D}^1$ might include such elements as $\langle a \rangle$.

Next we may consider $\mathbb{D}^0$. Setting $n = 0$ in the definition above, it follows that $\mathbb{D}^0 = \{\varnothing\}$ given that $\langle\rangle = \varnothing$. We will use $\mathbb{D}^0$ to interpret sentence letters so that every sentence letter is assigned to a subset of $\mathbb{D}^0$, i.e., to either $\varnothing$ or $\{\varnothing\}$. As it happens, these are the standard von Neumann definitions of the first two ordinal numbers $0 = \varnothing$ and $1 = \{\varnothing\}$, and so our present approach will maintain consistency with the conventions introduced for $\mathcal{L}^{\text{PL}}$. Following the pattern above, a $\mathcal{L}^{\text{FOL}}$ interpretation $\mathcal{I}$ may assign a 0-place predicate A^0 to a subset of $\{\varnothing\}$, and so either $\mathcal{I}(A^0) = \varnothing = 0$ or $\mathcal{I}(A^0) = \{\varnothing\} = 1$.

As already noted, $\mathcal{L}^{\text{FOL}}$ interpretations are parasitic on a domain, and so we cannot provide interpretations of $\mathcal{L}^{\text{FOL}}$ on their own. Rather, to interpret $\mathcal{L}^{\text{FOL}}$ we will specify an interpretation together with a domain where this pair will be referred to as a *model* of $\mathcal{L}^{\text{FOL}}$.

[1] Officially, the n-tuple $\langle \mathbf{x}_1, \ldots, \mathbf{x}_n \rangle$ is the surjective function f_n from the domain $\{m \in \mathbb{N} : 1 \leqslant m \leqslant n\}$ to the range $\{\mathbf{x}_i : 1 \leqslant i \leqslant n\}$ where $f_n(i) = \mathbf{x}_i$ for all $1 \leqslant i \leqslant n$. Thus $\langle\rangle = \varnothing$.

8.2 Models

We are now in a position to interpret the constants and n-place predicates of $\mathcal{L}^{\text{FOL}}$ over a given domain. In particular, $\mathcal{I}$ is an INTERPRETATION of $\mathcal{L}^{\text{FOL}}$ over $\mathbb{D}$ just in case it satisfies:

Constants: $\mathcal{I}(\alpha) \in \mathbb{D}$ for every constant α of $\mathcal{L}^{\text{FOL}}$.

Predicates: $\mathcal{I}(\mathcal{F}^n) \subseteq \mathbb{D}^n$ for every n-place predicate $\mathcal{F}^n$ of $\mathcal{L}^{\text{FOL}}$ where $n \geqslant 0$.

Whereas *Constants* requires the constants of $\mathcal{L}^{\text{FOL}}$ to be assigned to individuals in the domain $\mathbb{D}$, *Predicates* requires the n-place predicates to be assigned to subsets of $\mathbb{D}^n$. Since there would be no way to satisfy *Constants* if the domain were empty, we will require the domain to be nonempty $\mathbb{D} \neq \varnothing$. More specifically, a MODEL of $\mathcal{L}^{\text{FOL}}$ is any ordered pair $\mathcal{M} = \langle \mathbb{D}, I \rangle$ where $\mathbb{D}$ is a nonempty set and $\mathcal{I}$ is any interpretation of $\mathcal{L}^{\text{FOL}}$ over $\mathbb{D}$.

Consider the following regimentation for the sentences A1 – A3 given above:

B1. $Pc \wedge \neg Dc$.
B2. Lmc.
B3. $\forall x(Px \rightarrow Dx)$.

Px: x is at the party.
Dx: x is dancing.
Lxy: x loves y.
c: Casey.
m: Max.

Given the definition of a $\mathcal{L}^{\text{FOL}}$ model above, we may consider what it would look like to interpret these sentences. In particular, we must specify a domain $\mathbb{D}$ along with interpretations of the constants and predicates in $\mathcal{L}^{\text{FOL}}$. Since this would take a long time— there are infinitely many constants and predicates— we will restrict our ambitions to the more modest task of interpreting the constants and predicates with which we are concerned, officially referring to this as a PARTIAL MODEL of $\mathcal{L}^{\text{FOL}}$, often calling it a MODEL for short. Consider the following:

$$
\begin{aligned}
\mathbb{D} &= \{c, m\} \\
\mathcal{I}(c) &= c \\
\mathcal{I}(m) &= m \\
\mathcal{I}(P) &= \{\langle c \rangle\} \\
\mathcal{I}(D) &= \varnothing \\
\mathcal{I}(L) &= \{\langle m, c \rangle\}
\end{aligned}
$$

Here the domain consists of two elements c and m where the extension of the predicate P is the set $\{\langle c \rangle\}$, the extension of the predicate D is empty, and the extension of the predicate L is the set $\{\langle a, m \rangle\}$. It is important to observe that the constants c and m are doing double duty since they are each assigned to themselves. That is, we happened to pick a domain consisting of the constants which we are using to name themselves as elements of the domain. This is often convenient, but by no means necessary. For instance, we could have let $\mathbb{D} = \{1, 2\}$ where $\mathcal{I}(c) = 1$, $\mathcal{I}(m) = 2$, $\mathcal{I}(P) = \{\langle 1\rangle\}$, $\mathcal{I}(D) = \varnothing$, and $\mathcal{I}(L) = \{\langle 2, 1\rangle\}$. The only reason to prefer our first model as opposed to the second is that it is easy to keep track of what refers to what by taking the constants to name themselves.

Although we have interpreted the constants and predicates included in the symbolization key, we have not yet said anything about how variables are to be understood, nor have we provided a way to determine whether the sentences B1 – B3 are true or false. Although $\mathcal{L}^{\text{FOL}}$ models do not interpret variables directly, this does not mean that quantified sentences which include variables do not have truth-values. Rather, we will rely on the notion of a variable assignment which maps variables to elements of the domain, where this will resemble the interpretation of the constants, but may be varied independently of the model. It is by appealing to variable assignments that we may provide semantic clauses for the quantifiers.

8.3 Variable Assignments

Recall from before that every domain is required to be nonempty. Accordingly, there is guaranteed to be a way to interpret all of the constants included in $\mathcal{L}^{\text{FOL}}$ even if many of those constants end up referring to the same elements of the domain. If there is just one element d in the domain $\mathbb{D} = \{d\}$, every constant will be assigned to d, and so there is no latitude at all for how to interpret the constants in $\mathcal{L}^{\text{FOL}}$. By contrast, if $\mathbb{D}$ includes more than one element, suddenly there are many different ways for a $\mathcal{L}^{\text{FOL}}$ interpretation over $\mathbb{D}$ to assign constants to elements of the domain. For instance, given a domain $\mathbb{D}$ with just two elements in the domain, each interpretation of $\mathcal{L}^{\text{FOL}}$ will decide which element in the domain each of the infinitely many constants in $\mathcal{L}^{\text{FOL}}$ is assigned to. That is already a lot of decisions.

Something similar may be observed in the case of assigning variables to elements of the domain. Given a domain $\mathbb{D}$, a VARIABLE ASSIGNMENT $\hat{a}$ over $\mathbb{D}$ is any function from the variables in $\mathcal{L}^{\text{FOL}}$ to elements of $\mathbb{D}$. Accordingly, $\hat{a}(\alpha) \in \mathbb{D}$ for all variables α in $\mathcal{L}^{\text{FOL}}$. As in the case of assigning constants to elements in the domain, there are many different variable assignments so long as $\mathbb{D}$ includes more than one element. Nevertheless, we may quantify over the variable assignments defined for a given domain where it is by doing so that we will provide homophonic semantic clauses for the quantifiers included in $\mathcal{L}^{\text{FOL}}$.

Suppose that we have a variable assignment (v.a. for short) $\hat{a}$. In considering another v.a. $\hat{b}$, there is no guarantee that $\hat{a}$ and $\hat{b}$ will agree about the elements of the domain that they assign to the different variables in $\mathcal{L}^{\text{FOL}}$. Instead of considering any other v.a. $\hat{b}$ at all, it will often be convenient to consider *variations* of $\hat{a}$ which agree with $\hat{a}$ about the elements they assign to every variable with the only possible exception being some particular variable with which we happen to be concerned. More precisely, we will take $\hat{c}$ to be an α-VARIANT of $\hat{a}$ just in case $\hat{c}(\beta) = \hat{a}(\beta)$ for every variable $\beta \neq \alpha$. Accordingly, α-variants of $\hat{a}$ differ with $\hat{a}$ in at most the variable α, and may even agree about α. All we know is that α-variants agree with $\hat{a}$ about all variables with the only possible exception of α.

Given a domain $\mathbb{D} = \{1, 2, 3, 4, 5\}$, suppose that $\hat{a}(x) = 1$, $\hat{a}(y) = 2$, and $\hat{a}(z) = 3$. Letting $\hat{c}$ be a y-variant of $\hat{a}$, we know by definition that $\hat{c}(x) = \hat{a}(x) = 1$ and $\hat{c}(z) = \hat{a}(z) = 3$. What we don't know is the identity of $\hat{c}(y)$. Although it is possible that $\hat{c}(y) = \hat{a}(y) = 2$, all that we know is that $\hat{c}(y) \in \mathbb{D}$, and so there are exactly five possibilities given the size of our domain $\mathbb{D}$. It is by quantifying over variants of a v.a. that we will interpret the quantifiers in $\mathcal{L}^{\text{FOL}}$.

Given a model $\mathcal{M} = \langle \mathbb{D}, \mathcal{I} \rangle$ and a v.a. $\hat{a}$, it will be important to provide a uniform way to interpret singular terms. After all, a well-formed atomic formula of the form $\mathcal{F}^n\alpha_1, \ldots, \alpha_n$ may include both constants and variables, and we want to be able to treat these together in order to assign $\mathcal{F}^n\alpha_1, \ldots, \alpha_n$ a truth-value relative to both a modal and v.a. defined over the domain of that model. Thus we may define the VALUE (or REFERENCE) of a singular term:

$$v_{\mathcal{I}}^{\hat{a}}(\alpha) = \begin{cases} \mathcal{I}(\alpha) & \text{if } \alpha \text{ is a constant} \\ \hat{a}(\alpha) & \text{if } \alpha \text{ is a variable.} \end{cases}$$

If α happens to be a constant, then our function $v_{\mathcal{I}}^{\hat{a}}$ appeals to the interpretation $\mathcal{I}$ in order to specify the element of the domain to which it refers. If, however, α is a variable, then $v_{\mathcal{I}}^{\hat{a}}$ appeals to the v.a. $\hat{a}$ in order to specify an element of the domain. Given $\mathcal{I}$ and $\hat{a}$, we don't need to know whether α is a constant or a variable in order to specify the element in the domain to which it refers. This will turn out to be very important for providing truth-values for the atomic wffs of $\mathcal{L}^{\text{FOL}}$ since they may include both constants and variables.

8.4 Semantics

Having defined the models for $\mathcal{L}^{\text{FOL}}$ as well as the variable assignments for a given domain, we are now in a position to provide the semantic clauses by which we will assign truth and falsity to the sentences of $\mathcal{L}^{\text{FOL}}$. Here it is important to recall the difference between open sentences which include free variables and the sentences of $\mathcal{L}^{\text{FOL}}$ which do not. Whereas every model of $\mathcal{L}^{\text{FOL}}$ will determine the truth-value of the sentences of $\mathcal{L}^{\text{FOL}}$, the same does not hold for the open sentences of $\mathcal{L}^{\text{FOL}}$ which include free variables. Rather, such sentences must be interpreted at a model together with a variable assignment.

In order to interpret all wfs of $\mathcal{L}^{\text{FOL}}$ given only a model of the language, we will begin by interpreting all wff of $\mathcal{L}^{\text{FOL}}$ given both a model $\mathcal{M} = \langle \mathbb{D}, \mathcal{I} \rangle$ and a v.a. $\hat{a}$ defined over $\mathbb{D}$. Given these ingredients, we may provide a recursive definition of the VALUATION FUNCTION $\mathcal{V}_{\mathcal{I}}^{\hat{a}}$ which assigns ever wffs of $\mathcal{L}^{\text{FOL}}$ to a unique truth-value. Since wfs do not have free variables, it will turn out that if $\mathcal{V}_{\mathcal{I}}^{\hat{a}}(\varphi) = \mathcal{V}_{\mathcal{I}}^{\hat{c}}(\varphi)$ for any model $\mathcal{M}$ of $\mathcal{L}^{\text{FOL}}$, wfs φ, and v.a.s $\hat{a}$ and $\hat{c}$. Put otherwise, v.a.s only make a difference to the truth-values assigned to open sentences. Accordingly, we will take a wfs of $\mathcal{L}^{\text{FOL}}$ to be true in a model $\mathcal{M} = \langle \mathbb{D}, \mathcal{I} \rangle$ just in case it is true in $\mathcal{M}$ for every v.a. $\hat{a}$ defined over $\mathbb{D}$. It is in terms of this definition of truth in a model independent of any v.a. that we will go on to define logical consequence for $\mathcal{L}^{\text{FOL}}$.

With these ambitions in mind, given any model $\mathcal{M} = \langle \mathbb{D}, \mathcal{I} \rangle$ and v.a. $\hat{a}$ defined over $\mathbb{D}$, we may assign truth-values to all wffs of $\mathcal{L}^{\text{FOL}}$ by recursively defining the valuation function $\mathcal{V}_{\mathcal{I}}^{\hat{a}}$ from the domain of wffs for $\mathcal{L}^{\text{FOL}}$ to truth values $\{0, 1\}$ by way of the following semantics:

Valuation Function: For any wffs φ and ψ of $\mathcal{L}^{\text{FOL}}$, n-place predicate $\mathcal{F}^n$ of $\mathcal{L}^{\text{FOL}}$, and n singular terms $\alpha_1, \ldots, \alpha_n$ of $\mathcal{L}^{\text{FOL}}$:

$\mathcal{V}_{\mathcal{I}}^{\hat{a}}(\mathcal{F}^n\alpha_1, \ldots, \alpha_n) = 1$ *iff* $\langle \boldsymbol{v}_{\mathcal{I}}^{\hat{a}}(\alpha_1), \ldots, \boldsymbol{v}_{\mathcal{I}}^{\hat{a}}(\alpha_n) \rangle \in \mathcal{I}(\mathcal{F}^n)$.

$\mathcal{V}_{\mathcal{I}}^{\hat{a}}(\forall\alpha\varphi) = 1$ *iff* $\mathcal{V}_{\mathcal{I}}^{\hat{c}}(\varphi) = 1$ for every α-variant $\hat{c}$ of $\hat{a}$.

$\mathcal{V}_{\mathcal{I}}^{\hat{a}}(\exists\alpha\varphi) = 1$ *iff* $\mathcal{V}_{\mathcal{I}}^{\hat{c}}(\varphi) = 1$ for some α-variant $\hat{c}$ of $\hat{a}$.

$\mathcal{V}_{\mathcal{I}}^{\hat{a}}(\neg\varphi) = 1$ *iff* $\mathcal{V}_{\mathcal{I}}^{\hat{a}}(\varphi) \neq 1$.

$\mathcal{V}_{\mathcal{I}}^{\hat{a}}(\varphi \vee \psi) = 1$ *iff* $\mathcal{V}_{\mathcal{I}}^{\hat{a}}(\varphi) = 1$ or $\mathcal{V}_{\mathcal{I}}^{\hat{a}}(\psi) = 1$ (or both).

$\mathcal{V}_{\mathcal{I}}^{\hat{a}}(\varphi \wedge \psi) = 1$ *iff* $\mathcal{V}_{\mathcal{I}}^{\hat{a}}(\varphi) = 1$ and $\mathcal{V}_{\mathcal{I}}^{\hat{a}}(\psi) = 1$.

$\mathcal{V}_{\mathcal{I}}^{\hat{a}}(\varphi \to \psi) = 1$ *iff* $\mathcal{V}_{\mathcal{I}}^{\hat{a}}(\varphi) = 0$ or $\mathcal{V}_{\mathcal{I}}^{\hat{a}}(\psi) = 1$ (or both).

$\mathcal{V}_{\mathcal{I}}^{\hat{a}}(\varphi \leftrightarrow \psi) = 1$ *iff* $\mathcal{V}_{\mathcal{I}}^{\hat{a}}(\varphi) = \mathcal{V}_{\mathcal{I}}^{\hat{a}}(\psi)$.

The semantic clauses for the truth-functional operators $\neg, \wedge, \vee, \to, \leftrightarrow$ have been preserved from $\mathcal{L}^{\text{PL}}$ with the exception that a parameter has been added for a variable assignment. Nevertheless, variable assignments do not do any work in the semantic clauses for the truth-functional operators, and so we may focus attention on the first three clauses in which variables assignments make an essential contribution.

It will be convenient to refer to an interpretation and v.a. while leaving reference to the model over which they are defined implicit. For instance, we might consider some interpretation $\mathcal{I}$ and $\hat{a}$, where it is assumed that $\mathcal{I}$ belongs to a model $\mathcal{M} = \langle \mathbb{D}, \mathcal{I} \rangle$ of $\mathcal{L}^{\text{FOL}}$ where $\hat{a}$ is defined over the domain of that model $\mathbb{D}$. This convention will ease the following exposition.

Given any interpretation $\mathcal{I}$ and v.a. $\hat{a}$, an atomic wff such as G^3axy is true on $\mathcal{I}$ given $\hat{a}$ just in case $\langle \boldsymbol{v}_{\mathcal{I}}^{\hat{a}}(a), \boldsymbol{v}_{\mathcal{I}}^{\hat{a}}(x), \boldsymbol{v}_{\mathcal{I}}^{\hat{a}}(y) \rangle$ is a member of $\mathcal{I}(G^3)$. In this case $\boldsymbol{v}_{\mathcal{I}}^{\hat{a}}(a) = \mathcal{I}(a)$, $\boldsymbol{v}_{\mathcal{I}}^{\hat{a}}(x) = \hat{a}(x)$, and $\boldsymbol{v}_{\mathcal{I}}^{\hat{a}}(y) = \hat{a}(y)$, and so $\langle \boldsymbol{v}_{\mathcal{I}}^{\hat{a}}(a), \boldsymbol{v}_{\mathcal{I}}^{\hat{a}}(x), \boldsymbol{v}_{\mathcal{I}}^{\hat{a}}(y) \rangle = \langle \mathcal{I}(a), \hat{a}(x), \hat{a}(y) \rangle$. Since G^3 is a 3-place predicate, we know that $\mathcal{I}(G^3) \subseteq \mathbb{D}^3$ is a set of ordered triples. The question remains whether $\langle \mathcal{I}(a), \hat{a}(x), \hat{a}(y) \rangle \in \mathcal{I}(G^3)$. If so, G^3axy is true on $\mathcal{I}$ given $\hat{a}$, and false otherwise.

The wff $\forall xG^3axy$ is true on $\mathcal{I}$ given $\hat{a}$ just in case G^3axy is true on $\mathcal{I}$ given any x-variant $\hat{c}$ of $\hat{a}$. This requires $\langle \mathcal{I}(a), \hat{c}(x), \hat{c}(y) \rangle \in \mathcal{I}(G^3)$ for every x-variant $\hat{c}$ of $\hat{a}$. Since $x \neq y$, we know that $\hat{c}(y) = \hat{a}(y)$ for all x-variants $\hat{c}$ of $\hat{a}$. By contrast, $\hat{c}(x)$ is permitted to vary, where $\hat{c}(x)$ can range over all elements in $\mathbb{D}$. Thus by quantifying over all x-variants of $\hat{a}$, we are requiring $\langle \mathcal{I}(a), \mathbf{x}, \hat{a}(y) \rangle \in \mathcal{I}(G^3)$ for all $\mathbf{x}$ in the domain $\mathbb{D}$, leaving $\mathcal{I}(a)$ and $\hat{a}(y)$ unchanged. For instance, assuming $\mathbb{D} = \{1, 2, 3\}$ where $\mathcal{I}(a) = 1$ and $\hat{a}(y) = 3$, it follows that $\forall xG^3axy$ is true on $\mathcal{I}$ given $\hat{a}$ just in case $\langle 1, 1, 3 \rangle$, $\langle 1, 2, 3 \rangle$, and $\langle 1, 3, 3 \rangle$ are all members of $\mathcal{I}(G^3)$.

Suppose that some wff φ is true on an interpretation $\mathcal{I}$ given a v.a. $\hat{a}$. What can we conclude? Very little. Even though $\mathcal{V}_{\mathcal{I}}^{\hat{a}}(\varphi) = 1$, we don't know whether this wff will remain true on other variable assignments. Moreover, it is not clear what to make of truth on an interpretation *given a variable assignment*. What we want to know is whether the sentence is true on an interpretation independent of the variable assignment.

So long as a wff of $\mathcal{L}^{\text{FOL}}$ includes free variables, there is no way to assign that sentence a truth-value without appealing to a variable assignment. For instance, perhaps $\forall x G^3 axy$ is true on $\mathcal{I}$ given $\hat{a}$ since $\mathcal{I}(G^3) = \{\langle 1,1,3\rangle, \langle 1,2,3\rangle, \langle 1,3,3\rangle, \langle 2,3,1\rangle, \langle 1,1,1\rangle\}$. By contrast, the wfss of $\mathcal{L}^{\text{FOL}}$ which do not include free variables may have truth-values that are independent of any particular v.a. and determined entirely by the models of $\mathcal{L}^{\text{FOL}}$. For instance, consider the sentence $\exists y \forall x G^3 axy$. Because $\mathcal{V}_{\mathcal{I}}^{\hat{a}}(\forall x G^3 axy) = 1$ and $\hat{a}$ is a y-variant of itself, it follows that $\hat{e} = \hat{a}$ has a y-variant $\hat{a}$ where $\mathcal{V}_{\mathcal{I}}^{\hat{a}}(\forall x G^3 axy) = 1$, and so $\mathcal{V}_{\mathcal{I}}^{\hat{e}}(\exists y \forall x G^3 axy) = 1$ by the semantics for the existential quantifier. Whereas we chose $\hat{e} = \hat{a}$ for convenience, we could have taken $\hat{e}$ to be any v.a. whatsoever. This is because for any v.a. $\hat{e}$, it will have a y-variant $\hat{g}$ where $\hat{g}(y) = \hat{a}(y) = 3$, and so $\mathcal{V}_{\mathcal{I}}^{\hat{g}}(\forall x G^3 axy) = 1$ for the same reasons given above.

This case brings out the general point mentioned above: if φ is a wfs of $\mathcal{L}^{\text{FOL}}$ and so does not have any free variables, then $\mathcal{V}_{\mathcal{I}}^{\hat{a}}(\varphi) = \mathcal{V}_{\mathcal{I}}^{\hat{c}}(\varphi)$ for any v.a.s $\hat{a}$ and $\hat{c}$. This is perhaps easiest to see in the case where a sentence of $\mathcal{L}^{\text{FOL}}$ does not have any variables at all. For instance $\mathcal{V}_{\mathcal{I}}^{\hat{a}}(F^1 b) = 1$ just in case $\langle \mathcal{I}(b)\rangle \in \mathcal{I}(F^1)$. Since the v.a. $\hat{a}$ does not appear in $\langle \mathcal{I}(b)\rangle \in \mathcal{I}(F^1)$, we can be sure that $\mathcal{V}_{\mathcal{I}}^{\hat{a}}(F^1 b) = 1$ just in case $\mathcal{V}_{\mathcal{I}}^{\hat{c}}(F^1 b) = 1$ for every v.a. $\hat{c}$ whatsoever.

Although variable assignments play a critical role in assigning truth-values to the open sentences of $\mathcal{L}^{\text{FOL}}$, they play at most an auxiliary role in assigning truth-values to the sentences of $\mathcal{L}^{\text{FOL}}$. We may then define the truth-values of the wfss of $\mathcal{L}^{\text{FOL}}$ as follows:

Theory of Truth: For any wfs φ and model $\mathcal{M} = \langle \mathbb{D}, \mathcal{I}\rangle$ of $\mathcal{L}^{\text{FOL}}$:

$$\mathcal{V}_{\mathcal{I}}(\varphi) = 1 \text{ just in case } \mathcal{V}_{\mathcal{I}}^{\hat{a}}(\varphi) = 1 \text{ for every v.a. } \hat{a} \text{ defined over } \mathbb{D}.$$

Whereas the semantics for $\mathcal{L}^{\text{PL}}$ provided a theory of truth for the wfss of $\mathcal{L}^{\text{PL}}$ all by itself, the same cannot be said for the semantics for $\mathcal{L}^{\text{FOL}}$. Rather, the semantics for $\mathcal{L}^{\text{FOL}}$ defined truth relative to both a model of $\mathcal{L}^{\text{FOL}}$ and variable assignment defined over the domain of that model. Nevertheless, out primary concern is the same as it was in $\mathcal{L}^{\text{PL}}$: to interpret the wfs of $\mathcal{L}^{\text{FOL}}$ across a range of interpretations (models) of the language, independent of any other parameter. Instead of defining truth relative to both a model and v.a., the truth theory for $\mathcal{L}^{\text{FOL}}$ given above provides a way to abstract from the v.a.s upon which the truth-values of the wffs of $\mathcal{L}^{\text{FOL}}$ depend. This abstraction is licensed by the fact that the wfs of $\mathcal{L}^{\text{FOL}}$ do not contain free variables, and so the variable assignments have no work left to do.

Having specified what it is for a sentence of $\mathcal{L}^{\text{FOL}}$ to be true in a model, we may now move to define a range of semantic notions that are of interest as we did before where most prominent among them is the theory of logical consequence for $\mathcal{L}^{\text{FOL}}$.

8.5 Logical Consequence

Using the double turnstile symbol '$\vDash$' for logical consequence in $\mathcal{L}^{\text{FOL}}$ in the same way as we did for $\mathcal{L}^{\text{PL}}$, we may define logical consequence as follows:

> Logical Consequence: For any set of wfss $\Gamma \cup \{\varphi\}$ of $\mathcal{L}^{\text{FOL}}$:
>
> $\Gamma \vDash \varphi$ *iff* for any model $\mathcal{M} = \langle \mathbb{D}, \mathcal{I} \rangle$ of $\mathcal{L}^{\text{FOL}}$, if $\mathcal{V}_{\mathcal{I}}(\gamma) = 1$ for all $\gamma \in \Gamma$, then $\mathcal{V}_{\mathcal{I}}(\varphi) = 1$.

As before, $\vDash \varphi$ is shorthand for $\varnothing \vDash \varphi$, which requires that $\mathcal{V}_{\mathcal{I}}(\varphi) = 1$ in every $\mathcal{L}^{\text{FOL}}$ model. We may now restated all of the same semantic definitions that we introduced in Chapter 2 where it is understood that by quantifying overt all $\mathcal{I}$ and $\mathcal{J}$ we are implicitly quantifying over all models $\mathcal{M} = \langle \mathbb{D}, \mathcal{I} \rangle$ and $\mathcal{M}' = \langle \mathbb{D}', \mathcal{J} \rangle$ of $\mathcal{L}^{\text{FOL}}$ whatsoever.

> tautology: φ is a *tautology iff* $\mathcal{V}_{\mathcal{I}}(\varphi) = 1$ for all $\mathcal{I}$.
>
> contradiction: φ is a *contradiction iff* $\mathcal{V}_{\mathcal{I}}(\varphi) = 0$ for all $\mathcal{I}$.
>
> contingent: φ is *logically contingent iff* $\mathcal{V}_{\mathcal{I}}(\varphi) \neq \mathcal{V}_{\mathcal{J}}(\varphi)$ for some $\mathcal{I}$ and $\mathcal{J}$.
>
> entailment: φ *logically entails* ψ *iff* $\mathcal{V}_{\mathcal{I}}(\varphi) \leqslant \mathcal{V}_{\mathcal{I}}(\psi)$ for all $\mathcal{I}$.
>
> equivalence: φ is *logically equivalent* to ψ *iff* $\mathcal{V}_{\mathcal{I}}(\varphi) = \mathcal{V}_{\mathcal{I}}(\psi)$ for all $\mathcal{I}$.
>
> satisfiable: Γ is *satisfiable iff* there is some $\mathcal{I}$ where $\mathcal{V}_{\mathcal{I}}(\gamma) = 1$ for all $\gamma \in \Gamma$.

These semantic notions will play much the same role that they did in $\mathcal{L}^{\text{PL}}$. The only difference is that $\mathcal{L}^{\text{FOL}}$ is a much more expressively powerful language than $\mathcal{L}^{\text{PL}}$. Just as we constructed interpretations to make sentences of $\mathcal{L}^{\text{PL}}$ either true or false depending on our aims, we will do something similar for the sentences of $\mathcal{L}^{\text{FOL}}$ in order to construct countermodels.

8.6 Minimal Models

Suppose that we want to show that $\forall x Axx \rightarrow Bd$ is not a tautology. This requires showing that the sentence is not satisfied by every model. If we can provide a model in which the sentence is false, then we will have shown that the sentence is not a tautology.

What would such a model look like? In order for $\forall x Axx \rightarrow Bd$ to be false, the antecedent $\forall x Axx$ must be true, and the consequent Bd must be false. Whenever a sentence is true in a model, it is typically true in more than one model, and some models are more complicated than others. To avoid confusion, we will strive to keep things as simple as possible, constructing MINIMAL MODELS which do what we want without adding any unnecessary elements.

We being by writing '$\mathbb{D} = \{d$,' leaving off the bracket on the right to indicate that we may end up adding more elements to the domain, but only if we must. The reason we added d to the domain is that we know $\mathbb{D}$ is nonempty given that it is a domain. Note that we chose d instead of another element. This was not necessary but it is convenient since d appears in the sentence with which we are concerned. As brought out above, we will take constants to play double duty. Note that it does not matter whether Bd is true or false in the model since either way we will need to talk about what 'd' refers to, i.e., itself.

In order to make $\forall xAxx$ true, all members of the domain $\mathbb{D}$ must bear the relation A to themselves. So far we just have one element d in the domain, and so all that is required is that $\langle d, d\rangle \in \mathcal{I}(A)$. Accordingly, we may write '$\mathcal{I}(A) = \{\langle d, d\rangle$,' leaving off the bracket on the right as before since we might want to add more elements.

Next we want Bd to be false. By setting $\mathcal{I}(d) = d$, we may take d to refer to itself as we intended all along, assuming that $\langle d\rangle \notin \mathcal{I}(B)$. Accordingly, we may take '$\mathcal{I}(B) = \{$' to be the empty extension for the time being, leaving off the right bracket as before.

Given that we made some changes to the model in order to make Bd false, it is always prudent to check that we have not changed the truth-value of $\forall xAxx$. However, in this case, all we did was assign a constant to the only element in the domain and took B to have the empty extension. Accordingly, $\forall xAxx$ is true for the same reason as before.

For contrast, if we had added another element to the domain, then further changes would be required. For instance, if we added c to the domain so that $\mathbb{D} = \{d, c$, then we would have to add $\langle c, c\rangle$ to the extension of A so that $\mathcal{I}(A) = \{\langle d, d\rangle, \langle c, c\rangle$. Since we didn't change the domain or the extension of A in merely assigning d to itself and taking B to have the empty extension, we don't need to make these changes, maintaining minimality.

Having achieved what we wanted, we may finish our model by closing off all of the sets. Accordingly, we have constructed the following model $\mathcal{M} = \langle\mathbb{D}, \mathcal{I}\rangle$:

$$\begin{aligned}
\mathbb{D} &= \{d\}\\
\mathcal{I}(A) &= \{\langle d, d\rangle\}\\
\mathcal{I}(B) &= \varnothing\\
\mathcal{I}(d) &= d
\end{aligned}$$

Strictly speaking, a model specifies an extension for *every* predicate of $\mathcal{L}^{\text{FOL}}$ and a referent for *every* constant, and $\mathcal{M}$ does not do this. That would require specifying infinitely many extensions and infinitely many referents. We may do this all at once by saying that the extension of every other predicate is empty, and that every constant refers to d. Although we may add these details to complete our model, doing so is hardly necessary since these details don't effect the truth-value of the sentence with which we are concerned. Accordingly we will typically omit this extra step where this is similar to only specifying the truth-values of the sentence letters which occur in the $\mathcal{L}^{\text{PL}}$ wfss with which we are concerned and not worrying about the rest. Although we could always go on to say that all other sentence letters are false, or true, there is no need to do so given our limited aims.

Instead of providing a model of $\mathcal{L}^{\text{FOL}}$, $\mathcal{M}$ provides a partial model which fully specifies the truth-value of the sentence with which we are concerned, but does not fix the truth-values of other sentences. Moreover, the model above may be said to be minimal insofar as it does not add any superfluous details. Rather, we only made forced moves.

Perhaps you are wondering what the predicate A means in English? It doesn't really matter. For formal purposes, the existence of models like the one described above is enough to show that $\forall x Axx \rightarrow Bd$ is not a tautology. But we can offer an interpretation in English if we like.

Axy: x knows y's biggest secret.
Bx: x's powers derive from gamma radiation.
d: Miles Morales

This is one way we can interpret the model above. Add is true, because Miles does know Miles's biggest secret. Bd is false since Miles's powers came from a genetically enhanced spider, not from gamma radiation. But the partial model constructed above includes none of these details. All it says is that A is a predicate which is true of d, and that B is a predicate which does not apply to Miles. There are indefinitely many predicates in English that have this extension. For instance, Axy might instead mean 'x is the same size as y' or 'x and y live in the same city'. Similarly, Bx might translate as 'x is a billionaire' or 'x has an uncle'. In constructing a model and giving extensions for A and B, we do not need to specify what English predicates A and B should be used to translate. We are concerned with whether the wfs $\forall x Axx \rightarrow Bd$ comes out true or false, and all that matters for truth and falsity in $\mathcal{L}^{\text{FOL}}$ is the information included in the model that we construct.

We can just as easily show that $\forall x Axx \rightarrow Bd$ is not a contradiction. We need only specify a model in which $\forall x Axx \rightarrow Bd$ is true, i.e., a model in which either $\forall x Axx$ is false or Bd is true. Here is a minimal partial model $\mathcal{M}' = \langle \mathbb{D}, \mathcal{J} \rangle$ with the same domain as before:

$$\begin{aligned}
\mathbb{D} &= \{d\}\\
\mathcal{J}(A) &= \{\langle d, d\rangle\}\\
\mathcal{J}(B) &= \{\langle d\rangle\}\\
\mathcal{J}(d) &= d
\end{aligned}$$

On this model, $\forall x Axx \rightarrow Bd$ is true, since it is a conditional with a true consequent. Alternatively, since conditionals with false antecedents are true, we could have taken the extensions of both A and B to be empty, where this is even simpler. Either way, $\forall x Axx \rightarrow Bd$ is not a contradiction, and so together with what was shown before, $\forall x Axx \rightarrow Bd$ is contingent. As before, showing that a sentence is contingent will require two models: one in which the sentence is true and another in which the sentence is false.

You might be wondering what happened to the variable assignments from before. In order to prove that a quantified sentence in $\mathcal{L}^{\text{FOL}}$ is true or false in a given model, shouldn't we have to say something about variable assignments? The answer is 'Yes', but sometimes we can say very little. For instance, letting $\hat{a}$ be any v.a., we may observe that Bd is true in $\mathcal{M}'$ given $\hat{a}$, and so $\forall x Axx \to Bd$ is true in $\mathcal{M}'$ given $\hat{a}$ by the semantics for the material conditional. Officially what this looks like is that $\mathcal{V}^{\hat{a}}_{\mathcal{J}}(\forall x Axx \to Bd) = 1$. Since $\forall x Axx \to Bd$ has no free variables and $\hat{a}$ was arbitrary, we may conclude that $\mathcal{V}_{\mathcal{J}}(\forall x Axx \to Bd) = 1$. Although the variable assignment $\hat{a}$ comes along for the ride, it does not do any substantive work.

In order to show that $\forall x Axx \to Bd$ is false, the variable assignments are no longer free wheels. Letting $\hat{a}$ be an arbitrary v.a. defined over $\mathbb{D}$, we may show that $\forall x Axx$ is true in our first model $\mathcal{M}$ by choosing an arbitrary x-variant $\hat{c}$ of $\hat{a}$. Since $\mathbb{D}$ only has one element, there is only one v.a. that can be defined over $\mathbb{D}$ which assigns every variable to d, and so $\hat{c}(x) = \hat{a}(x) = d$. We may then observe that $\langle \hat{c}(x), \hat{c}(x) \rangle \in \mathcal{I}(A)$ where $\hat{c}(x) = v^{\hat{c}}_{\mathcal{I}}(x)$, and so $\mathcal{V}^{\hat{c}}_{\mathcal{I}}(Axx) = 1$ by the semantics. Given that $\hat{c}$ was arbitrary, $\mathcal{V}^{\hat{a}}_{\mathcal{I}}(\forall x Axx) = 1$ follows by the semantics. Since $\langle \mathcal{I}(d) \rangle \notin \mathcal{I}(B)$, we may observe that $\mathcal{V}^{\hat{a}}_{\mathcal{I}}(Bd) = 0$, and so $\mathcal{V}^{\hat{a}}_{\mathcal{I}}(\forall x Axx \to Bd) = 0$. Generalizing on $\hat{a}$, it follows that that $\mathcal{V}_{\mathcal{I}}(\forall x Axx \to Bd) = 0$ as desired.

Although officially we need to go through all of these mechanics to show that a quantified sentence is true or false, it is often easy to see what is required to construct a minimal partial model. For instance, to show that $\forall x Axx$ is true, we need everything in the domain to be A-related to itself, i.e., $\langle \mathbf{x}, \mathbf{x} \rangle$ must be in the extension of A for all $\mathbf{x} \in \mathbb{D}$. In other cases, especially when multiple quantifiers are involved, a lot more care may be required to keep things straight and to produce appropriate models.

Suppose that we want to show that $\forall x Sx$ and $\exists x Sx$ are not logically equivalent. We need to construct a model in which the two sentences have different truth-values. We start by specifying a nonempty domain $\mathbb{D} = \{1$. Since the sentences with which we are concerned include the same predicate, there is no chance that S may have different extensions. Moreover, given a domain with just one member, there is no difference between something being S and everything being S. Thus we must add another element to the domain $\mathbb{D} = \{1, 2$. In order to make $\exists x Sx$ true without making $\forall x Sx$ true, we may take $\mathcal{I}(S) = \{\langle a \rangle$. Letting $\hat{a}$ be an arbitrary v.a. defined over $\mathbb{D}$, we may take $\hat{c}$ to be an x-variant of $\hat{a}$ where $\hat{c}(x) = 1$, it follows that $\langle \hat{c}(x) \rangle \in \mathcal{I}(S)$, and so by definition $\langle v^{\hat{c}}_{\mathcal{I}}(x) \rangle \in \mathcal{I}(S)$. Thus $\mathcal{V}^{\hat{c}}_{\mathcal{I}}(Sx) = 1$, and so $\mathcal{V}^{\hat{a}}_{\mathcal{I}}(\exists x Sx) = 1$ since $\hat{c}$ is an x-variant of $\hat{a}$. By generalizing on $\hat{a}$, it follows that $\mathcal{V}_{\mathcal{I}}(\exists x Sx) = 1$ given that $\exists x Sx$ is a wfs of $\mathcal{L}^{\text{FOL}}$ on account of having no free variables.

What about $\forall x Sx$? In order to show that $\mathcal{V}_{\mathcal{I}}(\forall x Sx) = 0$, we must find some v.a. $\hat{a}$ where $\mathcal{V}^{\hat{a}}_{\mathcal{I}}(\forall x Sx) = 0$. Since $\langle 2 \rangle \notin \mathcal{I}(S)$, we may let $\hat{a}$ be a v.a. defined over $\mathbb{D}$ where $\hat{a}(x) = 2$. It follows that $\langle \hat{a}(x) \rangle \notin \mathcal{I}(S)$, and so $\langle \mathcal{V}^{\hat{a}}_{\mathcal{I}}(x) \rangle \notin \mathcal{I}(S)$. Thus $\mathcal{V}^{\hat{a}}_{\mathcal{I}}(Sx) \neq 1$, and so $\mathcal{V}^{\hat{a}}_{\mathcal{I}}(\forall x Sx) \neq 1$ since $\hat{a}$ is an x-variant of itself and so not every x-variant $\hat{c}$ of $\hat{a}$ is such that $\mathcal{V}^{\hat{c}}_{\mathcal{I}}(Sx) = 1$. Since $\forall x Sx$ is a wfs of $\mathcal{L}^{\text{FOL}}$, we may conclude that $\mathcal{V}_{\mathcal{I}}(\forall x Sx) = 0$ as desired.

Having produced a partial model where $\forall x Sx$ and $\exists x Sx$ have different truth-values, we may close off the sets we defined above. The result may be stated as follows:

$$\begin{aligned} \mathbb{D} &= \{a, b\} \\ \mathcal{I}(S) &= \{\langle a \rangle\} \end{aligned}$$

This partial model shows that the two sentences are *not* logically equivalent since $\exists x Sx$ is true on this model and $\forall x Sx$ is false. Whereas only one model is required to show that two sentences of $\mathcal{L}^{\text{FOL}}$ are not logically equivalent, to show that two sentences are logically equivalent we will need to quantify over all $\mathcal{L}^{\text{FOL}}$ models. We will attend to this in the following section. However, before doing so, let's wrap up one loose end from before.

Back in §7.7, we said that this argument would be invalid in $\mathcal{L}^{\text{FOL}}$:

C1. $\underline{(K_2c \wedge Rc) \wedge Tc}$
C2. $K_1c \wedge Tc$

Now we can prove that this is so. To show that this argument is invalid, we need to show that there is some model in which the premise is true and the conclusion is false. We can construct such a model as follows:

$$\begin{aligned} \mathbb{D} &= \{c\} \\ \mathcal{I}(T) &= \{\langle c \rangle\} \\ \mathcal{I}(K_1) &= \varnothing \\ \mathcal{I}(K_2) &= \{\langle c \rangle\} \\ \mathcal{I}(R) &= \{\langle c \rangle\} \\ \mathcal{I}(c) &= c \end{aligned}$$

This time, no variable assignments are required in any substantive capacity. All we need to do is observe that $\langle c \rangle$ is a member of the extensions $\mathcal{I}(K_2)$, $\mathcal{I}(R)$, and $\mathcal{I}(T)$. Since $\mathcal{I}(c) = c$ where 'c' is a constant, it follows that $\langle \mathcal{V}_{\mathcal{I}}^{\hat{a}}(c) \rangle = \langle \mathcal{I}(c) \rangle = \langle c \rangle$ is a member of the same extensions where $\hat{a}$ is any v.a. over $\mathbb{D}$. Thus $\mathcal{V}_{\mathcal{I}}^{\hat{a}}(K_2c) = \mathcal{V}_{\mathcal{I}}^{\hat{a}}(Rc) = \mathcal{V}_{\mathcal{I}}^{\hat{a}}(Tc) = 1$, and so $\mathcal{V}_{\mathcal{I}}^{\hat{a}}((K_2c \wedge Rc) \wedge Tc) = 1$ by the semantics for conjunction. Since there are no free variables in this sentence and $\hat{a}$ was arbitrary, we may conclude that $\mathcal{V}_{\mathcal{I}}((K_2c \wedge Rc) \wedge Tc) = 1$.

Next we may let $\hat{e}$ be a particular v.a. over $\mathbb{D}$. Given that $\langle c \rangle \notin \mathcal{I}(K_2)$, it follows from the definitions that $\mathcal{V}_{\mathcal{I}}^{\hat{e}}(c) \notin \mathcal{I}(K_2)$, and so $\mathcal{V}_{\mathcal{I}}^{\hat{e}}(K_2c) \neq 1$. By the semantics for conjunction $\mathcal{V}_{\mathcal{I}}^{\hat{e}}(K_2c \wedge Tc) \neq 1$, and so $\mathcal{V}_{\mathcal{I}}(K_2c \wedge Tc) \neq 1$ given that $K_2c \wedge Tc$ is a wfs of $\mathcal{L}^{\text{FOL}}$. Since the partial model above makes the premise true and the conclusion false, the argument is invalid.

Suppose that we want to show that a set of sentences is satisfiable. For instance, consider the set $\Gamma = \{(K_2c \wedge Rc) \wedge Tc,\ K_1c \wedge Tc\}$ which includes the premise and conclusion from the argument above. We may tweak our model from before in order to satisfy this set, thereby proving that it is consistent. In particular, we may take $\mathcal{J}$ to be just like $\mathcal{I}$ given above except that $\mathcal{J}(K_1) = \{\langle c \rangle\}$. It is easy to show by similar reasoning that $\mathcal{V}_{\mathcal{J}}(K_2c \wedge Tc) = 1$, and so Γ is satisfiable. Thus we may conclude that Γ is consistent. Although officially *models* of $\mathcal{L}^{\text{FOL}}$ satisfy sets of sentences and interpretations like $\mathcal{J}$ do not, it is often convenient to refer directly to the interpretation when the model is clear from context. Thus it is common to use 'true in a model $\mathcal{M}$' and 'true in an interpretation $\mathcal{I}$' interchangeably.

8.7 Reasoning About all Models

We can show that a wfs of $\mathcal{L}^{\text{FOL}}$ is not a tautology by providing a carefully specified model in which the wfs in question is false. Similarly, to show that a wfs is not a contradiction, we only need to produce one model in which the wfs is true. In order to show that a wfs is contingent, we need to produce two models where the wfs is true in one model and false in the other. By contrast, only one model is required to show that two wfss are not equivalent on account of having different truth-values in that model. Similarly, only one model is required to show that a set of wfss is satisfiable.

For the same reasons observed in Chapter 2, we cannot appeal to any one or two models in order to show that a wfs of $\mathcal{L}^{\text{FOL}}$ is a tautology since this requires showing that the wfs is true in every model. Whereas producing one or two models is constructive in nature, establishing that a wfs is true in all models takes a general form where no individual constructions will suffice. For the same reason, we cannot show that a wfs is a contradiction by constructing a particular model, since what we need to show is that the wfs is false in every model. In both cases, we must reason about all models of $\mathcal{L}^{\text{FOL}}$ where this will require a distinct set of strategies to those brought out above.

In addition to showing that a wfs is a tautology or contradiction, reasoning about all models is also required to show that an argument is valid, that two wfss are logically equivalent, or that a set of wfss is inconsistent. To summarizes, consider the following table:

	YES	NO
φ is a tautology	show that φ must be true in any model	*construct a model* in which φ is false
φ is a contradiction	show that φ must be false in any model	*construct a model* in which φ is true
φ is contingent	*construct two models*, one where φ is true and one where φ is false	show that φ is a tautology or that φ is a contradiction
φ and ψ are equivalent	show that φ and ψ have the same truth-value in any model	*construct a model* in which φ and ψ have different truth-values
Γ is consistent	*construct a model* in which all the wfss in Γ are true	show that there is no model that satisfies Γ
$\varphi_1, \varphi_2, \ldots \therefore \psi$ is valid	show that any model that satisfies $\{\varphi_1, \varphi_2, \ldots\}$ also satisfies ψ	*construct a model* that satisfies $\{\varphi_1, \varphi_2, \ldots\}$ but does not satisfy ψ

Consider, for example, the wfs $Raa \leftrightarrow Raa$. In order to show that this wfs of $\mathcal{L}^{\text{FOL}}$ is a tautology, we need to show something about all models. Since there is no hope of doing so one at a time, one way to proceed is by *reductio*. Consider the following proof:

Proof: Assume that there is a $\mathcal{L}^{\text{FOL}}$ model $\mathcal{M} = \langle \mathbb{D}, \mathcal{I} \rangle$ where $\mathcal{V}_{\mathcal{I}}(Raa \leftrightarrow Raa) \neq 1$. It follows that $\mathcal{V}_{\mathcal{I}}^{\hat{a}}(Raa \leftrightarrow Raa) \neq 1$ for some v.a. $\hat{a}$ defined over $\mathbb{D}$. Accordingly, $\mathcal{V}_{\mathcal{I}}^{\hat{a}}(Raa) \neq \mathcal{V}_{\mathcal{I}}^{\hat{a}}(Raa)$, and so $\langle \mathcal{V}_{\mathcal{I}}^{\hat{a}}(a), \mathcal{V}_{\mathcal{I}}^{\hat{a}}(a) \rangle \notin \mathcal{I}(R)$ and $\langle \mathcal{V}_{\mathcal{I}}^{\hat{a}}(a), \mathcal{V}_{\mathcal{I}}^{\hat{a}}(a) \rangle \in \mathcal{I}(R)$. But this is a contradiction, and so $\mathcal{V}_{\mathcal{I}}(Raa \leftrightarrow Raa) = 1$ for every $\mathcal{L}^{\text{FOL}}$ model $\mathcal{M}$. □

The variable assignment $\hat{a}$ did no substantive work above. By contrast, consider the tautology $\forall x(Rxx \leftrightarrow Rxx)$. It might be tempting to reason in this way: $Rxx \leftrightarrow Rxx$ is true in every model, so $\forall x(Rxx \leftrightarrow Rxx)$ must also be true. The problem is that $Rxx \leftrightarrow Rxx$ is *not* true in every model. Since x is a variable rather than a constant, $Rxx \leftrightarrow Rxx$ is not a wfs, and so it is neither true nor false in any model. Rather, $Rxx \leftrightarrow Rxx$ is an open sentence, and so only has a truth-value in a model *given a variable assignment*. Consider the following proof:

Proof: Assume there is a $\mathcal{L}^{\text{FOL}}$ model $\mathcal{M} = \langle \mathbb{D}, \mathcal{I} \rangle$ where $\mathcal{V}_{\mathcal{I}}(\forall x(Rxx \leftrightarrow Rxx)) \neq 1$. It follows that $\mathcal{V}_{\mathcal{I}}^{\hat{a}}(\forall x(Rxx \leftrightarrow Rxx)) \neq 1$ for some v.a. $\hat{a}$ defined over $\mathbb{D}$. As a result there is some x-variant $\hat{c}$ of $\hat{a}$ where $\mathcal{V}_{\mathcal{I}}^{\hat{c}}(Rxx \leftrightarrow Rxx) \neq 1$. Accordingly, $\mathcal{V}_{\mathcal{I}}^{\hat{c}}(Rxx) \neq \mathcal{V}_{\mathcal{I}}^{\hat{c}}(Rxx)$, and so $\langle \mathcal{V}_{\mathcal{I}}^{\hat{c}}(x), \mathcal{V}_{\mathcal{I}}^{\hat{c}}(x) \rangle \notin \mathcal{I}(R)$ and $\langle \mathcal{V}_{\mathcal{I}}^{\hat{c}}(x), \mathcal{V}_{\mathcal{I}}^{\hat{c}}(x) \rangle \in \mathcal{I}(R)$. But this is a contradiction, so $\mathcal{V}_{\mathcal{I}}(\forall x(Rxx \leftrightarrow Rxx)) = 1$ for every $\mathcal{L}^{\text{FOL}}$ model $\mathcal{M}$. □

This proof is very similar. If you feel like you would struggle to come up with these proofs, note that each step follows immediately from the definitions. All you need to do for simple proofs like these is to assume that there is a model which makes the sentence in question false and use the definitions to derive a contradiction. It can take some time to become familiar with these definitions, but no better way to practice them than by writing proofs.

Once multiple quantifiers are involved, things get a lot trickier. For instance, suppose we want to show that $\forall x \forall y(Fxy \to \neg Fyx) \vDash \forall x \neg Fxx$. The proof proceeds in a similar fashion, assuming that there is a model which makes the premise true and the conclusion false. However, instead of relying solely on the definitions to lead us to a contradiction, a little bit of strategy will be required. Consider the following proof:

Proof: Assume for contradiction that there is a $\mathcal{L}^{\text{FOL}}$ model $\mathcal{M} = \langle \mathbb{D}, \mathcal{I} \rangle$ where $\mathcal{V}_{\mathcal{I}}(\forall x \forall y(Fxy \to \neg Fyx)) = 1$ and $\mathcal{V}_{\mathcal{I}}(\forall x \neg Fxx) \neq 1$. It follows from the former claim that $\mathcal{V}_{\mathcal{I}}^{\hat{a}}(\forall x \forall y(Fxy \to \neg Fyx)) = 1$ for all v.a. $\hat{a}$ defined over $\mathbb{D}$, where $\mathcal{V}_{\mathcal{I}}^{\hat{c}}(\forall x \neg Fxx) \neq 1$ for some particular v.a. $\hat{c}$ defined over $\mathbb{D}$ follows from the latter claim. Thus $\mathcal{V}_{\mathcal{I}}^{\hat{c}}(\forall x \forall y(Fxy \to \neg Fyx)) \neq 1$ follows from the former.

By the semantics, $\mathcal{V}_{\mathcal{I}}^{\hat{e}}(\neg Fxx) \neq 1$ for some x-variant of $\hat{e}$ of $\hat{c}$, and so $\mathcal{V}_{\mathcal{I}}^{\hat{e}}(Fxx) = 1$ by the semantics. Thus $\langle \mathcal{V}_{\mathcal{I}}^{\hat{e}}(x), \mathcal{V}_{\mathcal{I}}^{\hat{e}}(x) \rangle \in \mathcal{I}(F)$, and so $\langle \hat{e}(x), \hat{e}(x) \rangle \in \mathcal{I}(F)$.

Since $\hat{e}$ is an x-variant of $\hat{c}$, we know from above that $\mathcal{V}_{\mathcal{I}}^{\hat{e}}(\forall y(Fxy \to \neg Fyx)) = 1$. Let $\hat{g}$ be the y-variant of $\hat{e}$ where $\hat{g}(y) = \hat{e}(x)$. Thus $\mathcal{V}_{\mathcal{I}}^{\hat{g}}(Fxy \to \neg Fyx) = 1$. By the semantics, $\mathcal{V}_{\mathcal{I}}^{\hat{g}}(Fxy) \neq 1$ or $\mathcal{V}_{\mathcal{I}}^{\hat{g}}(\neg Fyx) = 1$, and so $\mathcal{V}_{\mathcal{I}}^{\hat{g}}(Fxy) \neq 1$ or

$\mathcal{V}_{\mathcal{I}}^{\hat{g}}(Fyx) \neq 1$. Thus $\langle \mathcal{V}_{\mathcal{I}}^{\hat{g}}(x), \mathcal{V}_{\mathcal{I}}^{\hat{g}}(y)\rangle \notin \mathcal{I}(F)$ or $\langle \mathcal{V}_{\mathcal{I}}^{\hat{g}}(y), \mathcal{V}_{\mathcal{I}}^{\hat{g}}(x)\rangle \notin \mathcal{I}(F)$, and since x and y are variables, $\langle \hat{g}(x), \hat{g}(y)\rangle \notin \mathcal{I}(F)$ or $\langle \hat{g}(y), \hat{g}(x)\rangle \notin \mathcal{I}(F)$.

Since $\hat{g}$ is a y-variant of $\hat{e}$ and $x \neq y$, it follows that $\hat{g}(x) = \hat{e}(x)$. Moreover, $\hat{g}(y) = \hat{e}(x)$ by stipulation, and so $\langle \hat{e}(x), \hat{e}(x)\rangle \notin \mathcal{I}(F)$, contradicting the above. Thus there is no model $\mathcal{M} = \langle \mathbb{D}, \mathcal{I}\rangle$ where $\mathcal{V}_{\mathcal{I}}(\forall x\forall y(Fxy \to \neg Fyx)) = 1$ and $\mathcal{V}_{\mathcal{I}}(\forall x\neg Fxx) \neq 1$. It follows that every model to make $\forall x\forall y(Fxy \to \neg Fyx)$ true also makes $\forall x\neg Fxx$ true, and so $\forall x\forall y(Fxy \to \neg Fyx) \vDash \forall x\neg Fxx$. $\square$

This proof was a lot more complicated, and required some careful moves. In particular, we used $\mathcal{V}_{\mathcal{I}}(\forall x\neg Fxx) \neq 1$ to conclude that $\mathcal{V}_{\mathcal{I}}^{\hat{c}}(\forall x\neg Fxx) \neq 1$ for a particular v.a. $\hat{c}$, and used $\mathcal{V}_{\mathcal{I}}(\forall x\forall y(Fxy \to \neg Fyx)) = 1$ for every v.a. $\hat{a}$, and so for $\hat{c}$ in particular. We then unpacked the former claim since it produced an x-variant $\hat{e}$ where $\langle \hat{e}(x), \hat{e}(x)\rangle \in \mathcal{I}(F)$. The remainder of the proof drew on the general claim $\mathcal{V}_{\mathcal{I}}^{\hat{a}}(\forall x\forall y(Fxy \to \neg Fyx)) = 1$ to show that $\langle \hat{e}(x), \hat{e}(x)\rangle \notin \mathcal{I}(F)$. It was important to observe that $\hat{e}$ was a x-variant of $\hat{c}$, and to have carefully chosen the y-variant $\hat{g}$ of $\hat{e}$ so that $\hat{g}(y) = \hat{e}(x)$. The rest follows by the definitions.

8.8 Constants and Quantifiers

In writing semantic proofs, it is best to unpack existential claims before universal claims.[2] This as analogous to the idea that it is best to unpack conjunctions before disjunctions in writing PL derivations. Just as a negated conjunction has a similar character to a disjunction, and a negated disjunction has a similar character to a conjunction, something similar holds for the quantifiers. In particular, a negated (or false) universal claim has a similar character to a existential claim, and a negated (or false) existential claim has a similar character to a universal claim. We may then restate our previous recommendation: it is best to unpack claims with an *existential character* before unpacking claims with a *universal character*.

Consider the wfss of $\mathcal{L}^{\text{FOL}}$ $\forall x\neg Fxx$, $\neg\exists y\neg Gy$, $\exists z\neg Kz$, and $\neg Hbc$. Which of these has an existential character, and which has a universal character? Although $\forall x\neg Fxx$ includes a negation sign, it is making a general claim— i.e., that nothing is F-related to itself— and so has a universal character. Next consider $\neg\exists y\neg Gy$ which, says that nothing is not G, and so everything is G. Again this has a universal character. Although $\exists z\neg Kz$ includes a negation sign, what we are saying is that something is not K, where this has an existential character. Lastly, what are we to make of $\neg Hbc$? Although this wfs does not include any quantifiers at all, this wfs has in some ways the most existential character of all. Not only does $\neg Hbc$ say that *something* is H-related to *something*, it names the things that are H-related, though we don't know if 'a' and 'c' name the same thing or not.

In order to bring out the existential character that constants have, consider the logical consequence $\neg Hbc \vDash \exists x\exists y\neg Hxy$. Whereas above we used a *reductio* style proof, this time we may write a direct proof without too much trouble:

[2] We will see an analogue of this same idea show up in the proof system that we will introduce for $\mathcal{L}^{\text{FOL}}$.

Proof: Let $\mathcal{M} = \langle \mathbb{D}, \mathcal{I} \rangle$ be a model of $\mathcal{L}^{\text{FOL}}$ where $\mathcal{V}_{\mathcal{I}}(\neg Hbc) = 1$. It follows that $\mathcal{V}_{\mathcal{I}}^{\hat{a}}(\neg Hbc) = 1$ for every v.a. $\hat{a}$ defined over $\mathbb{D}$, and so $\mathcal{V}_{\mathcal{I}}^{\hat{c}}(\neg Hbc) = 1$ for some $\hat{c}$ in particular. Thus $\mathcal{V}_{\mathcal{I}}^{\hat{c}}(Hbc) \neq 1$, and so $\langle \boldsymbol{v}_{\mathcal{I}}^{\hat{c}}(b), \boldsymbol{v}_{\mathcal{I}}^{\hat{c}}(c) \rangle \notin \mathcal{I}(H)$.

Let $\hat{e}$ be an arbitrary v.a. defined over $\mathbb{D}$ where $\hat{g}$ is the x-variant of $\hat{e}$ where $\hat{g}(x) = \boldsymbol{v}_{\mathcal{I}}^{\hat{c}}(b)$. We may then let $\hat{h}$ be the y-variant of $\hat{g}$ where $\hat{h}(y) = \boldsymbol{v}_{\mathcal{I}}^{\hat{c}}(c)$. Since $x \neq y$, we know that $\hat{h}(x) = \hat{g}(x)$, and so $\langle \hat{h}(x), \hat{h}(y) \rangle \notin \mathcal{I}(H)$ given the above. It follows that $\langle \boldsymbol{v}_{\mathcal{I}}^{\hat{h}}(x), \boldsymbol{v}_{\mathcal{I}}^{\hat{h}}(y) \rangle \notin \mathcal{I}(H)$, and so $\mathcal{V}_{\mathcal{I}}^{\hat{h}}(Hxy) \neq 1$. Thus $\mathcal{V}_{\mathcal{I}}^{\hat{h}}(\neg Hxy) = 1$.

Since $\hat{h}$ is a y-variant of $\hat{g}$, it follows that $\mathcal{V}_{\mathcal{I}}^{\hat{g}}(\exists y \neg Hxy) = 1$. By the same reasoning, $\mathcal{V}_{\mathcal{I}}^{\hat{e}}(\exists x \exists y \neg Hxy) = 1$ since $\hat{g}$ is a x-variant of $\hat{e}$. Thus $\mathcal{V}_{\mathcal{I}}(\exists x \exists y \neg Hxy) = 1$ since $\exists x \exists y \neg Hxy$ is a sentence of $\mathcal{L}^{\text{FOL}}$ and $\hat{e}$ was arbitrary.

Generalizing on $\mathcal{M}$, it follows that $\exists x \exists y \neg Hxy$ is true in every $\mathcal{L}^{\text{FOL}}$ model in which $\neg Hbc$ is true, and so $\neg Hbc \vDash \exists x \exists y \neg Hxy$ as desired. $\square$

Although it is sometimes easier to write *reductio* style proofs, direct proofs are typically more illuminating. In this case, we may observe that the premise requires b and c to not be H-related, and so by existentially generalising on b and c, we may conclude that there is some x and some y which are not H-related. This sort of reasoning is common.

Note that the logical consequence moved from a particular claim about some individuals to a quantified claim about some individuals so that the quantifiers only appear on the right side of the logical consequence. Were we to reverse the order of these sentences, the logical consequence would no longer hold: just because there are some things that are not H-related, it does not follow that b and c in particular are not H-related. We find just the opposite pattern of reasoning with universal quantifiers. For instance, suppose that we know that everybody loves Deeya: $\forall x Lxd$. It follows that Deeya loves herself since she is also somebody: Ldd. Thus we may establish the logical consequence $\forall x Lxd \vDash Ldd$ with the following proof:

Proof: Let $\mathcal{M} = \langle \mathbb{D}, \mathcal{I} \rangle$ be a $\mathcal{L}^{\text{FOL}}$ model where $\mathcal{V}_{\mathcal{I}}(\forall x Lxd) = 1$. It follows that $\mathcal{V}_{\mathcal{I}}^{\hat{a}}(\forall x Lxd) = 1$ for every v.a. $\hat{a}$. Assume for *reductio* that $\mathcal{V}_{\mathcal{I}}(Ldd) \neq 1$, and so $\mathcal{V}_{\mathcal{I}}^{\hat{c}}(Ldd) \neq 1$ for some v.a. $\hat{c}$ defined over $\mathbb{D}$. Thus $\langle \boldsymbol{v}_{\mathcal{I}}^{\hat{c}}(d), \boldsymbol{v}_{\mathcal{I}}^{\hat{c}}(d) \rangle \notin \mathcal{I}(L)$. Since d is a constant, we know by definition that $\langle \mathcal{I}(d), \mathcal{I}(d) \rangle \notin \mathcal{I}(L)$.

Let $\hat{e}$ be an x-variant of $\hat{c}$ where $\hat{e}(x) = \mathcal{I}(d)$. Given the above, we know that $\mathcal{V}_{\mathcal{I}}^{\hat{c}}(\forall x Lxd) = 1$, and so $\mathcal{V}_{\mathcal{I}}^{\hat{e}}(Lxd) = 1$ since $\hat{e}$ is an x-variant of $\hat{c}$. As a result, $\langle \boldsymbol{v}_{\mathcal{I}}^{\hat{e}}(x), \boldsymbol{v}_{\mathcal{I}}^{\hat{e}}(d) \rangle \in \mathcal{I}(L)$ where $\boldsymbol{v}_{\mathcal{I}}^{\hat{e}}(x) = \hat{e}(x) = \mathcal{I}(d)$ and $\boldsymbol{v}_{\mathcal{I}}^{\hat{e}}(d) = \mathcal{I}(d)$. Thus we may conclude that $\langle \mathcal{I}(d), \mathcal{I}(d) \rangle \in \mathcal{I}(L)$, contradicting the above.

By *reductio*, it follows that $\mathcal{V}_{\mathcal{I}}(Ldd) = 1$. Since $\mathcal{M}$ was an arbitrary model in which $\mathcal{V}_{\mathcal{I}}(\forall x Lxd) = 1$, it follows that $\forall x Lxd \vDash Ldd$ as desired. $\square$

This proof was considerably easier than the previous proof given above. Although reasoning from universal claims to particular claims tends to be easier than reasoning from particular claims to existential claims, each case requires careful consideration.

We will conclude with an example which includes mixed quantifiers, where such cases typically require the most care. In particular, consider the logical consequence: $\exists x \forall y Lxy \vDash \forall y \exists x Lxy$. For simplicity, we will provide a *reductio* proof as before:

> *Proof:* Assume for contradiction that there is a $\mathcal{L}^{\text{FOL}}$ model $\mathcal{M} = \langle \mathbb{D}, \mathcal{I} \rangle$ where $\mathcal{V}_{\mathcal{I}}(\exists x \forall y Lxy) = 1$ and $\mathcal{V}_{\mathcal{I}}(\forall y \exists x Lxy) = 0$. It follows that $\mathcal{V}_{\mathcal{I}}^{\hat{a}}(\exists x \forall y Lxy) = 1$ for every v.a. $\hat{a}$, and $\mathcal{V}_{\mathcal{I}}^{\hat{c}}(\forall y \exists x Lxy) = 0$ for some v.a. $\hat{c}$, and so $\mathcal{V}_{\mathcal{I}}^{\hat{c}}(\forall y \exists x Lxy) = 0$ in particular. Thus $\mathcal{V}_{\mathcal{I}}^{\hat{e}}(\forall y Lxy) = 1$ for some x-variant $\hat{e}$ of $\hat{c}$, and $\mathcal{V}_{\mathcal{I}}^{\hat{g}}(\exists x Lxy) = 0$ for some y-variant $\hat{g}$ of $\hat{c}$. It follows that $\hat{e}(y) = \hat{c}(y)$ and $\hat{g}(x) = \hat{c}(x)$.
>
> Given the above, $\mathcal{V}_{\mathcal{I}}^{\hat{e_1}}(Lxy) = 1$ for the y-variant $\hat{e_1}$ of $\hat{e}$ where $\hat{e_1}(y) = \hat{g}(y)$. Additionally, $\mathcal{V}_{\mathcal{I}}^{\hat{g_1}}(Lxy) = 0$ for the x-variant $\hat{g_1}$ of $\hat{g}$ where $\hat{g_1}(x) = \hat{e_1}(x)$. Since $x \neq y$, it follows that $\hat{g}(y) = \hat{g_1}(y)$, and so $\hat{g_1}(y) = \hat{e_1}(y)$ given the above.
>
> It follows that $\langle \mathcal{V}_{\mathcal{I}}^{\hat{e_1}}(x), \mathcal{V}_{\mathcal{I}}^{\hat{e_1}}(y) \rangle \in \mathcal{I}(L)$ and $\langle \mathcal{V}_{\mathcal{I}}^{\hat{g_1}}(x), \mathcal{V}_{\mathcal{I}}^{\hat{g_1}}(y) \rangle \notin \mathcal{I}(L)$, and since x and y are variables, $\langle \hat{e_1}(x), \hat{e_1}(y) \rangle \in \mathcal{I}(L)$ and $\langle \hat{g_1}(x), \hat{g_1}(y) \rangle \notin \mathcal{I}(L)$. However, given the identities above, if follows from the former that $\langle \hat{g_1}(x), \hat{g_1}(y) \rangle \in \mathcal{I}(L)$, thereby contradicting the latter. Thus $\exists x \forall y Lxy \vDash \forall y \exists x Lxy$. □

Given our *reductio* assumption, we began with two claims with an existential character evaluated at the same variable assignment $\hat{c}$. However, unpacking these claims split in two direction, yielding the x-variant $\hat{e}$ and the y-variant $\hat{g}$, where the result was two claims with a universal character. Since these claim entail something about *all* y-variants of $\hat{e}$ and *all* x-variants of $\hat{g}$ respectively, we chose $\hat{e_1}(y) = \hat{g}(y)$ and $\hat{g_1}(x) = \hat{e_1}(x)$ in order to get these variable assignments to clash. Since $\hat{g_1}$ was an x-variant of $\hat{g}$, we know that $\hat{g}(y) = \hat{g_1}(y)$, where making appropriate substitutions resulted in a contradiction.

8.9 Particular Models

We have already seen some tricky examples that require reasoning about all models. It remains to evaluate sentences at particular models. This differs from constructing models in which a given sentence is true or false since we are supposing the model to be provided. For instance, consider the following partial model $\mathcal{M}$:

$$\begin{aligned} \mathbb{D} &= \{1, 2, 3\} \\ \mathcal{I}(R) &= \{\langle 1, 2 \rangle, \langle 2, 1 \rangle, \langle 1, 3 \rangle\} \end{aligned}$$

Suppose that we want to show that $\forall x \exists y Rxy$ is false in $\mathcal{M}$. Reading 'R' as 'sees' for convenience, this claim says that everything sees something. However, looking into our model, we may observe that although 1 sees 2, and 2 sees 3, there is nothing in the domain $\mathbb{D}$ that 3 sees. Thus the claim is false. It remains to provide a proof.

Proof: Let $\hat{a}$ be a v.a. over $\mathbb{D}$ and $\hat{c}$ be an x-variant of $\hat{a}$ where $\hat{c}(x) = 3$. Next, we may let $\hat{g}$ be an arbitrary y-variant of $\hat{c}$. Since $\hat{g}$ is a y-variant of $\hat{c}$ and $x \neq y$, it follows that $\hat{g}(x) = \hat{c}(x) = 3$ where $\hat{g}(y) \in \mathbb{D}$. However, since $\langle 3, \mathbf{x} \rangle \notin \mathcal{I}(R)$ for all $\mathbf{x} \in \mathbb{D}$, we may conclude that $\langle \hat{g}(x), \hat{g}(y) \rangle \notin \mathcal{I}(R)$, and so $\langle \boldsymbol{v}_{\mathcal{I}}^{\hat{g}}(x), \boldsymbol{v}_{\mathcal{I}}^{\hat{g}}(y) \rangle \notin \mathcal{I}(R)$. Thus $\mathcal{V}_{\mathcal{I}}^{\hat{g}}(Rxy) \neq 1$ where $\hat{g}$ was an arbitrary y-variant of $\hat{c}$. It follows that $\mathcal{V}_{\mathcal{I}}^{\hat{c}}(\exists y Rxy) \neq 1$, and so $\mathcal{V}_{\mathcal{I}}^{\hat{a}}(\forall x \exists y Rxy) \neq 1$ since $\hat{c}$ is an x-variant of $\hat{a}$. We may then conclude that $\mathcal{V}_{\mathcal{I}}(\forall x \exists y Rxy) \neq 1$ as desired. □

This proof shows that $\forall x \exists y Rxy$ is false in $\mathcal{M}$. In just the same way, we may show that a sentence is true on a given model. For instance, consider the sentence $\forall x \exists y Ryx$. Maintaining our reading from before, this says that everything is seen by something. We can show that this sentence is true in the same model $\mathcal{M}$ by means of the following proof:

Proof: Let $\hat{a}$ be a v.a. defined over $\mathbb{D}$ and $\hat{c}$ be an x-variant of $\hat{a}$. It follows that $\hat{c}(x) \in \{1, 2, 3\}$, and so there are three cases to consider:

Case 1: Assume $\hat{c}(x) = 1$ and let $\hat{e}$ be a y-variant of $\hat{c}$ where $\hat{e}(y) = 2$. Thus $\langle \hat{e}(y), \hat{e}(x) \rangle \in \mathcal{I}(R)$ since $\hat{e}(x) = \hat{c}(x) = 1$, and so $\langle \boldsymbol{v}_{\mathcal{I}}^{\hat{e}}(y), \boldsymbol{v}_{\mathcal{I}}^{\hat{e}}(x) \rangle \in \mathcal{I}(R)$. Hence $\mathcal{V}_{\mathcal{I}}^{\hat{e}}(Ryx) = 1$, and so $\mathcal{V}_{\mathcal{I}}^{\hat{c}}(\exists y Ryx) = 1$ since $\hat{e}$ is a y-variant of $\hat{c}$.

Case 2: Assume $\hat{c}(x) = 2$ and let $\hat{e}$ be a y-variant of $\hat{c}$ where $\hat{e}(y) = 1$. Thus $\langle \hat{e}(y), \hat{e}(x) \rangle \in \mathcal{I}(R)$ since $\hat{e}(x) = \hat{c}(x) = 2$, and so $\langle \boldsymbol{v}_{\mathcal{I}}^{\hat{e}}(y), \boldsymbol{v}_{\mathcal{I}}^{\hat{e}}(x) \rangle \in \mathcal{I}(R)$. Hence $\mathcal{V}_{\mathcal{I}}^{\hat{e}}(Ryx) = 1$, and so $\mathcal{V}_{\mathcal{I}}^{\hat{c}}(\exists y Ryx) = 1$ since $\hat{e}$ is a y-variant of $\hat{c}$.

Case 3: Assume $\hat{c}(x) = 3$ and let $\hat{e}$ be a y-variant of $\hat{c}$ where $\hat{e}(y) = 1$. Thus $\langle \hat{e}(y), \hat{e}(x) \rangle \in \mathcal{I}(R)$ since $\hat{e}(x) = \hat{c}(x) = 3$, and so $\langle \boldsymbol{v}_{\mathcal{I}}^{\hat{e}}(y), \boldsymbol{v}_{\mathcal{I}}^{\hat{e}}(x) \rangle \in \mathcal{I}(R)$. Hence $\mathcal{V}_{\mathcal{I}}^{\hat{e}}(Ryx) = 1$, and so $\mathcal{V}_{\mathcal{I}}^{\hat{c}}(\exists y Ryx) = 1$ since $\hat{e}$ is a y-variant of $\hat{c}$.

Thus $\mathcal{V}_{\mathcal{I}}^{\hat{c}}(\exists y Ryx) = 1$ for every x-variant $\hat{c}$ of $\hat{a}$, and so $\mathcal{V}_{\mathcal{I}}^{\hat{a}}(\forall x \exists y Ryx) = 1$. We may conclude that $\mathcal{V}_{\mathcal{I}}(\forall x \exists y Ryx) = 1$ as desired. □

8.10 Conclusion

This chapter has presented one of the trickiest topics that we will cover in this course. Unlike the semantics for $\mathcal{L}^{\text{PL}}$, the semantics for $\mathcal{L}^{\text{FOL}}$ has a lot of moving pieces, and it is can be hard to prevent them from getting tangled. Even once you have mastered the definitions and can use them effectively to provide semantic arguments in the manner demonstrated above, this still takes quite a bit of work. To avoid having to provide complicated semantic arguments, the following chapter will extend PL to provide a natural deduction system for $\mathcal{L}^{\text{FOL}}$ called *first-order logic* (FOL). Although it will be somewhat easier to write proofs in FOL, there is no substitute for understanding the semantics for $\mathcal{L}^{\text{FOL}}$ itself. After all, logical consequence provides an important account of formal reasoning that we ought to expect our proof system to accommodate. We will attend to these details in due course. For the time being, there is no better way to master the semantics of $\mathcal{L}^{\text{FOL}}$ than working through problems for yourself.

Chapter 9

Identity

The last two chapters introduced the syntax and semantics for $\mathcal{L}^{\text{FOL}}$. In this chapter, we will extend both the syntax and semantics of $\mathcal{L}^{\text{FOL}}$ to accommodate the IDENTITY predicate '=', referring to this extended language as $\mathcal{L}^{=}$.

It is important to emphasize that to say that **x** and **y** are identical is different from saying that **x** and **y** are *duplicates* though this is a common way of using the word 'identical' in English. For instance, consider the use of the word 'identical' in the following case:

> `Spheres`: Consider a possible world in which there is nothing but two identical spheres, similar in every way to each other, separated by just one meter. Although there is no property that they do not share in common, the two spheres are distinct. After all, there are two spheres, not just one.

Insofar as the spheres are distinct— there are two of them, not one— we will say that they are not *numerically identical*, or just not *identical* for short. If 'a' names one sphere and 'b' names the other, we may express this with the sentence '$\neg{=}ab$' where '=' is a two place predicate, or with '$a \neq b$' for the sake of readability and familiarity. In this sense of identity, it is not true that the two spheres are identical as claimed in `Spheres`. Indeed, no two things whatsoever are identical in our sense since if they were, then there would not be two of them but rather just one thing perhaps with different names.

Before showing how to include a designated predicate for identity in the syntax and semantics for $\mathcal{L}^{=}$, it will help to guide our ambitions by considering some of what motivates this addition. After all, $\mathcal{L}^{\text{FOL}}$ is a very powerful language, at least by comparison to $\mathcal{L}^{\text{PL}}$. Why should we need to further extend $\mathcal{L}^{\text{FOL}}$? Can't we get by without including identity in the language? In particular, we could take a 2-place predicate in the language (e.g., I) to symbolize identity in just the same way as we would for any other 2-place predicate. Why should identity deserve special treatment, and how should we think about the difference between identity and the other predicates that we will included in the language $\mathcal{L}^{=}$?

9.1 Identity and Logic

It turns out that there is a lot that cannot be said without an identity predicate. You might be wondering why we can't just declare that a certain predicate be used to express identity the way that we do in regimenting other predicates in $\mathcal{L}^{\text{FOL}}$. For instance, suppose we were to regiment 'Hesperus is Phosphorus' as 'Ihp' given the following symbolization key:

Ixy: x is y
h: Hesperus
p: Phosphorus
v: Venus

One might take the regimentation given above to do as good a job as any of our regimentations in $\mathcal{L}^{\text{FOL}}$. Why does identity deserve special treatment?

Consider the following English argument regimented with the symbolization given above:

A1. Hesperus is Phosphorus.	B1. Ihp
A2. Phosphorus is Venus.	B2. Ipv
A3. Hesperus is Venus.	B3. Ihv

This argument is invalid. For instance, consider the following countermodel:

$$\begin{aligned} \mathbb{D} &= \{h, p, v\} \\ \mathcal{I}(I) &= \{\langle h, p\rangle, \langle p, v\rangle\} \\ \mathcal{I}(h) &= h \\ \mathcal{I}(p) &= p \\ \mathcal{I}(v) &= v \end{aligned}$$

Since $\langle h, p\rangle, \langle p, v\rangle \in \mathcal{I}(I)$ but $\langle h, v\rangle \notin \mathcal{I}(I)$, it follows that $\langle \mathcal{I}(h), \mathcal{I}(p)\rangle, \langle \mathcal{I}(p), \mathcal{I}(v)\rangle \in \mathcal{I}(I)$ but $\langle \mathcal{I}(h), \mathcal{I}(v)\rangle \notin \mathcal{I}(I)$. Given any variable assignment $\hat{a}$, it follows by definition that both $\langle \boldsymbol{v}_{\mathcal{I}}^{\hat{a}}(h), \boldsymbol{v}_{\mathcal{I}}^{\hat{a}}(p)\rangle, \langle \boldsymbol{v}_{\mathcal{I}}^{\hat{a}}(p), \boldsymbol{v}_{\mathcal{I}}^{\hat{a}}(v)\rangle \in \mathcal{I}(I)$ while $\langle \boldsymbol{v}_{\mathcal{I}}^{\hat{a}}(h), \boldsymbol{v}_{\mathcal{I}}^{\hat{a}}(v)\rangle \notin \mathcal{I}(I)$. It follows from the semantics that $\mathcal{V}_{\mathcal{I}}^{\hat{a}}(Ihp) = \mathcal{V}_{\mathcal{I}}^{\hat{a}}(Ipv) = 1$ and yet $\mathcal{V}_{\mathcal{I}}^{\hat{a}}(Ihv) \neq 1$. Since $\hat{a}$ was an arbitrary v.a. and Ihp, Ipv, and Ihv are all wfss of $\mathcal{L}^{\text{FOL}}$, we may conclude that $\mathcal{V}_{\mathcal{I}}(Ihp) = \mathcal{V}_{\mathcal{I}}(Ipv) = 1$ and yet $\mathcal{V}_{\mathcal{I}}(Ihv) \neq 1$. Having produced a model which makes the premises true and the conclusion false, it follows that the argument is not valid.

An analogous argument shows that the following argument is also invalid:

Txy: x is taller than y
k: Kate
s: Sam
l: Lu

C1. Kate is taller than Sam.
C2. Sam is taller than Lu.
C3. Kate is taller than Lu.

$$\begin{aligned} \mathbb{D} &= \{k, s, l\} \\ \mathcal{I}(T) &= \{\langle k, s\rangle, \langle s, l\rangle\} \\ \mathcal{I}(k) &= k \\ \mathcal{I}(s) &= s \\ \mathcal{I}(l) &= l \end{aligned}$$

D1. Tks
D2. $\underline{Tsl\quad}$
D3. Tkl

Replacing 'I' with 'T' and similarly replacing 'h' with 'k' and so on for the other constants, the semantic proof given above could be adapted to show that the premises in the previous argument do not entail the conclusion. Here we may ask if this is right, and if so, why we shouldn't say the very same thing about the identity argument.

Certainly it should be admitted that the taller-than argument is a very strong argument in ordinary contexts. After all, given the intended interpretation of English, any *possibility* in which Kate is taller than Sam and in which Sam is taller than Lu is also a possibility in which Kate is taller than Lu. The reason the argument is invalid is that nothing forces us to interpret the predicate 'is taller than' as meaning what it usually means. Put otherwise, the dyadic predicate 'is taller than' is a NON-LOGICAL term of our language, and is to be regimented by a non-logical dyadic predicate in $\mathcal{L}^{\text{FOL}}$ which we may interpret by any subset of $\mathbb{D}^2$ given any domain $\mathbb{D}$ whatsoever. This makes the argument easy to invalidate.

In order to make the taller-than argument valid, we would have to add an additional premise. For instance, here are two ways that we might make the argument valid:

E1. $\forall x\forall y\forall z((Txy \wedge Tyz) \rightarrow Txz)$
E2. Tks
E3. $\underline{Tsl\quad}$
E4. Tkl

F1. $(Tks \wedge Tsl) \rightarrow Tkl$
F2. Tks
F3. $\underline{Tsl\quad}$
F4. Tkl

Both of the arguments above are valid. Whereas the argument on the left starts off by asserting that the taller-than relation is transitive, the argument on the right appeals to a particular instances of the transitivity of the taller-than relation.

It is natural to assume that on the intended interpretation of 'is taller than' in English, we mean to express a transitive relation since this is how heights behave. In reasoning from Tks and Tsl to Tkl, we are implicitly relying on our intuitive grasp of a particular interpretation rather than general logical features of the sentences involved. That is, the argument is convincing not because of its logical form, but because of the particular interpretation that we are assuming, i.e., where 'is taller than' expresses a transitive relation.

When we add a premises which requires T to be transitive (or else add the relevant instance), we are making our assumptions about how to interpret 'is taller than' explicit in a way that avoids reliance on a particular intended interpretation of our language. The amended arguments are valid since the conclusions are true in any model in which all of the premises true. By showing that the conclusion is true when the premises are true in any model of $\mathcal{L}^{\text{FOL}}$ whatsoever, we are interpreting the non-logical terms in all possible ways while holding the meanings of the logical terms fixed. Accordingly, we may conclude that the conclusion follows by virtue of the logical forms of the sentences involved and not a particular interpretation.

What about the identity argument? Certainly we could reproduce a similar story, claiming that as it stands, the identity argument we started off with is not valid but could be made valid by adding a premise that requires identity to be transitive. The question is whether this would be appropriate in the case of the identity predicate. More specifically, is it permissible to interpret identity as *any* subset of $\mathbb{D}^2$ over any domain $\mathbb{D}$?

The answer is certainly 'Yes' since this is exactly how we would interpret the identity argument in $\mathcal{L}^{\text{FOL}}$. Nevertheless, there is good reason not to go this way, choosing instead to include a designated predicate for identity in $\mathcal{L}^{=}$. Recast in $\mathcal{L}^{=}$, the identity argument becomes:

G1. $h = p$
G2. $p = v$
G3. $h = v$

Instead of taking this argument to only be convincing when we restrict consideration to an intended interpretation where '=' means identity, we are taking identity to be a logical notion akin to negation, conjunction, and the quantifiers. Rather than relying on the intended interpretation of our language to tell us what identity means, we are going to provide a semantic clause for identity which holds its meaning fixed across all interpretations in just the same way that we did for the other logical terms included in our language. As a result, the argument above will turn out to be valid as it stands.

You might be wondering why we don't do something similar for the 'is taller than' predicate, and so on for other notions like 'between', or 'is older than', etc. There are two reasons worth considering. The first is that there is no clear stopping point. Were we to start expanding the range of logical predicates whose interpretation we hold fixed by providing semantic clauses, we could go and go forever. This in itself does not require that we do so— we could just choose to include certain predicates in the logical vocabulary of our language and not others given our purposes. The second reason is more forceful: in order to provide a semantic clause for the taller-than predicate 'T', we would have to provide a theory of what it is for one thing to be taller than another. Without providing such a theory, nothing guarantees that T is transitive, and so the taller-than argument would remain invalid.

Although one might attempt to provide a theory of the taller-than relation, doing so reaches beyond the subject-matter of logic. Moreover, it would be natural to use a language like $\mathcal{L}^{=}$ in order to develop such a theory. The same cannot so easily be said for identity. Instead of falling outside the subject-matter of logic, identity is taken to fit squarely within our present aim to develop the most basic conceptual resources that we need to articulate theories. Instead of requiring that we develop an independent theory of identity, the semantics for identity will rely on our understanding of identity in the metalanguage in the same way that the semantics for negation relied on an understanding of negation in the metalanguage. In doing so, we are making our intuitive grasp on the logical terms in English explicit.

Before pressing on, it is worth considering three more cases involving identity. To begin with, consider the following example originally presented by Gottlob Frege:

Rx: x is rising.
h: Hesperus
p: Phosphorus

H1. Hesperus is rising.
H2. Hesperus is Phosphorus.
H3. Phosphorus is rising.

As specified below, identity is a primitive symbol of $\mathcal{L}^{=}$. Accordingly, we do not need to include identity in the symbolization key given above to provide the following regimentation.

I1. Rh
I2. $h = p$
I3. Rp

As we will soon see, this is a perfectly valid argument. Instead of restricting consideration to an intended interpretation, or else adding some further assumptions, the conclusion is entailed by the premises given the semantics that we will provide for $\mathcal{L}^{=}$.

Next consider the regimentation of the following arguments:

Lxy: x loves y
c: Cara
p: Pedro
d: DJ Faro

J1. Only Cara loves Pedro.
J2. DJ Faro loves Pedro.
J3. Cara is DJ Faro.

K1. $\forall x(Lxp \leftrightarrow x = c)$
K2. Ldp
K3. $c = d$

This is a valid argument. Although we could say that Cara loves Pedro in $\mathcal{L}^{\text{FOL}}$, we could not say that *only* Cara loves Pedro in $\mathcal{L}^{\text{FOL}}$ since to do so we would need to say that anything that loves Pedro is identical to Cara in addition to saying that Cara loves Pedro. Here we may accomplish both claims at once by saying that for anything, it loves Pedro just in case it is identical to Cara. Since Cara is identical to herself, she must love Pedro, and moreover, for anything that loves Pedro, it must be identical to Cara. Since DJ Faro loves Pedro, we may conclude that DJ Faro must be identical to Cara. Reasoning in this way requires that we extend the expressive power of $\mathcal{L}^{\text{FOL}}$ by including identity in the language.

The example above suggests the manner in which certain properties may serve to uniquely identify a given object. In doing so, we are effectively saying there is one thing which has a given property. More generally, identity can be used to say how many things have a given property, or else satisfy some combination of properties. For instance, consider the following sentence and its various competing regimentations:

L1. Mozart composed at least two things.

Cxy: x composed y		M1. $\exists xCmx \wedge \exists yCmy$.
m: Mozart		M2. $(\exists xCmx \wedge \exists yCmy) \wedge x \neq y$.
		M3. $\exists x\exists y((Cmx \wedge Cmy) \wedge x \neq y)$.

Although M1 regiments the sentence L1 in $\mathcal{L}^{\text{FOL}}$, this regimentation does not require that there are at least two things that Mozart composed. This is because both conjuncts could be satisfied by the same thing, and so L1 would be true if there was just one thing that Mozart composed. The regimentation given in M2 is worse since this is not a wfs of $\mathcal{L}^{\text{FOL}}$. Rather, M2 is an open sentence since it includes two free variables which fall outside of the scope of the quantifiers. By contrast, sentence M3 provides an adequate regimentation, though does so by making the quantifiers have scope over all instances of x and y.

The success of M3 as a regimentation of L1 has profound consequences for it means that we can regiment 'at least two' in $\mathcal{L}^{=}$. As we will soon see, we may also regiment 'at most two', where 'exactly two' will be regimented by their conjunction. This means that cardinality may be expressed in $\mathcal{L}^{=}$ by answering questions of the form "How many things are such that φ" with wfss that regiment claims of the form "Exactly n things are such that φ". By contrast, such expressions cannot be regimented in $\mathcal{L}^{\text{FOL}}$ without identity.

The cases considered above demonstrate that if we want to capture the logical relationships having to do with identity, we need designated logical vocabulary to do so. Just as we introduced the '$\forall$' and '$\exists$' to regiment quantified claims, we also need a special symbol '$=$' for identity. Whereas the following section will specify a syntax for our expanded language, we will then proceed to provide its semantics the section after.

9.2 The Syntax for $\mathcal{L}^{=}$

The primitive symbols included in $\mathcal{L}^{=}$ are exactly the same as those included in $\mathcal{L}^{\text{FOL}}$ with the single addition of the identity predicate '$=$'. Thus we have the following:

n-place predicates for $n \geqslant 0$ with subscripts, as needed	$A^n, B^n, C^n, \ldots, Z^n$ $A^n_1, B^n_1, Z^n_1, A^n_2, A^n_{25}, J^n_{375}, \ldots$
constants with subscripts, as needed	$a, b, c, \ldots, v$ $a_1, w_4, h_7, m_{32}, \ldots$
variables with subscripts, as needed	w, x, y, z $x_1, y_1, z_1, x_2, \ldots$
sentential connectives	$\neg, \wedge, \vee, \rightarrow, \leftrightarrow$
identity	$=$
quantifiers	$\forall, \exists$
parentheses	(,)

Here we have included '=' in our alphabet of primitive symbols. This may seem like a small change and given the examples above, it may be obvious to you how to build up sentences in $\mathcal{L}^{=}$. Nevertheless, we need to define the wffs of $\mathcal{L}^{=}$ afresh, where something similar will be repeated for our other recursive definitions that we provided before.

Although much will be as it was before, it is important to attend to the differences that occur throughout the definitions given in the following two sections. In addition to reviewing the details that remain the same, hopefully our methodology should be beginning to feel familiar. As we expand the expressive power of our language, we need to say what counts as a wfs of that language. If the language includes variables, we will need to first say what counts as a wff of the language. By defining what it is for a variable to be free in a wff, we may specify that the wfss of the language are those wffs without free variables. In just the same way that we did for $\mathcal{L}^{\text{FOL}}$, we will now provide these definitions for $\mathcal{L}^{=}$.

Whereas there was one way to form atomic wffs in $\mathcal{L}^{\text{FOL}}$, we now have two ways to form atomic wffs. Thus we will define the WELL-FORMED FORMULAS (wffs) of $\mathcal{L}^{=}$ as follows:

1. $\mathcal{F}^n\alpha_1,\ldots,\alpha_n$ is a wff of $\mathcal{L}^{=}$ if $\mathcal{F}^n$ is an n-place predicate of $\mathcal{L}^{=}$ and $\alpha_1,\ldots,\alpha_n$ are singular terms (i.e., variables or constants) of $\mathcal{L}^{=}$.
2. $\alpha = \beta$ is a wff of $\mathcal{L}^{=}$ if α and β are singular terms of $\mathcal{L}^{=}$.
3. For any wffs φ and ψ of $\mathcal{L}^{=}$ and variable α of $\mathcal{L}^{=}$:
   (a) $\exists\alpha\varphi$ is a wff of $\mathcal{L}^{=}$;
   (b) $\forall\alpha\varphi$ is a wff of $\mathcal{L}^{=}$;
   (c) $\neg\varphi$ is a wff of $\mathcal{L}^{=}$;
   (d) $(\varphi \wedge \psi)$ is a wff of $\mathcal{L}^{=}$;
   (e) $(\varphi \vee \psi)$ is a wff of $\mathcal{L}^{=}$;
   (f) $(\varphi \rightarrow \psi)$ is a wff of $\mathcal{L}^{=}$; and
   (g) $(\varphi \leftrightarrow \psi)$ is a wff of $\mathcal{L}^{=}$.
4. Nothing else is a wff of $\mathcal{L}^{=}$.

Officially, the clauses above are non-sense, and can only be made sense of by adding corner quotes in appropriate places. Having explained how to do this above, we will rely on the reader to know where these corner quotes are implicitly intended as we did before.

We may either form atomic wffs as we did in $\mathcal{L}^{\text{FOL}}$, or we may form wffs with the identity predicate together with two singular terms. Nevertheless, nothing requires identity wffs to be wfss since they may include free variables in just the same way that 2-place predicates may combine with free variables. This means that there is a new way for free variables to occur in wffs and so we will have to extend our definition of free variables accordingly:

1. α is free in $\mathcal{F}^n\alpha_1, \ldots, \alpha_n$ if $\alpha = \alpha_i$ for some $1 \leqslant i \leqslant n$ where α is a variable, $\mathcal{F}^n$ is an n-place predicate, and $\alpha_1, \ldots, \alpha_n$ are singular terms.
2. α is free in $\beta = \gamma$ if $\alpha = \beta$ or $\alpha = \gamma$ where α is a variable.
3. If φ and ψ are wffs of $\mathcal{L}^=$ and α and β are variables, then:
   (a) α is free in $\exists\beta\varphi$ if α is free in φ and $\alpha \neq \beta$;
   (b) α is free in $\forall\beta\varphi$ if α is free in φ and $\alpha \neq \beta$;
   (c) α is free in $\neg\varphi$ if α is free in φ;
   (d) α is free in $(\varphi \wedge \psi)$ if α is free in φ or α is free in ψ;
   (e) α is free in $(\varphi \vee \psi)$ if α is free in φ or α is free in ψ;
   (f) α is free in $(\varphi \rightarrow \psi)$ if α is free in φ or α is free in ψ;
   (g) α is free in $(\varphi \leftrightarrow \psi)$ if α is free in φ or α is free in ψ;
4. Nothing else is a free variable.

Given the definition of free variables in $\mathcal{L}^=$, we may define a WELL-FORMED SENTENCE (wfs) of $\mathcal{L}^=$ to be any wffs of $\mathcal{L}^=$ which does not include any free variables, and an OPEN SENTENCE of $\mathcal{L}^=$ is any wff of $\mathcal{L}^=$ which does include free variables.

In order to establish that all wfss of $\mathcal{L}^=$ have certain properties, it will be important to be able to organize the wfss of $\mathcal{L}^=$ into a countable sequence of stages. To do so, we will define the COMPLEXITY $\texttt{Comp}(\varphi) \in \mathbb{N}$ of any wff of $\mathcal{L}^=$ to be the number of occurrences of sentential operators that belong to $\mathcal{L}^=$, where both $\forall\alpha$ and $\exists\alpha$ are unary operators for any variable α.

1. $\texttt{Comp}(\mathcal{F}^n\alpha_1, \ldots, \alpha_n) = 0$ if $\mathcal{F}^n$ is an n-place predicate of $\mathcal{L}^=$ and $\alpha_1, \ldots, \alpha_n$ are singular terms (i.e., variables or constants) of $\mathcal{L}^=$.
2. $\texttt{Comp}(\alpha = \beta) = 0$ if α and β are singular terms of $\mathcal{L}^=$.
3. For any wffs φ and ψ of $\mathcal{L}^=$ and variable α of $\mathcal{L}^=$:
   (a) $\texttt{Comp}(\exists\alpha\varphi) = \texttt{Comp}(\varphi) + 1$;
   (b) $\texttt{Comp}(\forall\alpha\varphi) = \texttt{Comp}(\varphi) + 1$;
   (c) $\texttt{Comp}(\neg\varphi) = \texttt{Comp}(\varphi) + 1$;
   (d) $\texttt{Comp}(\varphi \wedge \psi) = \texttt{Comp}(\varphi) + \texttt{Comp}(\psi) + 1$;
   (e) $\texttt{Comp}(\varphi \vee \psi) = \texttt{Comp}(\varphi) + \texttt{Comp}(\psi) + 1$;
   (f) $\texttt{Comp}(\varphi \rightarrow \psi) = \texttt{Comp}(\varphi) + \texttt{Comp}(\psi) + 1$;
   (g) $\texttt{Comp}(\varphi \leftrightarrow \psi) = \texttt{Comp}(\varphi) + \texttt{Comp}(\psi) + 1$;

Given these definitions, we may now turn to interpret the sentences of $\mathcal{L}^=$.

9.3 The Semantics for $\mathcal{L}^=$

Since the identity predicate '=' belongs to the logical vocabulary of $\mathcal{L}^=$, including identity in the list or primitive symbols does not effect the manner in which the non-logical symbols are interpreted (i.e., the constants or predicates), no change is required to the definition of a model. Accordingly, $\mathcal{L}^=$ and $\mathcal{L}^{\text{FOL}}$ have precisely the same models. As in Chapter 8, we may define a MODEL of $\mathcal{L}^=$ to be any ordered pair $\mathcal{M} = \langle \mathbb{D}, \mathcal{I} \rangle$ where the DOMAIN $\mathbb{D}$ is any nonempty set and the INTERPRETATION $\mathcal{I}$ over $\mathbb{D}$ satisfies the following conditions:

Constants: $\mathcal{I}(\alpha) \in \mathbb{D}$ for every constant α of $\mathcal{L}^=$.

Predicates: $\mathcal{I}(\mathcal{F}^n) \subseteq \mathbb{D}^n$ for every n-place predicate $\mathcal{F}^n$ of $\mathcal{L}^=$ where $n \geqslant 0$.

Recall the manner in which VARIABLE ASSIGNMENTS were defined over a domain $\mathbb{D}$ to be any function $\hat{a}$ from the variables in $\mathcal{L}^{\text{FOL}}$ to elements of $\mathbb{D}$. Again, no change is required since neither the variables nor the domains that we might consider are effected by the addition of the identity predicate to the language. For a similar reason, we may also preserve the definition of a α-VARIANT of $\hat{a}$ as any variable assignment $\hat{c}$ where $\hat{c}(\beta) = \hat{a}(\beta)$ for every variable $\beta \neq \alpha$. Lastly, we define the VALUE (REFERENCE) of singular terms as before:

$$v_{\mathcal{I}}^{\hat{a}}(\alpha) = \begin{cases} \mathcal{I}(\alpha) & \text{if } \alpha \text{ is a constant} \\ \hat{a}(\alpha) & \text{if } \alpha \text{ is a variable.} \end{cases}$$

So far, all of the semantic definitions have remained just as they were in $\mathcal{L}^{\text{FOL}}$. Nevertheless, the semantics for $\mathcal{L}^=$ will differ insofar as it includes an extra clause for identity, mirroring the changes we made to the definition of the wffs of $\mathcal{L}^=$ given above.

VALUATION FUNCTION: For any wffs φ and ψ of $\mathcal{L}^=$, n-place predicate $\mathcal{F}^n$ of $\mathcal{L}^=$, and n singular terms $\alpha_1, \ldots, \alpha_n$ of $\mathcal{L}^=$:

$\mathcal{V}_{\mathcal{I}}^{\hat{a}}(\mathcal{F}^n\alpha_1, \ldots, \alpha_n) = 1$ *iff* $\langle v_{\mathcal{I}}^{\hat{a}}(\alpha_1), \ldots, v_{\mathcal{I}}^{\hat{a}}(\alpha_n) \rangle \in \mathcal{I}(\mathcal{F}^n)$.

$\mathcal{V}_{\mathcal{I}}^{\hat{a}}(\alpha = \beta) = 1$ just in case $v_{\mathcal{I}}^{\hat{a}}(\alpha) = v_{\mathcal{I}}^{\hat{a}}(\beta)$.

$\mathcal{V}_{\mathcal{I}}^{\hat{a}}(\forall\alpha\varphi) = 1$ *iff* $\mathcal{V}_{\mathcal{I}}^{\hat{c}}(\varphi) = 1$ for every α-variant $\hat{c}$ of $\hat{a}$.

$\mathcal{V}_{\mathcal{I}}^{\hat{a}}(\exists\alpha\varphi) = 1$ *iff* $\mathcal{V}_{\mathcal{I}}^{\hat{c}}(\varphi) = 1$ for some α-variant $\hat{c}$ of $\hat{a}$.

$\mathcal{V}_{\mathcal{I}}^{\hat{a}}(\neg\varphi) = 1$ *iff* $\mathcal{V}_{\mathcal{I}}^{\hat{a}}(\varphi) \neq 1$.

$\mathcal{V}_{\mathcal{I}}^{\hat{a}}(\varphi \vee \psi) = 1$ *iff* $\mathcal{V}_{\mathcal{I}}^{\hat{a}}(\varphi) = 1$ or $\mathcal{V}_{\mathcal{I}}^{\hat{a}}(\psi) = 1$ (or both).

$\mathcal{V}_{\mathcal{I}}^{\hat{a}}(\varphi \wedge \psi) = 1$ *iff* $\mathcal{V}_{\mathcal{I}}^{\hat{a}}(\varphi) = 1$ and $\mathcal{V}_{\mathcal{I}}^{\hat{a}}(\psi) = 1$.

$\mathcal{V}_{\mathcal{I}}^{\hat{a}}(\varphi \rightarrow \psi) = 1$ *iff* $\mathcal{V}_{\mathcal{I}}^{\hat{a}}(\varphi) = 0$ or $\mathcal{V}_{\mathcal{I}}^{\hat{a}}(\psi) = 1$ (or both).

$\mathcal{V}_{\mathcal{I}}^{\hat{a}}(\varphi \leftrightarrow \psi) = 1$ *iff* $\mathcal{V}_{\mathcal{I}}^{\hat{a}}(\varphi) = \mathcal{V}_{\mathcal{I}}^{\hat{a}}(\psi)$.

Having added the clause for identity, the other clauses continue to apply. It is worth comparing the semantic clause for identity to the clause for negation and considering the following worry.

> `Homophonic`: The semantic clause for identity doesn't tell us anything because we have used identity— indeed the same symbol— on both sides of the semantic clause. So in order to know about whether an identity sentence such as $\alpha = \beta$ is true in a model on an assignment, we already need to know what is identical to what. Thus the semantics does not tell us anything we didn't already know.

If we were attempting to understand what '=' means without drawing on any previous understanding, then certainly we should agree that the semantic clauses given above would be a complete failure. As brought out before, the very same thing may be said for negation, conjunction, disjunction, and the quantifiers. In each of these cases, analogues of the terms with which we are concerned appear in the metalanguage and play a critical role in stating the semantic clauses. Thus we cannot lean on our semantics to learn what these terms mean without any prior understanding of at least their analogues in the metalanguage.

All of this we must learn to accept. Where the complaint above goes wrong is in assuming that semantics ought to draw on independently understood terms in order to shed light on new terms. Instead of constructing something out nothing, the semantic clauses allow us to use the meanings we already grasp in English to interpret a simplified formal language in a way that is both systematic and explicit. Identity is no exception, though perhaps even more poignant given that we have used the same symbol in the metalanguage for identity.

Having defined truth relative to a model and a v.a., we are now in a position to specify what it is for a wfs of $\mathcal{L}^{=}$ to be true in a model independent of an v.a.:

> Theory of Truth: For any wfs φ and model $\mathcal{M} = \langle \mathbb{D}, \mathcal{I} \rangle$ of $\mathcal{L}^{=}$:
>
> $\mathcal{V}_{\mathcal{I}}(\varphi) = 1$ just in case $\mathcal{V}_{\mathcal{I}}^{\hat{a}}(\varphi) = 1$ for every v.a. $\hat{a}$ defined over $\mathbb{D}$.

This definition is just as it was before save that '$\mathcal{L}^{\text{FOL}}$' has been replaced by '$\mathcal{L}^{=}$'. For completeness, we will copy over the definitions of logical consequence $\vDash$ given its importance, though the other semantic definitions are just as they were in $\mathcal{L}^{\text{FOL}}$ and $\mathcal{L}^{\text{PL}}$.

> Logical Consequence: For any set of wfss $\Gamma \cup \{\varphi\}$ of $\mathcal{L}^{=}$:
>
> $\Gamma \vDash \varphi$ *iff* for any model $\mathcal{M} = \langle \mathbb{D}, \mathcal{I} \rangle$ of $\mathcal{L}^{=}$, if $\mathcal{V}_{\mathcal{I}}(\gamma) = 1$ for all $\gamma \in \Gamma$, then $\mathcal{V}_{\mathcal{I}}(\varphi) = 1$.

Although there is a lot of redundancy with the syntax and semantics that we provided for $\mathcal{L}^{\text{FOL}}$, hopefully providing the details in full helps to give you a good overview of all of the working piece that make up these theories. In the remaining sections of this chapter, we will put these theories to work in order to evaluate sentences and arguments in $\mathcal{L}^{=}$ that we could not adequate regiment in $\mathcal{L}^{\text{FOL}}$. As we will see, the language $\mathcal{L}^{=}$ is very powerful and perhaps for this reason has become a *lingua franca* by which a wide range of theories have been developed. One especially prominent example is set theory where the dyadic predicate '$\in$' for set-membership may be axiomatized in $\mathcal{L}^{=}$.

As already evident from the previous chapter, the semantics for $\mathcal{L}^{\text{FOL}}$ and $\mathcal{L}^{=}$ has a lot of moving parts and it can be very easy to get tied up in knots while attempting to write semantic proofs. The good news is that soon we will introduce a sound and complete proof system for $\mathcal{L}^{=}$, limiting the need to writing semantic proofs that $\Gamma \vDash \varphi$, or that φ is a tautology or, similarly, a contradiction. Nevertheless, we will need to write semantic proofs to show that $\Gamma \nvDash \varphi$, or φ is not a tautology or, not a contradiction. In order to streamline the semantic proofs that we will write, it will help to establish two supporting lemmas which will help us simplify the use of v.a.s in our proofs. We will establish these results in the following section before presenting a number applications in later sections.

9.4 Assignment Lemmas

Recall that a wfs φ of $\mathcal{L}^{=}$ is true in a model $\mathcal{M} = \langle \mathbb{D}, \mathcal{I} \rangle$ just in case it is true in true in that model relative to all v.a.s defined over the domain $\mathbb{D}$. This definition captures the intuition that the truth-value of a wfs φ in a given model of $\mathcal{L}^{=}$ should be independent of the v.a.s defined over the domain of that model since φ does not have any free variables. Nevertheless, it is not always convenient to be told that φ is true in a model $\mathcal{M}$ relative to *all* v.a.s defined over the domain of that model, and similarly, it is not always convenient to show that φ is true in a model $\mathcal{M}$ relative to *all* v.a.s defined over the domain of that model. Since a wfs has no free variables, if a wfs φ is true in all v.a.s defined over the domain of the model in question, then φ should be true in *some* v.a. defined over the domain, and *vice versa*. In order to establish this equivalence, we may now provide proofs of the following lemmas.

Lemma 9.1 If $\hat{a}(\alpha) = \hat{c}(\alpha)$ for all free variables α in a wff φ of $\mathcal{L}^{=}$, then $\mathcal{V}_{\mathcal{I}}^{\hat{a}}(\varphi) = \mathcal{V}_{\mathcal{I}}^{\hat{c}}(\varphi)$.

Letting $\mathcal{M} = \langle \mathbb{D}, \mathcal{I} \rangle$ be a model of $\mathcal{L}^{=}$, the proof proceeds by induction on the complexity of wffs of $\mathcal{L}^{=}$ where now there are two cases in the base step.

Base Step: Letting φ be a wff of $\mathcal{L}^{=}$ where $\varphi(\texttt{Comp}) = 0$ and both $\hat{a}$ and $\hat{c}$ are v.a.s defined over $\mathbb{D}$, we may assume that $\hat{a}(\alpha) = \hat{c}(\alpha)$ for all free variables α in φ. It follows that either $\varphi = \ulcorner \mathcal{F}^n \alpha_1, \ldots, \alpha_n \urcorner$ or $\varphi = \ulcorner \alpha_1 = \alpha_n \urcorner$ where in the latter case $n = 2$ and using corner quotes to improve clarity. In either case, $\boldsymbol{v}_{\mathcal{I}}^{\hat{a}}(\alpha_i) = \boldsymbol{v}_{\mathcal{I}}^{\hat{c}}(\alpha_i)$ for all $1 \leqslant i \leqslant n$. We may then consider the following biconditionals:

$$\begin{array}{rll} \mathcal{V}_{\mathcal{I}}^{\hat{a}}(\varphi) = 1 & \textit{iff} & \mathcal{V}_{\mathcal{I}}^{\hat{a}}(\mathcal{F}\alpha_1, \ldots, \alpha_n) = 1 \\ & \textit{iff} & \langle \boldsymbol{v}_{\mathcal{I}}^{\hat{a}}(\alpha_1), \ldots, \boldsymbol{v}_{\mathcal{I}}^{\hat{a}}(\alpha_n) \rangle \in \mathcal{I}(\mathcal{F}^n) \\ (\star) & \textit{iff} & \langle \boldsymbol{v}_{\mathcal{I}}^{\hat{c}}(\alpha_1), \ldots, \boldsymbol{v}_{\mathcal{I}}^{\hat{c}}(\alpha_n) \rangle \in \mathcal{I}(\mathcal{F}^n) \\ & \textit{iff} & \mathcal{V}_{\mathcal{I}}^{\hat{c}}(\mathcal{F}\alpha_1, \ldots, \alpha_n) = 1 \\ & \textit{iff} & \mathcal{V}_{\mathcal{I}}^{\hat{c}}(\varphi) = 1. \end{array}$$

$$\begin{array}{rll} \mathcal{V}_{\mathcal{I}}^{\hat{a}}(\varphi) = 1 & \textit{iff} & \mathcal{V}_{\mathcal{I}}^{\hat{a}}(\alpha_1 = \alpha_n) = 1 \\ & \textit{iff} & \boldsymbol{v}_{\mathcal{I}}^{\hat{a}}(\alpha_1) = \boldsymbol{v}_{\mathcal{I}}^{\hat{a}}(\alpha_n) \\ (\star) & \textit{iff} & \boldsymbol{v}_{\mathcal{I}}^{\hat{c}}(\alpha_1) = \boldsymbol{v}_{\mathcal{I}}^{\hat{c}}(\alpha_n) \\ & \textit{iff} & \mathcal{V}_{\mathcal{I}}^{\hat{c}}(\alpha_1 = \alpha_n) = 1 \\ & \textit{iff} & \mathcal{V}_{\mathcal{I}}^{\hat{c}}(\varphi) = 1. \end{array}$$

Whereas the starred biconditionals follow by the identities above, the other biconditionals are immediate from case assumptions and semantics for $\mathcal{L}^=$. Thus the lemma holds for any wff φ of $\mathcal{L}^=$ where $\texttt{Comp}(\varphi) = 0$.

Induction Step: Assume for induction that the lemma holds for every wff φ of $\mathcal{L}^=$ where $\texttt{Comp}(\varphi) \leqslant n$. Letting φ be a wff of $\mathcal{L}^=$ where $\texttt{Comp}(\varphi) = n + 1$ and both $\hat{a}$ and $\hat{c}$ are v.a.s defined over $\mathbb{D}$, we may assume that $\hat{a}(\alpha) = \hat{c}(\alpha)$ for all free variables α in φ. It remains to show that $\mathcal{V}_{\mathcal{I}}^{\hat{a}}(\varphi) = \mathcal{V}_{\mathcal{I}}^{\hat{c}}(\varphi)$.

Case 1: Assume $\varphi = \neg\psi$. Since $\texttt{Comp}(\neg\psi) = \texttt{Comp}(\psi) + 1 = n + 1$ by assumption and the definition of complexity, we know that $\texttt{Comp}(\psi) = n$, and so $\mathcal{V}_{\mathcal{I}}^{\hat{a}}(\psi) = \mathcal{V}_{\mathcal{I}}^{\hat{c}}(\psi)$ by hypothesis. We may then observe the following biconditionals:

$$\begin{array}{rll} \mathcal{V}_{\mathcal{I}}^{\hat{a}}(\varphi) = 1 & \textit{iff} & \mathcal{V}_{\mathcal{I}}^{\hat{a}}(\neg\psi) = 1 \\ & \textit{iff} & \mathcal{V}_{\mathcal{I}}^{\hat{a}}(\psi) = 0 \\ (\star) & \textit{iff} & \mathcal{V}_{\mathcal{I}}^{\hat{c}}(\psi) = 0 \\ & \textit{iff} & \mathcal{V}_{\mathcal{I}}^{\hat{c}}(\neg\psi) = 1 \\ & \textit{iff} & \mathcal{V}_{\mathcal{I}}^{\hat{c}}(\varphi) = 1. \end{array}$$

Whereas the starred biconditional follows from the induction hypothesis, the other biconditionals hold by the semantics and case assumption.

Case 2: Assume $\varphi = \psi \wedge \chi$. Since $\texttt{Comp}(\psi \wedge \chi) = \texttt{Comp}(\psi) + \texttt{Comp}(\chi) + 1 = n+1$ by assumption and the definition of complexity, we know that $\texttt{Comp}(\psi), \texttt{Comp}(\chi) \leqslant n$, and so $\mathcal{V}_{\mathcal{I}}^{\hat{a}}(\psi) = \mathcal{V}_{\mathcal{I}}^{\hat{c}}(\psi)$ and $\mathcal{V}_{\mathcal{I}}^{\hat{a}}(\chi) = \mathcal{V}_{\mathcal{I}}^{\hat{c}}(\chi)$ by hypothesis. Now consider:

$$\begin{array}{rll} \mathcal{V}_{\mathcal{I}}^{\hat{a}}(\varphi) = 1 & \textit{iff} & \mathcal{V}_{\mathcal{I}}^{\hat{a}}(\psi \wedge \chi) = 1 \\ & \textit{iff} & \mathcal{V}_{\mathcal{I}}^{\hat{a}}(\psi) = \mathcal{V}_{\mathcal{I}}^{\hat{a}}(\chi) = 1 \\ (\star) & \textit{iff} & \mathcal{V}_{\mathcal{I}}^{\hat{c}}(\psi) = \mathcal{V}_{\mathcal{I}}^{\hat{c}}(\chi) = 1 \\ & \textit{iff} & \mathcal{V}_{\mathcal{I}}^{\hat{c}}(\psi \wedge \chi) = 1 \\ & \textit{iff} & \mathcal{V}_{\mathcal{I}}^{\hat{c}}(\varphi) = 1. \end{array}$$

Whereas the starred biconditional follows from the induction hypothesis, the other biconditionals hold by the semantics and case assumption.

Case 3: Assume $\varphi = \psi \vee \chi$. (Exercise for the reader.)

Case 4: Assume $\varphi = \psi \rightarrow \chi$. (Exercise for the reader.)

Case 5: Assume $\varphi = \psi \leftrightarrow \chi$. (Exercise for the reader.)

Case 6: Assume $\varphi = \forall\gamma\psi$. Since $\texttt{Comp}(\forall\gamma\psi) = \texttt{Comp}(\psi)+1 = n+1$ by assumption and the definition of complexity, we know that $\texttt{Comp}(\psi) = n$, and so by hypothesis, $\mathcal{V}_{\mathcal{I}}^{\hat{e}}(\psi) = \mathcal{V}_{\mathcal{I}}^{\hat{g}}(\psi)$ for any v.a.s $\hat{e}$ and $\hat{g}$ defined over $\mathbb{D}$ where $\hat{e}(\alpha) = \hat{g}(\alpha)$ for all free variables α in ψ. We may then observe the following biconditionals:

$$\begin{aligned}
\mathcal{V}_{\mathcal{I}}^{\hat{a}}(\varphi) = 1 \quad & \textit{iff} \;\; \mathcal{V}_{\mathcal{I}}^{\hat{a}}(\forall\gamma\psi) = 1 \\
& \textit{iff} \;\; \mathcal{V}_{\mathcal{I}}^{\hat{e}}(\psi) = 1 \text{ for every } \gamma\text{-variant } \hat{e} \text{ of } \hat{a} \\
(\star) \quad & \textit{iff} \;\; \mathcal{V}_{\mathcal{I}}^{\hat{e}}(\psi) = 1 \text{ for every } \gamma\text{-variant } \hat{e} \text{ of } \hat{c} \\
& \textit{iff} \;\; \mathcal{V}_{\mathcal{I}}^{\hat{c}}(\forall\gamma\psi) = 1 \\
& \textit{iff} \;\; \mathcal{V}_{\mathcal{I}}^{\hat{c}}(\varphi) = 1.
\end{aligned}$$

Letting $\hat{e}$ be any γ-variant of $\hat{a}$, it follows that $\hat{e}(\beta) = \hat{a}(\beta)$ for every variable $\beta \neq \gamma$. By assumption, $\hat{a}(\alpha) = \hat{c}(\alpha)$ for all free variables α in φ. Since $\varphi = \forall\gamma\psi$, we know by the definition of free variables that γ is not free in φ, and so $\hat{e}(\beta) = \hat{c}(\beta)$ for every variable $\beta \neq \gamma$. By generalizing on $\hat{e}$, it follows that every γ-variant $\hat{e}$ of $\hat{a}$ is also a γ-variant of $\hat{c}$, where the converse holds by parity of reasoning. This establishes the stared biconditional. As before, the other biconditionals given above follow by the semantics and case assumption.

Case 7: Assume $\varphi = \exists\gamma\psi$. (Exercise for the reader.) □

Lemma 9.2 If φ is a wfs of $\mathcal{L}^{=}$: $\mathcal{V}_{\mathcal{I}}(\varphi) = 1$ *iff* $\mathcal{V}_{\mathcal{I}}^{\hat{a}}(\varphi) = 1$ for some v.a. $\hat{a}$ over $\mathbb{D}$.

Letting φ be a wfs and $\mathcal{M} = \langle \mathbb{D}, \mathcal{I} \rangle$ of $\mathcal{L}^{=}$, we aim to prove both directions of the biconditional: $\mathcal{V}_{\mathcal{I}}(\varphi) = 1$ just in case $\mathcal{V}_{\mathcal{I}}^{\hat{a}}(\varphi) = 1$ for some v.a. $\hat{a}$ over $\mathbb{D}$.

Assume $\mathcal{V}_{\mathcal{I}}(\varphi) = 1$. By definition, $\mathcal{V}_{\mathcal{I}}^{\hat{a}}(\varphi) = 1$ for every v.a. $\hat{a}$ defined over $\mathbb{D}$. Since $\mathbb{D} \neq \varnothing$, we know that there is some $d \in \mathbb{D}$. We may then consider the constant v.a. where $\hat{c}(\alpha) = d$ for every variable α of $\mathcal{L}^{=}$. Given the above, $\mathcal{V}_{\mathcal{I}}^{\hat{c}}(\varphi) = 1$, and so we may conclude that $\mathcal{V}_{\mathcal{I}}^{\hat{a}}(\varphi) = 1$ for some v.a. $\hat{a}$ defined over $\mathbb{D}$.

Assume instead that $\mathcal{V}_{\mathcal{I}}^{\hat{a}}(\varphi) = 1$ for some v.a. $\hat{a}$ defined over $\mathbb{D}$. Let $\hat{c}$ be any v.a. defined over $\mathbb{D}$. Since φ has no free variables, $\hat{a}(\alpha) = \hat{c}(\alpha)$ holds vacuously for all free variables α in φ, and so $\mathcal{V}_{\mathcal{I}}^{\hat{a}}(\varphi) = \mathcal{V}_{\mathcal{I}}^{\hat{c}}(\varphi)$ follows by **Lemma 9.1**. Since $\hat{c}$ was arbitrary, $\mathcal{V}_{\mathcal{I}}^{\hat{c}}(\varphi) = 1$ for all v.a. $\hat{c}$ over $\mathbb{D}$, and so $\mathcal{V}_{\mathcal{I}}(\varphi) = 1$. □

Given **Lemma 9.1**, the lemma given above follows easily. As we will see, **Lemma 9.2** will never be entirely necessary, but often convenient. After all, we made do just fine without **Lemma 9.2** until now. Nevertheless, it will help to streamline our semantic proofs to be able to appeal to these lemmas when convenient since it is often easier to work with a particular v.a. rather than a claim about all v.a.s defined over a given domain.

9.5 Uniqueness

Recall the following argument from before:

J1. Only Cara loves Pedro.	K1. $\forall x(Lxp \leftrightarrow x = c)$
J2. DJ Faro loves Pedro.	K2. Ldp
J3. DJ Faro is Cara.	K3. $d = c$

We are now in a position to show that this argument is valid.

Proof: Let $\mathcal{M} = \langle \mathbb{D}, \mathcal{I} \rangle$ be a $\mathcal{L}^{=}$ model where: (1) $\mathcal{V}_{\mathcal{I}}(\forall x(Lxp \leftrightarrow x = c)) = 1$; and (2) $\mathcal{V}_{\mathcal{I}}(Ldp) = 1$. It follows from the latter assumption by **Lemma 9.2** that $\mathcal{V}_{\mathcal{I}}^{\hat{c}}(Ldp) = 1$ for some $\hat{c}$ defined over $\mathbb{D}$, and so $\langle v_{\mathcal{I}}^{\hat{c}}(d), v_{\mathcal{I}}^{\hat{c}}(p) \rangle \in \mathcal{I}(L)$. Since d and p are constants, we know by definition that $\langle \mathcal{I}(d), \mathcal{I}(p) \rangle \in \mathcal{I}(L)$.

Let $\hat{e}$ be a v.a. defined over $\mathbb{D}$ where $\hat{e}(x) = \mathcal{I}(d)$. Since $\mathcal{V}_{\mathcal{I}}^{\hat{a}}(\forall x(Lxp \leftrightarrow x = c)) = 1$ for every $\hat{a}$ defined over $\mathbb{D}$ by (1), we know that $\mathcal{V}_{\mathcal{I}}^{\hat{e}}(\forall x(Lxp \leftrightarrow x = c)) = 1$ in particular. By the semantics, $\mathcal{V}_{\mathcal{I}}^{\hat{e}}(Lxp) = \mathcal{V}_{\mathcal{I}}^{\hat{e}}(x = c)$. Thus $\langle v_{\mathcal{I}}^{\hat{e}}(x), v_{\mathcal{I}}^{\hat{e}}(p) \rangle \in \mathcal{I}(L)$ just in case $v_{\mathcal{I}}^{\hat{e}}(x) = v_{\mathcal{I}}^{\hat{e}}(c)$, and so $\langle \hat{e}(x), \mathcal{I}(p) \rangle \in \mathcal{I}(L)$ just in case $\hat{e}(x) = \mathcal{I}(c)$. Since $\hat{e}(x) = \mathcal{I}(d)$, we know that $\langle \mathcal{I}(d), \mathcal{I}(p) \rangle \in \mathcal{I}(L)$ just in case $\mathcal{I}(d) = \mathcal{I}(c)$.

Given the above, we may conclude that $\mathcal{I}(d) = \mathcal{I}(c)$. Thus $v_{\mathcal{I}}^{\hat{g}}(d) = v_{\mathcal{I}}^{\hat{g}}(c)$ where $\hat{g}$ is any v.a., and so $\mathcal{V}_{\mathcal{I}}^{\hat{g}}(d = c) = 1$. Since $d = c$ is a wfs, we may conclude that $\mathcal{V}_{\mathcal{I}}(d = c) = 1$ by **Lemma 9.2**. Hence $\forall x(Lxp \leftrightarrow x = c), Ldp \vDash d = c$. $\square$

The proof begins by considering an arbitrary model in which the premises are true, showing that the conclusion must also be true in that model. The idea is to extend the general claim given in K1 to the particular claim given in K2 by considering an assignment of the variable x to whatever the constant d happens to refer to in the model at hand. Accordingly, it makes sense to begin by unpacking what we know about K2 before making use of K1.

Once we have worked out that $\langle \mathcal{I}(d), \mathcal{I}(p) \rangle \in \mathcal{I}(L)$, we are now in a position to make a strategic choice. Although it follows that $\mathcal{V}_{\mathcal{I}}^{\hat{a}}(Lxp \leftrightarrow x = c) = 1$ for any v.a. $\hat{a}$ defined over the domain, we were careful to consider a v.a. $\hat{e}$ where $\hat{e}(x) = \mathcal{I}(d)$. This is akin to instantiating x by d, resulting in the sentence $Ldp \leftrightarrow d = c$ which, together with Ldp entails $d = c$. Instead of replacing 'x' with 'd', we chose the x-variant $\hat{e}$ where $\hat{e}(x) = \mathcal{I}(d)$. This sort of reasoning will reappear once we introduce a proof system for $\mathcal{L}^{=}$.

Given that only Cara loves Pedro, we may think of her as the unique-Pedro-lover. That is, not only is there something out there that loves Pedro, Cara is *the* Pedro-lover. Suppose we forget Cara's name, but remember this prominent fact about her. We might then ask: is the Pedro-lover the same as DJ Faro? Instead of using her name, we are using this distinguishing feature to refer to her. This is a common practice since we don't have names for everything in English, and even when we do, we don't always know everything name.

Whereas Cara's distinguishing feature was loving Pedro, in general we may appeal to any condition however complex so long as it succeeds in picking out exactly one individual. That is, if we know that there is some particular way that one thing happens to be where only that thing is that way, then we may use that way of being to pick out that particular thing. For instance, perhaps many people love Pedro, but Cara is the only DJ to love Pedro. We may express this with $\forall x((Dx \wedge Lxp) \leftrightarrow x = c)$. Replacing the constant 'c' with a variable as in $\forall x((Dx \wedge Lxp) \leftrightarrow x = y)$ returns an open sentence which corresponds to the condition of *being the only DJ to love Pedro*. This brings us to the topic of definite descriptions.

9.6 Definite Descriptions

In 1905, Bertrand Russell famously characterized *definite descriptions* in terms of identity. In the paradigm cases, definite descriptions use the definite article 'the'. Suppose one were to hear crying in the next room, saying 'The baby is hungry'. This is to claim that the *one and only baby* (in the vicinity) is hungry. Russell was motivated in part by the apparent fact that one can use this sort of language in a meaningful way even if one is wrong about whether there is anything fitting the description. If there is no baby— the crying is a recording — the statement is false, but it's still meaningful. For this reason, Russell was reluctant to suppose that we should understand 'the baby' as a name. Instead, the sentence can be understood to be an existentially quantified claim about a unique baby in the vicinity though there may not be a baby in the vicinity. According to Russell, saying 'The baby is hungry' is to say three things: there is a baby, there's no other baby than that one, and that baby is hungry. As brought out above, uniqueness can be expressed in $\mathcal{L}^{=}$ with the help of identity.

In order to contrast names with definite descriptions, consider the symbolization key and regimentations for 'Jonathan is hungry' and 'The baby is hungry':

Bx: x is a baby	N1.	Hj
Hx: x is hungry	N2.	$\exists x((Bx \wedge \forall y(By \rightarrow y = x)) \wedge Hx)$.
j: Jonathan	N3.	$\exists x(\forall y(By \leftrightarrow y = x) \wedge Hx)$.

Whereas 'Jonathan is hungry' is easilyt regimented by N1, according to Russell's theory of definite descriptions, 'The baby is hungry' has a much more complex logical form. Whereas the regimentation given in N2 says that there is some baby which is the only baby and is hungry, N3 collapses the first two parts of N2, claiming that the unique baby is hungry.

As mentioned in the previous section, the distinguishing feature by which an individual is uniquely identified need not be expressed by a single predicate. For instance, suppose there is a baby who is sleeping right in front of us, but we hear crying from the other room. One may then be a little more specific by saying 'The crying baby is hungry'. Accordingly, we may expand our symbolization key to regiment this more specific claim.

Cx: x is crying

O1. $\exists x(\forall y((Cy \wedge By) \leftrightarrow y = x) \wedge Hx)$.

Instead of the single predicate 'B', we have used 'C' together with 'B' in order to form the open sentence '$Cy \wedge By$' which describes the unique individual to which we intend to refer.

In order to speak generally about the means by which we may refer to some unique individual satisfying a certain condition, let $\varphi(\alpha)$ be any wff of $\mathcal{L}^{=}$ in which the variable α is free. If α is the only free variable in $\varphi(\alpha)$, we may take $\varphi(\alpha)$ to be a DESCRIPTION. Moreover, $\varphi(\alpha)$ provides a DEFINITE DESCRIPTION in a model $\mathcal{M}$ just in case $\varphi(\alpha)$ is a description which just one thing satisfies, i.e., $\exists\beta\forall\alpha(\varphi(\alpha) \leftrightarrow \alpha = \beta)$ where $\alpha \neq \beta$ are distinct variables. Given a definite description $\varphi(\alpha)$, we may make claims about the object satisfying that description by conjoining another description $\psi(\beta)$ within the scope of the existential quantifier as follows:

$$\exists\beta(\forall\alpha(\varphi(\alpha) \leftrightarrow \alpha = \beta) \wedge \psi(\beta)).$$

This reads, the unique thing for which φ is such that ψ. The sentence O1 is an instance of this general recipe, and reads: the unique thing for which it is a crying baby, is hungry. Russell's idea is that this is what is going on when we use the definite article 'the' since we may say the same thing much more naturally with: *the* crying baby is hungry.

One of the interesting features of Russell's theory is that 'The baby is not hungry' is not the negation of 'The baby is hungry'. Instead, the negation applies only to the last conjunct:

P1. $\exists x(\forall y((Cy \wedge By) \leftrightarrow y = x) \wedge \neg Hx)$.

The reason Russell designed his theory this way was that he thought that both of these sentences equally implied that there is a baby. If there is no baby, then you'd be mistaken in asserting either 'The baby is hungry' or 'The baby is not hungry'. Consequently, one can't be the negation of the other, but rather requires the analysis given above.

As a treatment of the truth conditions of English sentences, Russell's theory is controversial. For instance, consider the following case:

Kxy: x is king of y
Bx: x is Bald
f: France

1. The king of France is bald.
2. $\exists x(\forall y(Kyf \leftrightarrow y = x) \wedge Bx)$.

Some philosophers of language think that sentences that seem to presuppose the existence of something that isn't there aren't straightforwardly false, but are rather defective in some other way— perhaps they fail to be meaningful, or perhaps they take on some truth value other than true or false. These matters are beyond the scope of this book, and so we will remain neutral on whether Russell's theory is an accurate treatment of English. Nevertheless, without including identity in the language, this question would not even arise. This helps to bring out what is distinctive about the expressive power of $\mathcal{L}^{=}$ in contrast to $\mathcal{L}^{\text{FOL}}$.

9.7 Quantities

Including identity in $\mathcal{L}^{=}$ permits us to express claims about quantities that we couldn't in $\mathcal{L}^{\text{FOL}}$. In §9.1 we considered the sentence 'Mozart composed at least two things' where identity was found to play a critical role. In particular, we provided the following regimentation:

M3. $\exists x \exists y((Cmx \wedge Cmy) \wedge x \neq y)$.

Given that we were able to regiment 'at least two', you might suspect that we can also regiment 'at least three', and so on for the other natural numbers. Consider the following:

R1. $\exists x \exists y \exists z(((((Cmx \wedge Cmy) \wedge Cmz) \wedge x \neq y) \wedge x \neq z) \wedge y \neq z)$.

Though it is a lot longer than sentence M3, the sentence above says that there are at least three things that Mozart composed. Given that conjunction is associative and commutative, all but the outermost parentheses are more trouble than they are worth. In general, we will indulge in the convention of dropping the parentheses that occur in long conjunctions and long disjunctions. Thus we may rewrite sentence R1, as follows:

S1. $\exists x \exists y \exists z(Cmx \wedge Cmy \wedge Cmz \wedge x \neq y \wedge x \neq z \wedge y \neq z)$.

This is a lot easier to read and nothing significant is lost. It is important to stress that we can only drop parentheses in sentences which *only* include conjunction, or *only* include disjunction. Even so, these sentences are bound to get very long for large values of n.

In order to characterize quantities in a more general way, it can be useful to introduce some abbreviations for what we will refer to as the *inequality quantifiers*. Instead of adding new primitive symbols to $\mathcal{L}^{=}$, we are only providing conventions for abbreviating long expression with much shorter expressions for the sake of readability.

In order to state these abbreviations in a general way, we will take β to be free for α in φ just in case there is no free occurrence of α in φ in the scope of a quantifier that binds β. For instance, y is not free for x in '$\forall yFxy$' since replacing 'x' with 'y' would yield '$\forall yFyy$' where the quantifier '$\forall y$' would end up binding an extra variable. Roughly speaking, you can take 'β is free for α' to mean 'β can replace α without leading to extra binding'.

We may then define $\varphi[\beta/\alpha]$ to be the substitution that results from replacing all free occurrences of α in φ with β where β is required to be free for α in φ. For instance, $\forall yFxy[z/x]$ is the wff $\forall yFzy$, and $\forall yFxy[y/x]$ is undefined since y is not free for x. Given this notation, we may define the following abbreviations for quantifiers of the form 'at least n things are such that φ':

$$\exists_{\geqslant 1}\alpha\varphi := \exists\alpha\varphi(\alpha)$$
$$\exists_{\geqslant n+1}\alpha\varphi := \exists\alpha(\varphi(\alpha) \wedge \exists_{\geqslant n}\beta(\alpha \neq \beta \wedge \varphi[\beta/\alpha])) \text{ where } \beta \text{ is free for } \alpha \text{ in } \varphi.$$

As above ':=' represents that the left side is merely an abbreviation for the right side. These abbreviations have a recursive structure which defines $\exists_{\geqslant n}\alpha$ for all n. Even in the base clause, it is important to require α to be free in φ where we have achieved this by writing $\varphi(\alpha)$. Otherwise, claims like $\exists_{\geqslant n}xHj$ would be read as 'at least n things are such that Jonathan is hungry' despite the fact that Hj does not include any free variables for the quantifiers to bind. We can put these quantifiers to work in order to construct sets of sentences like:

$$\Gamma_\infty := \{\exists_{\geqslant n}x(x = x) : n \in \mathbb{N}\}.$$

For any natural number n, the set Γ_∞ includes a sentence that says at least n things are self-identical. Not only may we show that there are models which satisfy Γ, these models must have infinite domains. That we can begin to express claims about the infinite further demonstrates just how much more expressive power $\mathcal{L}^=$ has than $\mathcal{L}^{\text{FOL}}$.

In addition to being able to say that there are at least n things that satisfy a certain condition, $\mathcal{L}^=$ permits us to say that that there are at most n things that satisfy a certain condition. Consider the following sentence and its regimentation:

T1. Mozart composed at most two things.
T2. $\forall x\forall y\forall z((Cmx \wedge Cmy \wedge Cmz) \to (x = y \vee x = z \vee y = z))$.

This says that for any x, y, and z which Mozart composed, at least two of them are identical. So far, nothing prevents all of them from being identical or requires there to be something which Mozart composed. More generally, we may say that there are at most n things which satisfy a given condition φ. Although we could define this recursively in a similar fashion to what was given above, we may avoid doing so by adopting the following convention for all n.

$$\exists_{\leqslant n}\alpha\varphi := \neg\exists_{\geqslant n+1}\alpha\varphi.$$

This says that it is not the case that there are at least $n + 1$ things that are φ, and so no more than n things that are φ. We may combine these two types of quantifiers to say that there are between n and m things that are φ as given by the following abbreviation:

$$\exists_{[n,m]}\alpha\varphi := \exists_{\geqslant n}\alpha\varphi \wedge \exists_{\leqslant m}\alpha\varphi.$$

Whereas $\exists_{[n,m]}\alpha\varphi$ says that between n and m things are φ, in the special case where $n = m$, the statement $\exists_{[n,n]}\alpha\varphi$ says that exactly n things are φ. We have already seen instances of this above with uniqueness. After all, saying that *only* Cara loves Pedro is like saying there is *exactly one* thing that loves Pedro where this entails both that there is *at least* one thing that loves Pedro and that there is *at most* one thing that loves Pedro, namely Cara.

Suppose that we want to say that there are exactly two things that Mozart composed. One way to do this is to conjoin sentences M3 and T2 since this amounts to saying that there is at least two things that Mozart composed and at most two things that Mozart composed. However, the result is long and difficult to parse. Alternatively, we could use the notation $\exists_{[2,2]}\alpha$ introduced above, though this notation abbreviates something just as complicated.

Instead of employing the quantifier $\exists_{[2,2]}\alpha$, we can simplify the regimentation as follows:

U1. Mozart composed exactly two things.
U2. $\exists x \exists y(x \neq y \wedge \forall z(Cmz \leftrightarrow (z = x \vee z = y)))$.

More generally, consider the following definitions:

$$\exists_0 \alpha\varphi := \forall\alpha\neg\varphi(\alpha)$$
$$\exists_{n+1}\alpha\varphi := \exists\alpha(\varphi(\alpha) \wedge \exists_n\beta(\alpha \neq \beta \wedge \varphi[\beta/\alpha])) \text{ where } \beta \text{ is not free in } \varphi$$

What it is for no α to be φ is for everything to not be φ. What it is for exactly $n + 1$ things to be φ is for something to be φ and exactly n other things to be φ. Given these recursive definitions, we may work out the following logical equivalences:

$$\exists_0\alpha\varphi \;=\!\!\!\!|\!\!\models\; \forall\alpha\neg\varphi(\alpha)$$
$$\exists_1\alpha\varphi \;=\!\!\!\!|\!\!\models\; \exists\alpha\forall\beta(\varphi[\beta/\alpha] \leftrightarrow \beta = \alpha)$$
$$\exists_2\alpha\varphi \;=\!\!\!\!|\!\!\models\; \exists\alpha\exists\beta(\alpha \neq \beta \wedge \forall\gamma(\varphi(\gamma/\alpha) \leftrightarrow (\gamma = \alpha \vee \gamma = \beta)))$$
$$\exists_3\alpha\varphi \;=\!\!\!\!|\!\!\models\; \exists\alpha\exists\beta\exists\gamma(\alpha \neq \beta \wedge \alpha \neq \gamma \wedge \beta \neq \gamma \wedge \forall\delta(\varphi(\delta/\alpha) \leftrightarrow (\delta = \alpha \vee \delta = \beta \vee \delta = \gamma)))$$
$$\vdots$$

Although the results differ in logical form from those that we may derive from $\exists_{[n,n]}\alpha$ for different values of n, they are logically equivalent insofar as $\exists_n\alpha\varphi \;=\!\!\!\!|\!\!\models\; \exists_{[n,n]}\alpha\varphi$.

Whereas $\exists_{\geqslant n}\alpha$ and $\exists_{\leqslant n}\alpha$ are referred to as INEQUALITY QUANTIFIERS, we will refer to $\exists_{[n,m]}\alpha$ and $\exists_n\alpha$ as CARDINALITY QUANTIFIERS. Although the inequality operators are often useful, the cardinality operators express something very specific, further demonstrating the expressive power of $\mathcal{L}^{=}$. After all, how would one attempt any of the definitions given above in $\mathcal{L}^{\text{FOL}}$?

9.8 Leibniz's Law

In §9.1, we saw that treating identity as any other predicate invalidated the argument:

H1. Hesperus is rising.	I1. Rh
H2. Hesperus is Phosphorus.	I2. Ihp
H3. Phosphorus is rising.	I3. Rp

The problem was that if the extension of the identity predicate can be assigned to any subset of $\mathbb{D}^2$, there was no guarantee that h and p refer to the same object. The problem was avoided by including identity among the primitive symbols of the language and providing its semantics as above. We may replace 'I' with '$=$' in the argument above where the result is valid. This is because if '$h = p$' is true in a model, then 'h' and 'p' refer to the very same object in the domain, and so 'Rh' and 'Rp' will have the same truth-value. More generally, we may show that all instances of the following schema are valid where α and β are constants and φ is a sentence, referring to this as a version of *Leibniz's Law*:

$$\alpha = \beta \vDash \varphi \leftrightarrow \varphi[\beta/\alpha].$$

The idea behind this schema is that if we are given two names α and β for the same object, then whatever we can say about that object with one name will have the same truth-value if we use the other name. This might seem natural in the case above. After all, if Hesperus is Phosphorus, how could it be that Hesperus is rising without Phosphorus rising?

However compelling this particular instance may be, Leibniz's Law admits of a wide range of exceptions. For instance, consider the following argument:

Bxy: x believes that y is rising.
t: Thales
h: Hesperus
p: Phosphorus

V1. Bth
V2. $h = p$
V3. Btp

Since the identity of Hesperus and Phosphorus was not always known, we might imagine a time when Thales believed that Hesperus is rising without also believing that Phosphorus is rising. As a result, the argument above may have true premises and a false conclusion. To take another case, we may imagine that Lois loves Clark Kent and Lois does not love Superman despite the fact that Clark Kent is Superman. Nevertheless, both of these arguments are valid given the semantics for $\mathcal{L}^=$. Has something gone wrong?

The constants α and β may be said to *co-refer* just in case they name the same thing, i.e., $\ulcorner\alpha = \beta\urcorner$ is true. One common response holds that some claims are *opaque* insofar as we cannot freely substitute co-referring constants. Belief claims are paradigmatic of opacity: just because one believes that φ doesn't mean that one must believe that $\varphi[\beta/\alpha]$ whenever $\ulcorner\alpha = \beta\urcorner$ is true. For instance, suppose that Kaya is learning arithmetic and believes that 2 is even. Nevertheless, she hasn't learned anything about prime numbers and so does not believe that the first prime number is even despite the fact that 2 is the first prime number.

Given that Leibniz's Law is valid given the semantics for $\mathcal{L}^=$, restricting Leibniz's Law requires significant revisions to the present semantic theory. For our purposes here, we will assume that co-referring terms can always be substituted for each other as asserted by *Leibniz's Law*. This amounts to the assumption that none of the sentences with which we will be concerned are opaque. This is a significant limitation since it is easy to introduce predicates like B given above for belief ascriptions. Nevertheless, overcoming this limitation is far from straightforward and lies outside the scope of our present concern.

Although $\mathcal{L}^=$ is a flexible and expressive powerful language, every language has its limits. Nevertheless, $\mathcal{L}^=$ is perfectly adequate for a wide range of applications. In particular, it is natural to assume that mathematics is *transparent* insofar as it excludes consideration of opaque claims in which *Leibniz's Law* fails to hold.

Chapter 10

Natural Deduction in $\mathcal{L}^=$

This chapter extends our system PL to provide a natural deduction system for $\mathcal{L}^=$ which we will refer to as FOL$^=$. As we will show in later chapters, FOL$^=$ is both sound and complete with respect to the semantics for $\mathcal{L}^=$. That means that there are natural deduction proofs corresponding to all and only the logical consequences in $\mathcal{L}^=$.

Our proof system FOL$^=$ will include the same introduction and elimination rules that we provided for the sentential operators included in PL, but now we will add introduction and elimination rules for the quantifiers and identity, deriving a number of new rules.

10.1 Substitution Instances

Consider the following claims:

A1. Kin loves everything.
A2. Kin loves Cassandra.
A3. Kin loves something.

Sentence A2 is referred to as an *instance* of both A1 and A3. Whereas A1 makes a universal claim about everything and A3 makes an existential claim about something, A2 makes a claim about two things in particular. It is important to distinguish between all three types of sentences, where one preliminary way to do this is to appeal to their logical strength, i.e., which sentences are a logical consequence which. Specifically, A2 is a logical consequence of A1 but not *vice versa*, and A3 is a logical consequence of A2 but not *vice versa*. Similarly, A3 is a logical consequence of A1 but not *vice versa*. It is worth nothing that certain existential claims may fail to be logical consequences of universal claims if we had permitted models to have an empty domain since given an empty domain. We will contemplate this shortly.

We may regiment the sentences above with the following symbolization key:

Lxy: x loves y
k: Kin
c: Cassandra

B1. $\forall x Lkx$.
B2. Lkc.
B3. $\exists x Lkx$.

Given that B2 is a logical consequence of B1 and that B3 is a logical consequence of B2, we should expect a complete logic for $\mathcal{L}^=$ to include rules by which to derive B3 from B2, and to derive B2 from B1. In order to state these rules in a sufficiently general way we will need to make use of the notion of substitution from before. In particular, recall the following definition from §9.7 where α and β are any singular terms of $\mathcal{L}^=$:

> β is FREE FOR α in φ just in case there is no free occurrence of α in φ in the scope of a quantifier that binds β.

Roughly speaking, β is free for α just in case β can replace α without resulting in any extra binding. Whereas z is free for x in $\forall y(Fxy \to Fyx)$ since replacing x with z would yield $\forall y(Fzy \to Fyz)$, the variable y is not free for x in $\forall y(Fxy \to Fyx)$ since replacing x with y yields $\forall y(Fyy \to Fyy)$ where the quantifier $\forall y$ ends up binding an extra variable.

If β is a constant, it follows that β is free for any α in any wff φ. Given this definition, we may define substitution as in §9.7, where α and β are any singular terms of the language.

> If β is free for α in φ, then the SUBSTITUTION $\varphi[\beta/\alpha]$ is the result of replacing all free occurrences of α in φ with β.

We may take $\varphi[\beta/\alpha]$ to read: β for α in φ. For instance, $\forall y(Fxy \to Fyx)[z/x]$ is the wff $\forall y(Fzy \to Fyz)$ and $\forall y(Fxy \to Fyx)[y/x]$ is undefined since y is not free for x. We may also observe that $\forall y(Fxy \to Fyx)[y/z]$ and $\forall y(Fxy \to Fyx)[z/y]$ are both $\forall y(Fxy \to Fxy)$.

This new vocabulary permits us to define the substitution instances of both universal and existential generalizations of the form $\forall\alpha\varphi$ and $\exists\alpha\varphi$. In particular, consider the following:

> $\varphi[\beta/\alpha]$ is a SUBSTITUTION INSTANCE of $\forall\alpha\varphi$ and $\exists\alpha\varphi$ if β is a constant.

Since β is a constant, the requirement that β is free for α in φ is satisfied. In particular, $(Fxy \to Fyx)[c/y] = (Fxc \to Fcx)$ is a substitution instance of $\forall y(Fxy \to Fyx)$ where this is the result of stripping off the quantifier binding y and replacing all free occurrences of y with c. Nothing forced us to choose the constant c. Instead, we could have used d, producing the instance $(Fxd \rightarrow Fdx)$ where we may then drop outermost parentheses as usual. In general, there will be a different instance for every constant in the language. For contrast, the instances of $\forall y(Fxy \rightarrow \exists yFyx)$ include such wffs as $Fxc \rightarrow \exists yFyx$ and $Fxd \rightarrow \exists yFyx$ where in both cases the second occurrence of y is not replaced since it is not free. It will often be convenient to refer to the constants here as INSTANTIATING CONSTANTS.

10.2 Universal Elimination

Recall $\forall xLkx$ from before. We claimed that Lkc is a logical consequence of $\forall xLkx$. We are now in a position to state this a little more precisely. To begin with, we may observe that Lkc is a substitution instance of $\forall xLkx$ with instantiating constant c. More generally, universally quantified claims entail all of their substitution instances. In order to maintain completeness, we may include the following elimination rule (∀E) so that the substitution instances of a universally quantified claim can be derived from that universally quantified claim:

m	$\forall\alpha\varphi$	
	$\varphi[\beta/\alpha]$	:m ∀E (β is a constant)

Remember that the notation for a substitution instance is not a part of $\mathcal{L}^=$, so you cannot write it directly in a proof. Instead, you write the substitution instance itself including whichever constant β is being used to replace the variable α as in this example:

1	$\forall x(Mx \rightarrow Rxd)$	:PR
2	$Ma \rightarrow Rad$	:1 ∀E
3	$Md \rightarrow Rdd$	:1 ∀E

You are permitted to write down any instance you like on a new line. In this example, we have used the rule twice to produce two instances. In the first case, we instantiate with the constant a and we instantiate with the constant d in the second case.

Before moving on, it is worth noting that this rule is both easy and natural. It is easy insofar as there are no extra conditions that have to be satisfied in order to apply the rule, and it is natural insofar as we reason this way all the time. These same virtues will not be shared by all of the quantifier rules that we will introduce, and so it is worth noting which rules are easier to remember and apply than others. In particular, universal introduction is a much trickier rule to state and apply. Before moving on to consider this case, we will continue with existential introduction which is another easy rule that is natural to apply.

10.3 Existential Introduction

Recall that Lkc was said to be a logical consequence of $\exists xLkx$. Given that Kin loves Cassandra, it follows that Kin loves someone. More generally, any existentially quantified claim is a logical consequence of its substitution instances. In order to preserve completeness, we may include the following existential introduction rule ($\exists$I) in our proof system:

m	$\varphi[\beta/\alpha]$	(β is a constant and α is a variable)
	$\exists\alpha\varphi$	:m $\exists$I

In the example above, $\varphi[\beta/\alpha] = Lkx[c/x] = Lkc$. Given $\exists$I, we may derive $\exists\alpha\varphi = \exists xLkx$ by existentially generalizing on the instantiating constant c. Alternatively, we could have existentially generalized on k, deriving $\exists xLxc$ given that $\varphi[\beta/\alpha] = Lxc[k/x] = Lkc$. Both $Lkx[c/x]$ and $Lxc[k/x]$ are identical to Lkc. Similarly, in addition to deriving $\exists xLkx$ and $\exists xLxc$ from Lkc, we can also use the variable y to derive $\exists yLky$ and $\exists yLyc$ since Lkc is also identical to $Lky[c/y]$ and $Lyc[k/y]$. Something similar may be said for other variables.

The example above only included one occurrence of each constant. By contrast, the sentence $Ma \to Rad$ has two occurrences of the constant a. As a result, we may existentially generalize on either occurrence of a by itself, both occurrences together, or neither. Consider:

1	$Ma \to Rad$	:PR
2	$\exists x(Mx \to Rad)$	:1 $\exists$I
3	$\exists x(Ma \to Rxd)$	:1 $\exists$I
4	$\exists x(Mx \to Rxd)$	:1 $\exists$I
5	$\exists x(Ma \to Rad)$	:1 $\exists$I
6	$\exists y\exists x(Mx \to Ryd)$	:2 $\exists$I
7	$\exists z\exists y\exists x(Mx \to Ryz)$	:6 $\exists$I

Whereas line 2 existentially generalizes on the first occurrence of the constant a, line 3 generalizes on the second occurrence, and line 4 generalizes on both. Focusing on line 2, $\varphi[\beta/\alpha] = (Mx \to Rad)[a/x] = (Ma \to Rad)$, where similar identities hold for the other lines. For line 4 we have $\varphi[\beta/\alpha] = (Mx \to Rxd)[a/x] = (Ma \to Rad)$, and for line 5 we have $\varphi[\beta/\alpha] = (Ma \to Rad)[a/x] = (Ma \to Rad)$. In lines 6 and 7, one existential generalization is staged after the next, generalizing on difference occurrences of the same constant.

Put roughly, existential generalization permits one to replace any number of occurrences of a constant with a variable that is then bound by an existential quantifier. Strictly speaking, this approximation is not correct since we cannot generalize as follows:

1	$\exists x(Mx \to Rad)$	:PR
2	$\exists x \exists x(Mx \to Rxd)$	:1 ∃I (incorrect)
3	$\exists y \exists x(Mx \to Ryd)$	:2 ∃I

Whereas line 3 is the same as line 6 above and perfectly correct, line 2 does not follow by existential generalization. The reason is that we cannot construct an appropriate $\varphi[\beta/\alpha]$. For instance, we cannot take $\varphi[\beta/\alpha] = \exists x(Mx \to Rxd)[a/x]$ given the scope of the quantifier since $\exists x(Mx \to Rxd)[a/x]$ is not identical to the premise $\exists x(Mx \to Rad)$. By contrast, $\varphi[\beta/\alpha] = \exists x(Mx \to Ryd)[a/y]$ is acceptable since $\exists x(Mx \to Ryd)[a/y] = \exists x(Mx \to Rad)$ which is identical to the premise. Thus we may derive $\exists y \exists x(Mx \to Ryd)$ on line 3 by ∃I.

Whereas replacing a with x results in a quantifier which is already present and so picks up an extra bound variable, this does not happen if we replace a with y instead. The existential generalization rule is stated in such a way so as to ensure that this extra binding cannot occur without breaking the rule. Nevertheless, it is important to be careful to accurately follow the rule for existentially generalizing. An easy way to see if you are following the rule correctly is to check to see if you pick up any extra variable binding.

10.4 Universal Introduction

The rules provided above allow us to derive B2 from B1 and to derive B3 from B2. As we noted above, the converse derivations do not hold. After all, just because Kin loves Cassandra (B2), it does not follow that Kin loves everything (B1). Similarly, just because Kin loves something (B3), it does not follow that Kin loves Cassandra (B2). More generally, universal claims are not logical consequences of their substitution instances, not do particular substitution instances of an existential claim follow as logical consequences. Nevertheless, it is possible derive universal claims from their instances in specific circumstances and it is possible to make use of certain instances of existential claims to derive further claims. The following two section will introduce these derivation rules, discussing the conditions of their application. Whereas universal elimination was easy and existential introduction was only slightly trickier, these cases are much more constrained.

One way to think about a universal claim of the form $\forall \alpha \varphi \alpha$ is as a long conjunction $\varphi[a/\alpha] \wedge \varphi[b/\alpha] \wedge \ldots$ where the constants $a, b, \ldots$ name every element in the domain. However, this way of thinking is limited since we may not have enough constants to name every element in the domain, and even if there were enough constants to go around, they could all be interpreted as referring to the same element of the domain. These considerations are what motivated the introduction of variable assignments which we used to provide the semantics for the quantifiers. In particular, $\mathcal{V}_{\mathcal{I}}^{\hat{a}}(\forall \alpha \varphi) = 1$ just in case $\mathcal{V}_{\mathcal{I}}^{\hat{c}}(\varphi) = 1$ for every α-variant $\hat{c}$ of $\hat{a}$, and $\mathcal{V}_{\mathcal{I}}^{\hat{a}}(\exists \alpha \varphi) = 1$ just in case $\mathcal{V}_{\mathcal{I}}^{\hat{c}}(\varphi) = 1$ for some α-variant $\hat{c}$ of $\hat{a}$. We may simulate something analogous in the proof system for $\mathcal{L}^=$ by way of arbitrary constants.

The following examples will help to illustrate what it is for a constant to be arbitrary:

C1. $\forall x Mx$.
C2. $\forall y My$.

These sentences entail each other for indeed they say the very same thing, i.e., that everything is M. Insofar as the logical equivalence of these two sentences are to be reflected in the proof system, we need a way to derive sentence C2 from sentence C1 and *vice versa*.

Consider the following derivation:

1	$\forall x Mx$	:PR
2	Ma	:1 $\forall$E
3	$\forall y My$	:2 $\forall$I

In line 2, we have derived Ma from $\forall x Mx$ by $\forall$E given above. We could have equally derived $M\alpha$ for any other constant α. This much should come as no surprise. What is more surprising is the derivation of $\forall y My$ in line 3 from Ma in line 2. After all, this is like the inference from 'Alan is mad' to 'Everyone is mad' which is easy to invalidate. You might wonder what kind of constraints would make this inference count as logically valid.

The key idea behind the inference from line 2 to line 3 is that a is an *arbitrary* constant. In particular, a does not occur in the premise, nor does a occur in line 3 which we are deriving, and so there are no constraints on what a might name. Rather, the constant a could name anything whatsoever. You can think of a as analogous to $\hat{c}(x)$ for an arbitrary x-variant $\hat{c}$ by which we generalized over the elements in the domain to which x may refer. Given that a could refer to anything, we may derive line 3. Although we used the variable y in line 3, we could have used x, or indeed any other variable since there is no threat of any extra variable binding taking place. More generally, the universal introduction rule ($\forall$I) is stated as follows:

m	$\varphi[\beta/\alpha]$	(β is a constant and α is a variable)
n	$\forall \alpha \varphi$	:m $\forall$I (β does not occur in n or a premise or undischarged assumption)

It is easy to see that the inference from line 2 to line 3 in the proof given above follows this rule. For contrast, consider the following incorrect application:

1	$\forall x Rax$	:PR
2	Raa	:1 $\forall$E
3	$\forall y Ryy$	:2 $\forall$I (incorrect)

Intuitively, this argument would be like inferring the conclusion that everything respects itself from the premise that Arthur respects everything. Even if it follows from the premise that Arthur respects himself, it does not follow that everything respects itself. The reason the derivation goes astray is that a is not an arbitrary constant since it occurs in the premise which is an undischarged assumption. In particular, we might take a to name Arthur. Even if a named something else, it's interpretation would still be constrained in any interpretation in which the premise is true. As a result, the constant a is not arbitrary in way that we need it to be if we are to universally generalize on a with the rule ∀I.

We have just seen one way that ∀I can go wrong by offending the constraint that the instantiating constant not occur in a premise, where something similar may be said for undischarged assumptions. What about the other constraint? Consider the following:

1	$\forall x Rxx$	:PR
2	Raa	:1 ∀E
3	$\forall x Rax$	:2 ∀I (incorrect)

This argument infers that Arthur respects everything from the premise that everything respects itself. Letting $\varphi[\beta/\alpha] = Rax[a/x] = Raa$, we may observe that $\forall x\varphi = \forall x Rax$, and so one might have thought that we could derive $\forall x Rax$ from Raa. This would have been correct but for the provision that β not occur in $\forall\alpha\varphi$. In the case above, this requires that a not occur in $\forall x Rax$. Since a does occur in $\forall x Rax$, line 3 does not follow from line 2.

A good way to remember the restrictions on the ∀I rule is to remember that these restriction are put in place to require the instantiating constant to be arbitrary. In particular, we want the instantiating constant that we are generalizing on to be anything, and so it cannot occur in a premise, undischarged assumptions, or in the sentence that we are deriving. The incorrect inferences above are paradigmatic of cases in which these requirements are not satisfied.

Although the constant α is not permitted to occur in any undischarged assumption, it may occur in an assumption of a subproof that we have already closed. For example, here is a perfectly respectable proof of $\forall z(Dz \to Dz)$ from no premises.

1	$\quad Df$	:AS
2	$\quad Df \vee Ff$	:1 ∨I
3	$Df \to (Df \vee Ff)$	:1–2 →I
4	$\forall z(Dz \to (Dz \vee Fz))$	:3 ∀I

Reasoning of the kind given above is as common as it is important: if we can show that an arbitrary differentiable function is either differentiable or smooth, it follows that any differentiable function is either differentiable or smooth. Since the assumption given in line 1 above has been discharged, the instantiating constant f does not occur in any undischarged assumptions, and so line 4 follows from line 3 by ∀I without offending any restrictions.

10.5 Existential Elimination

A sentence with an existential quantifier tells us that *something* satisfies a given condition. For example, letting 'S' regiment 'is sad' and 'M' regiment 'is mad', the existential claims $\exists xSx$ and $\exists yMy$ say that something is sad and that something is mad but they don't tell us *what* is sad or mad. In particular, we cannot conclude that the same thing is sad and mad. This worry can be brought out with the following incorrect derivation:

1	$\exists xSx$	:PR
2	$\exists yMy$	:PR
3	Sa	:1 ∃I (INCORRECT)
4	Ma	:2 ∃I (INCORRECT)
5	$Sa \wedge Ma$	:4, 3 ∧I
6	$\exists z(Sz \wedge Mz)$	:5 ∃I

Whereas a is a perfectly arbitrary constant when it is introduced in line 3, reusing a in line 4 leads to the conclusion that the same thing is both sad and mad. Were we permitted to introduce lines 3 and 4 in this way, nothing would stop us from conjoining these claims and existentially generalizing, concluding that something is both sad and mad even though this might not be the case. For instance, one thing could be sad and something else could be mad where nothing requires these to be the same thing. Somehow we must prevent these two existential claims from running together as they do above.

Although it is important to avoid concluding too much, this does not mean that we cannot draw any inferences at all from an existential claim. For instance, suppose that we knew that $\exists xSx$ and $\forall x(Sx \rightarrow Mx)$, or in English, something is sad and, moreover, everything that is sad is also mad. It is natural to reason in the following way:

> `Bob:` By assumption, something is sad. Although we do not know what that thing is, call it 'Bob' for the purposes of this argument. We also know by assumption that everything that is sad is mad. In particular, if Bob is sad, then Bob is mad. Thus Bob is mad. Since 'Bob' was a name we introduced only for the sake of the argument, we may revert to the more general claim that something is mad.

Especially for long lines of reasoning, it is often useful to introduce a temporary constant in the manner employed above. In particular, we want the constant to be arbitrary, and so it cannot occur in any undischarged assumptions. At the same time, we also need to keep track of this constant, being careful not to make any other assumptions about the object to which this temporary constant refers, or to draw any conclusions which include this constant. The existential elimination rule (∃E) will achieve all of this by way of a subproof:

m	$\exists\alpha\varphi$	
n	$\quad$ $\varphi[\beta/\alpha]$	:AS (where β is a constant and does not occur in m, p, or any undischarged assumptions)
	$\quad$ $\vdots$	
p	$\quad$ ψ	
	ψ	:m, n–p $\exists$E

Note that existential elimination cites both the existential claim that is being instantiated (line m) as well as the subproof (lines n–p) which follows from its instantiation. The restriction on what β can be is what prevents $\exists$E from producing invalid arguments as brought out above. One easy way to satisfy these restrictions is to always choose a new instantiating constant β that does not appear anywhere outside the subproof. This practice is in keeping with the idea that β is an arbitrary place holder that we make temporary use of inside a subproof in order to expedite the reasoning we wish to develop.

Like the rules for conditional introduction and both negation introduction and elimination, the rule for existential elimination involves discharging an assumption. The pattern is to: (1) assume a substitution instance with an arbitrary constant; (2) reason your way to a conclusion that does not include that constant; and (3) discharge the assumption, closing the subproof. As before, the lines of a closed subproof are DEAD thereafter, where every line of a proof that is not dead at a line in a proof is LIVE at that line. The rules that cite individual lines (as opposed to subproofs) can only appeal to lines that are live at that line.

Having introduced the existential elimination rule, we are now in a position to regiment the informal argument given above in which 'Bob' served as our arbitrary constant.

1	$\exists xSx$	:PR
2	$\forall x(Sx \rightarrow Mx)$	:PR
3	$\quad$ Sa	:AS for $\exists$E
4	$\quad$ $Sa \rightarrow Ma$	:2 $\forall$E
5	$\quad$ Ma	:3, 4 $\rightarrow$E
6	$\quad$ $\exists xMx$	:5 $\exists$I
7	$\exists xTx$	:1, 3–6 $\exists$E

Given the introduction and elimination rules for both the existential and universal quantifiers, we may turn to derive the quantifier exchange rules which will often be useful. In addition to being extremely intuitive, these derivations will provide a good test that the rules that the quantifier introduction and elimination rules are in good order, allowing us to derive the logical consequences that we should expect of a complete proof system for $\mathcal{L}^{=}$.

10.6 Quantifier Exchange Rules

Sentences of the form $\neg\exists\alpha\varphi$ and $\forall\alpha\neg\varphi$ are logically equivalent, as are sentences of the form $\neg\forall\alpha\varphi$ and $\exists\alpha\neg\varphi$. Accordingly, we may assert the following entailments:

> Quantifier Exchange:
>
> $(\neg\exists)$ $\neg\exists\alpha\varphi \vDash \forall\alpha\neg\varphi$. $(\forall\neg)$ $\forall\alpha\neg\varphi \vDash \neg\exists\alpha\varphi$.
>
> $(\neg\forall)$ $\neg\forall\alpha\varphi \vDash \exists\alpha\neg\varphi$. $(\exists\neg)$ $\exists\alpha\neg\varphi \vDash \neg\forall\alpha\varphi$.

Given the introduction and elimination rules for the quantifiers, we may derive rules which correspond to these entailments. Consider the derivation of $(\neg\exists)$ given below:

1	$\mid\ \neg\exists\alpha\varphi$	:PR
2	$\mid\ \mid\ \varphi[\beta/\alpha]$	:AS for $\neg$I where β is a new constant
3	$\mid\ \mid\ \exists\alpha\varphi$	:2 $\exists$I
4	$\mid\ \mid\ \neg\exists\alpha\varphi$	:1 R
5	$\mid\ \neg\varphi[\beta/\alpha]$	:2–4 $\neg$I
6	$\mid\ \forall\alpha\neg\varphi$	:5 $\forall$I

This proof works by instantiating φ with a new constant β in line 2 and then existentially generalizing on β in order to derive a contradiction. As a result, $\neg\varphi[\beta/\alpha]$ follows by negation introduction. Since β does not occur in any undischarged assumptions, we may universally generalize on β in order to derive the conclusion in line 6. Consider the derivation of $(\forall\neg)$:

1	$\mid\ \forall\alpha\neg\varphi$	:PR
2	$\mid\ \mid\ \exists\alpha\varphi$	:AS for $\neg$I
3	$\mid\ \mid\ \mid\ \varphi[\beta/\alpha]$	:AS for $\exists$E where β is a new constant
4	$\mid\ \mid\ \mid\ \mid\ \exists\alpha\varphi$	:AS for $\neg$I
5	$\mid\ \mid\ \mid\ \mid\ \varphi[\beta/\alpha]$	:3 R
6	$\mid\ \mid\ \mid\ \mid\ \neg\varphi[\beta/\alpha]$	:1 $\forall$E
7	$\mid\ \mid\ \mid\ \neg\exists\alpha\varphi$	:4–6 $\neg$I
8	$\mid\ \mid\ \neg\exists\alpha\varphi$	:2, 3–7 $\exists$E
9	$\mid\ \neg\exists\alpha\varphi$	:2–8 $\neg$I

The proof begins as expected by stating the premise and assuming line 2 for negation introduction. We then assume an instance of line 2 in line 3. Although we could instantiate line 1 on line 4 in order to produce a contradiction with line 3, this is not what we need. Instead, we are looking to derive something which will contradict with line 2 that does not include β. Accordingly, we assume line 4 for negation introduction. We may now rewrite line 3 in line 5 and instantiate line 1 on line 6, producing a contradiction. By negation introduction on line 7, we have what we wanted, and since β does not occur in line 7, we may close the subproof with existential elimination on line 8. This produces another contradiction, and so we may conclude in line 9 with negation introduction.

Although we could go on to prove $(\neg\forall)$ and $(\exists\neg)$ in a similar fashion, there is another approach which turns on the following METARULE which takes one derivation to follow from another instead of taking one sentence to follow from some others.

(MCP) If $\varphi \vdash \psi$, then $\neg\psi \vdash \neg\varphi$.

Given any φ and ψ, this rule says that if we can derive ψ from φ, then we may also derive $\neg\varphi$ from $\neg\psi$, referring to this rule as *meta-contraposition*. It is worth comparing our standard contraposition rule CP which we derived before: $\varphi \rightarrow \psi \vdash \neg\psi \rightarrow \neg\varphi$. Whereas CP concerns the material conditional $\rightarrow$, MCP concerns $\vdash$. Nevertheless, the proof is similar where we begin by assuming $\varphi \vdash \psi$ for some otherwise arbitrary wfss φ and ψ of $\mathcal{L}^{=}$, calling this assumption $(\star)$. We may then write the following proof:

1	$\neg\psi$	:PR
2	$\quad \mid\ \varphi$	:AS for $\neg$I
3	$\quad \mid\ \psi$	:2 $(\star)$
4	$\quad \mid\ \neg\psi$	:1 R
5	$\neg\varphi$	:2–4 $\neg$I

Since we know that $\varphi \vdash \psi$ by assumption, there is some proof of ψ from φ. Accordingly, we may think of the proof given above as abbreviating a longer proof in which $(\star)$ is replaced with the proof of ψ from φ. The result is a complete proof of $\neg\varphi$ from $\neg\psi$.

There are two more derived rules $(\forall\text{DN})$ and $(\text{DN}\forall)$ which will come in handy:

1	$\forall\alpha\neg\neg\varphi$	:PR
2	$\neg\neg\varphi[\beta/\alpha]$	:1 $\forall$E
3	$\varphi[\beta/\alpha]$	:2 DN
4	$\forall\alpha\varphi$	:3 $\forall$I

1	$\exists\alpha\neg\neg\varphi$	:PR
2	$\quad \mid\ \neg\neg\varphi[\beta/\alpha]$	:AS
3	$\quad \mid\ \varphi[\beta/\alpha]$	:2 DN
4	$\quad \mid\ \exists\alpha\varphi$	:3 $\exists$I
5	$\exists\alpha\varphi$	:1, 2–4 $\exists$E

Without too much trouble, we may reverse the direction of the proofs above to derive: $\forall\alpha\varphi \vdash \forall\alpha\neg\neg\varphi$ and $\exists\alpha\varphi \vdash \exists\alpha\neg\neg\varphi$. We now have all the ingredients that we need to provide proofs which correspond to $(\neg\forall)$ and $(\exists\neg)$ from before. Since these proofs will make essential use of MCP, these proofs will take place in the metalanguage and be presented informally.

Lemma 10.1 $(\neg\forall)$ $\quad \neg\forall\alpha\varphi \vdash \exists\alpha\neg\varphi$.

Proof: Consider the result of replacing φ with $\neg\varphi$ in the derivation of $(\neg\exists)$ given above, where this constitutes a proof of $\neg\exists\alpha\neg\varphi \vdash \forall\alpha\neg\neg\varphi$. Since $\forall\alpha\neg\neg\varphi \vdash \forall\alpha\varphi$ by $\forall$DN, we may chain these proofs together to derive $\neg\exists\alpha\neg\varphi \vdash \forall\alpha\varphi$. If follows by MCP that $\neg\forall\alpha\varphi \vdash \neg\neg\exists\alpha\neg\varphi$, and so $\neg\forall\alpha\varphi \vdash \exists\alpha\neg\varphi$ by DN. □

Lemma 10.2 $(\exists\neg)$ $\quad \exists\alpha\neg\varphi \vdash \neg\forall\alpha\varphi$.

Proof: (Exercise for the reader.) □

You can think of these proofs as instruction manuals for constructing the appropriate derivations. In this case, it might have been easier to provide derivations for $(\neg\forall)$ and $(\exists\neg)$ in the same manner that we did for $(\neg\exists)$ and $(\forall\neg)$. Even so, the metarule MCP and derived rules $(\forall$DN$)$ and $(\exists$DN$)$ are important in their own right. It is important to note, however, that whereas $(\forall$DN$)$ and $(\exists$DN$)$ may be applied in the course of a derivation, MCP cannot. This is because MCP draws the conclusion $\neg\psi \vdash \neg\varphi$ from $\varphi \vdash \psi$, where the instances of these claims are not sentences in $\mathcal{L}^{=}$ since they include '$\vdash$'. It was for this reason that we provided informal proofs of $(\neg\forall)$ and $(\exists\neg)$ in the metalanguage.

10.7 Identity

We turn now to the introduction and elimination rules for identity, both of which are a lot easier to work with than the quantifier rules. To begin with, the introduction rule for identity is in keeping with the idea that everything is identical to itself. Accordingly, for any constant α, one may always write $\alpha = \alpha$ on any line of a proof. Moreover, instead of *assuming* $\alpha = \alpha$ for some particular constant α where this would constitute a further undischarged assumption, we may justify any instance of $\alpha = \alpha$ by citing the following rule:

$$\big|\ \alpha = \alpha \qquad :=\text{I} \quad (\alpha \text{ is a constant})$$

The =I rule is unlike the other rules in that it does appeal to any prior lines of the proof and so may be referred to as an AXIOM SCHEMA, meaning that for any constant α, we may add $\alpha = \alpha$ to a proof without any justification apart from citing =I itself.

The elimination rule for identity is more complicated, but only slightly. If you have derived φ and $\alpha = \beta$ for some constants α and β, then we may replace any occurrence of α with β in φ and similarly, we may replace any occurrence of β with α in φ. More specifically, the identity elimination rules (=E) may be stated as follows:

m	$\alpha = \beta$		m	$\alpha = \beta$	
n	$\varphi[\alpha/\gamma]$		n	$\varphi[\beta/\gamma]$	
	$\varphi[\beta/\gamma]$	:m, n =E		$\varphi[\alpha/\gamma]$	:m, n =E

It is important to note that not all occurrences of α (similarly β) need to be replaced with β (α) in any given φ. The identity elimination rules capture this by including $\varphi[\alpha/\gamma]$ and $\varphi[\beta/\gamma]$. For example, consider the following derivations:

1	Raa	
2	$a = b$	
3	Rab	:1, 2 =E
4	Rba	:1, 2 =E
5	Rbb	:1, 2 =E

Given the premises Raa and $a = b$, we may derive Rab, Rba, and Rbb. This is because $Raa = Rac[a/c] = Rca[a/c] = Rcc[a/c]$, leading to the replacements $Rac[b/c] = Rab$, $Rca[b/c] = Rba$, and $Rcc[b/c] = Rbb$, respectively. Although the justifications for each inference is the same, the inferences are licensed on account of different values of φ.

Here is a derivation of the transitivity law for identity:

1	$\quad a{=}b \wedge b{=}c$	:PR
2	$\quad a{=}b$	:1 $\wedge$E
3	$\quad b{=}c$	:1 $\wedge$E
4	$\quad a{=}c$	:2, 3 =E
5	$(a = b \wedge b = c) \rightarrow a = c$	:1–4 $\rightarrow$I
6	$\forall z((a = b \wedge b = z) \rightarrow a = z)$	:5 $\forall$I
7	$\forall y \forall z((a = y \wedge y = z) \rightarrow a = z)$	:6 $\forall$I
8	$\forall x \forall y \forall z((x = y \wedge y = z) \rightarrow x = z)$	:7 $\forall$I

At line 4, we employed the identity on line 3 in order to replace b in line 2 with a c. After discharging the assumption with the conditional introduction rule, the rest of the proof proceeded by universal introduction given that a, b, and c do not occur in any undischarged assumptions. A similar proof shows that identity is symmetric.

As a somewhat special case, it is worth considering a proof that identity is reflexive.

1	$a = a$	=I
2	$\forall x(x = x)$	:1 ∀I

Notice that there is no horizontal line below line 1. This signifies that there is no premise. Rather, identity introduction has been used in its unique capacity to introduce an identity claim which does not appeal to any other lines. Accordingly, line 1 is neither a premise nor an undischarged assumption. Since the constant a does not occur in any premises or undischarged assumptions, we may use universal introduction to conclude the proof.

10.8 Proofs and Provability in $\mathcal{L}^=$

We will refer to the natural deduction rules specified above together with the natural deduction rules provided in Chapter 3 as the natural deduction rules for FOL$^=$. We may then adapt the definition of a proof from before by replacing PL with FOL$^=$ as follows:

> A derivation (or proof) of φ from Γ in FOL$^=$ is any finite sequence of wfss of $\mathcal{L}^=$ ending in φ where every wfs in the sequence is either: (1) a premise in Γ; (2) an assumption which is eventually discharged; or (3) follows from previous lines by a natural deduction rule for FOL$^=$ besides AS.

A wfs φ of $\mathcal{L}^=$ is derivable (or provable) from Γ in FOL$^=$ just in case there is a natural deduction derivation (proof) of φ from Γ in FOL$^=$, where we may write this $\Gamma \vdash_{\text{FOL}^=} \varphi$, or just $\Gamma \vdash \varphi$ when it is clear from context which proof system we intend.

In keeping with the definitions given before, a wfs φ of $\mathcal{L}^=$ is a theorem of FOL$^=$ just in case $\vdash \varphi$ which abbreviates $\varnothing \vdash \varphi$, indicating that φ is derivable in FOL$^=$ from no premises. Since FOL$^=$ includes all the same rules and expressive resources of PL, all of the theorems of PL are also theorems of FOL$^=$. More generally, $\Gamma \vdash_{\text{FOL}^=} \varphi$ whenever $\Gamma \vdash_{\text{PL}} \varphi$. Put otherwise, FOL$^=$ is an extension of PL. As brought out in Chapters 11 - 12, we will similarly extend the soundness and completeness results for PL to cover FOL$^=$.

Two sentences φ and ψ are interderivable (or provably equivalent) in FOL$^=$ if and only if both $\varphi \vdash \psi$ and $\psi \vdash \varphi$. A set of sentences Γ is inconsistent if and only if $\Gamma \vdash \bot$ where $\bot$ is our arbitrarily chosen contradiction, e.g., $A \wedge \neg A$ from before.

10.9 Soundness of FOL$^{=}$

Recall the challenge of showing that a wfs φ of $\mathcal{L}^{\text{PL}}$ is not derivable in PL from a set of wfss Γ of $\mathcal{L}^{\text{PL}}$. Whereas providing a derivation of φ from Γ in PL is all that is required to show that φ is derivable from Γ, it is not possible to survey the space of all PL derivations to show that there is no derivation when none exists. The same predicament arises in attempting to show that φ cannot be derived from Γ in FOL$^{=}$. As before, we will overcome this difficulty by appealing to the following metalogical property:

FOL$^{=}$ SOUNDNESS: $\Gamma \vdash_{\text{FOL}^{=}} \varphi$ only if $\Gamma \vDash \varphi$.

In order to show that $\Gamma \nvdash_{\text{FOL}^{=}} \varphi$, we may show that $\Gamma \nvDash \varphi$ by providing a model in which $\mathcal{V}_{\mathcal{I}}(\gamma) = 1$ for all $\gamma \in \Gamma$ but $\mathcal{V}_{\mathcal{I}}(\varphi) = 0$. Given FOL$^{=}$ SOUNDNESS, it follows that $\Gamma \nvdash_{\text{FOL}^{=}} \varphi$, and so there is no derivation of φ from Γ in FOL$^{=}$. In the case where $\Gamma = \varnothing$, we need only provide a countermodel to φ in order to show that φ is not a theorem, i.e., $\nvdash_{\text{FOL}^{=}} \varphi$.

In addition to these advantages, FOL$^{=}$ SOUNDNESS shows that FOL$^{=}$ can be relied on to only derive logical consequences from a set of premises. Since writing semantic proofs is difficult and prone to error, establishing that $\Gamma \vDash \varphi$ by deriving φ from γ in FOL$^{=}$ is often much easier and more reliable. Moreover, given that there is at least one model $\mathcal{M}$ of $\mathcal{L}^{=}$, it follows that φ is true in $\mathcal{M}$ for any theorem $\vdash_{\text{FOL}^{=}} \varphi$. As a result, the set of all theorems of FOL$^{=}$ are satisfiable and so consistent since if $\bot$ was derivable from the set of theorems in FOL$^{=}$, then $\bot$ would be true in $\mathcal{M}$, but this leads to a contradiction. For these reasons, the following chapter will extend the proof of PL SOUNDNESS to establish FOL$^{=}$ SOUNDNESS.

Chapter 11

The Soundness of $\mathrm{FOL}^{=}$

In Chapter 4, we showed that PL was sound over its semantics. In particular, soundness showed that whenever a wfs φ of $\mathcal{L}^{\mathrm{PL}}$ is derivable in PL from some set of wfss Γ, we may conclude that φ is a logical consequence of Γ which we expressed as $\Gamma \vDash \varphi$. As a result, we could rely on derivations in PL in order to evaluate whether a conclusion of an argument was a logical consequence of its premises, making the argument valid.

Soundness was an important result to establish for PL where similar considerations extend to $\mathrm{FOL}^{=}$. If $\mathrm{FOL}^{=}$ were to fail to be sound, we would have little reason to care about $\mathrm{FOL}^{=}$ for we could not rely on it to establish valid arguments. Showing that $\mathrm{FOL}^{=}$ is sound will be the focus of this chapter where we will extend the proof that PL is sound from before.

As in the soundness proof for PL, the soundness proof for $\mathrm{FOL}^{=}$ will go by induction on the length of proof. If we can show that soundness holds for proofs of length 1 and that soundness holds for proofs of length $n + 1$ whenever soundness holds for proofs of length n or less, then we may conclude by induction that soundness holds for proofs of any length. Almost everything in the proof will remain the same as before, though now we need to check that a few extra rules also preserve logical consequence.

Since we will at times need to talk about derivability or logical consequence in different systems, subscripts will help to avoid ambiguity. Given that our primary concern is with $\mathrm{FOL}^{=}$, the default is to assume that $\vdash$ means derivability in $\mathrm{FOL}^{=}$ and $\vDash$ means logical consequence in $\mathcal{L}^{=}$. Occasionally it will improve readability to subscript these as well, writing $\vdash_{\mathrm{FOL}^{=}}$ and $\vDash_{\mathcal{L}^{=}}$ as needed. With these details in place, we now turn to the proof, beginning as before with the global argument which we will then turn to support be filling in the details.

If you get lost, or forget what was happening or why, it can help to return to this first part of the proof to regain your bearings, reflecting on what has previously been established.

11.1 Soundness

Assume $\Gamma \vdash_{\text{FOL}^=} \varphi$, written $\Gamma \vdash \varphi$ for readability. By definition, there is some proof X in FOL$^=$ of φ from the premises Γ. As in the proof of PL SOUNDNESS, it will help to introduce some notation that we will use throughout. In particular, φ_i is the sentence on the i-th line of the proof X and Γ_i includes all and only the premises and undischarged assumptions at the i-th line in X. We may then prove the following, writing $\vDash$ in place of $\vDash_{\mathcal{L}^=}$ for readability:

FOL$^=$ SOUNDNESS: Assume that $\Gamma \vdash \varphi$ for an arbitrary set Γ of wfss of $\mathcal{L}^=$ and wfs φ of $\mathcal{L}^=$. It follows that there is some FOL$^=$ derivation X of φ from Γ. Letting φ_i be the sentence on the i-th line of the derivation X and Γ_i be the set of premises that occur on any line $j \leqslant i$ of X together with the assumptions that are undischarged at line i, we may seek to prove:

Lemma 11.1: (Base Step) $\Gamma_1 \vDash \varphi_1$.

Lemma 11.13: (Induction Step) $\Gamma_{n+1} \vDash \varphi_{n+1}$ if $\Gamma_k \vDash \varphi_k$ for every $k \leqslant n$.

Given the lemmas above, it follows by strong induction that $\Gamma_n \vDash \varphi_n$ for all n. Since every proof is finite in length, there is a last line m of X where $\varphi_m = \varphi$ is the conclusion. By the definition of a FOL$^=$ derivation, we know that every assumption in X is eventually discharged, and so $\Gamma_m = \Gamma$ is the set of premises. Thus we may conclude that $\Gamma \vDash \varphi$. Discharging the assumption that $\Gamma \vdash \varphi$ and generalizing on Γ and φ completes the proof. □

Given the lemmas cited above, this proof establishes FOL$^=$ SOUNDNESS. We may now prove the supporting lemmas in a similar manner to before.

Lemma 11.1 (Base Step) $\Gamma_1 \vDash \varphi_1$.

Proof: By the definition of a FOL$^=$ derivation, φ_1 is either a premise or follows by one of the natural deduction rules for FOL$^=$. Since φ_1 is the first line of the proof, there are no earlier lines to be cited, and so φ_1 is either a premise, assumption, or follows by =I. In the first two cases, $\Gamma_1 = \{\varphi_1\}$ since φ_1 is not discharged at the first line, and so $\Gamma_1 \vDash \varphi_1$ is immediate. Thus it remains to show that $\Gamma_1 \vDash \varphi_1$ in the final case where φ_1 is $\alpha = \alpha$ for some constant α and $\Gamma_1 = \varnothing$.

Assume φ_1 is $\alpha = \alpha$ and let $\mathcal{M} = \langle \mathbb{D}, \mathcal{I} \rangle$ be an arbitrary model of $\mathcal{L}^=$. It follows that $\mathcal{I}(\alpha) \in \mathbb{D}$ where trivially $\mathcal{I}(\alpha) = \mathcal{I}(\alpha)$. Letting $\hat{a}$ be any variable assignment defined over the domain $\mathbb{D}$, it follows by definition that $\boldsymbol{v}_{\mathcal{I}}^{\hat{a}}(\alpha) = \boldsymbol{v}_{\mathcal{I}}^{\hat{a}}(\alpha)$, and so $\mathcal{V}_{\mathcal{I}}^{\hat{a}}(\alpha = \alpha) = 1$. Since $\hat{a}$ was arbitrary, $\mathcal{V}_{\mathcal{I}}(\alpha = \alpha) = 1$, and so $\vDash \alpha = \alpha$ follows be generalizing on $\mathcal{M}$. Thus $\Gamma_1 \vDash \varphi_1$ given the case assumption. □

We have already considered two proof rules in proving **Lemma 11.1** by showing that AS and =I preserve logical consequence at least in the special case as $\Gamma_1 \vDash \varphi_1$. More generally, we should like to show that all of the rules preserve logical consequence, and not just in the case of proofs with one line. Thus we will seek to establish the following:

FOL$^=$ Rules: If $\Gamma_k \vDash \varphi_k$ for every $k \leqslant n$ and φ_{n+1} follows by the proof rules for FOL$^=$, then $\Gamma_{n+1} \vDash \varphi_{n+1}$.

In order to divide the proof of FOL$^=$ Rules into more manageable parts, the following section will focus on the proof rules for PL. More specifically, we will aim to show:

PL Rules: If $\Gamma_k \vDash \varphi_k$ for every $k \leqslant n$ and φ_{n+1} follows by the proof rules for PL, then $\Gamma_{n+1} \vDash \varphi_{n+1}$.

In §11.4, we will extend the same strategy to the remaining proof rules that belong to FOL$^=$ in order to establish FOL$^=$ Rules. It is this latter result which will play a critical role in the proof of **Lemma 11.13** cited in the proof of FOL$^=$ Soundness above.

11.2 PL Rules

You might recognize PL Rules from Chapter 4, wondering why we can't simply cite this previous result. Despite the superficial similarities, PL Rules stated above says something about the wfss of $\mathcal{L}^=$, a language we had not introduced in Chapter 4. Even though PL Rules only concerns proofs rules that occur in PL, the semantic turnstile $\vDash$ used above quantifies over the models of $\mathcal{L}^=$, not the interpretations of $\mathcal{L}^{\text{PL}}$. Were we to disambiguate, we may replace the turnstiles above with $\vDash_{\mathcal{L}^=}$ and not $\vDash_{\mathcal{L}^{\text{PL}}}$, however tempting.

Given these caveats, you still might wonder why we can't just cite our previous result. After all, we showed that PL is sound over its semantics. Shouldn't the result somehow carry over to allow us to assert PL Rules without saying much more?

The answer is that there are proof strategies that go this way, though they typically go one of two ways. Either they merely wave their hands, suppressing the details that make the proof worth reading, or they define an injection from $\mathcal{L}^=$ into $\mathcal{L}^{\text{PL}}$ in order to make use of PL Soundness. Since this latter strategy is abstract and cumbersome, and the former is pointless, we will follow the much more concrete approach of simply revising our former proofs. In addition to providing the opportunity to review how the proof of PL Soundness worked before, we will also be in a position to omit certain elements when the details are very similar to the proofs that we already provided above. Nevertheless, by referring to the proofs in Chapter 4, it should be possible to reconstruct every element of the proof in rigorous detail.

11.2.1 Assumption and Reiteration

Before attending to the introduction and elimination rules for each of the logical operators included in PL, this section will focus on the assumption and reiteration rules. Whereas the proofs for most of the rules will appeal to the induction hypothesis given above, the proof for the assumption rule AS is an exception and is similar to what was given in **Lemma 11.1**.

Rule 1 (AS) $\Gamma_{n+1} \vDash \varphi_{n+1}$ if φ_{n+1} follows from Γ_{n+1} by the rule AS.

Proof: Assume that φ_{n+1} follows by the assumption rule AS from the wfss in Γ_{n+1}. Since φ_{n+1} is an undischarged assumption, it follows that $\varphi_{n+1} \in \Gamma_{n+1}$, and so $\mathcal{V}_{\mathcal{I}}(\varphi_{n+1}) = 1$ for any model $\mathcal{M} = \langle \mathbb{D}, \mathcal{I} \rangle$ where $\mathcal{V}_{\mathcal{I}}(\chi) = 1$ for all $\chi \in \Gamma_{n+1}$. By definition, it follows that $\Gamma_{n+1} \vDash \varphi_{n+1}$ as desired. □

The proof above does not require the induction hypothesis or any additional results. By contrast, it will help to establish the reiteration rule by first recalling the following lemmas.

Lemma 2.1: If $\Gamma \vDash \varphi$, then $\Gamma \cup \Sigma \vDash \varphi$.

Lemma 4.3: If φ_k is live at line n of an FOL$^=$ derivation where $k \leqslant n$, then $\Gamma_k \subseteq \Gamma_n$.

The proofs for these lemmas is very similar to what it was before though Γ is now permitted to be any set of wfss of $\mathcal{L}^=$, where similarly, φ is any wfs of $\mathcal{L}^=$. It is nevertheless worth looking back to confirm that the old proofs continue to apply. Given these two lemmas, we may now proceed to establish the reiteration rule R in the same manner as before.

Rule 2 (R) $\Gamma_{n+1} \vDash \varphi_{n+1}$ if φ_{n+1} follows from Γ_{n+1} by the rule R.

Proof: Assume that φ_{n+1} follows by the reiteration rule R from the sentences in Γ_{n+1}. It follows that $\varphi_{n+1} = \varphi_k$ for some $k \leqslant n$, and so $\Gamma_k \vDash \varphi_k$ by hypothesis. Since φ_k is live at line $n+1$, we know by **Lemma 4.3** that $\Gamma_k \subseteq \Gamma_{n+1}$, and so $\Gamma_{n+1} \vDash \varphi_k$ by **Lemma 2.1**. Thus $\Gamma_{n+1} \vDash \varphi_{n+1}$ given the identity above. □

Given that nothing needs to change about the proof of **Rule 2**, you might suspect that all of the proofs for the PL rules will be unchanged. This is true in some cases and not in others where the same may be said for some of the lemmas established before.

11.2.2 Negation Rules

In order to show that the negation rules preserve logical consequence, Chapter 4 appealed to two supporting lemmas. Since these lemmas will continue to be important for what follows, we prove that they continue to hold given the semantics for $\mathcal{L}^=$ which, unlike before, includes variable assignments. In particular, consider the following proof:

Lemma 11.2 If $\Gamma \vDash \varphi$ and $\Gamma \vDash \neg\varphi$, then Γ is unsatisfiable.

Proof: Assume $\Gamma \vDash \varphi$ and $\Gamma \vDash \neg\varphi$. Assume for contradiction that Γ is satisfiable, and so there is some model $\mathcal{M} = \langle \mathbb{D}, \mathcal{I} \rangle$ where $\mathcal{V}_{\mathcal{I}}(\chi) = 1$ for all $\chi \in \Gamma$. It follows from the assumption that both $\mathcal{V}_{\mathcal{I}}(\varphi) = 1$ and $\mathcal{V}_{\mathcal{I}}(\neg\varphi) = 1$, and so $\mathcal{V}_{\mathcal{I}}^{\hat{a}}(\neg\varphi) = 1$ follows from the latter for some $\hat{a}$ by **Lemma 9.2**. By the semantics for negation, $\mathcal{V}_{\mathcal{I}}^{\hat{a}}(\varphi) \neq 1$, and so $\mathcal{V}_{\mathcal{I}}(\varphi) \neq 1$ contradicting the above. Thus Γ is unsatisfiable. $\square$

The lemma above is much as it was before save for some extra details to do with variable assignments. Nevertheless, the result holds for the same basic reason. Something similar may be said for the following lemma which has been left as an exercise:

Lemma 11.3 $\Gamma \cup \{\varphi\}$ is unsatisfiable just in case $\Gamma \vDash \neg\varphi$.

Proof: This proof is left as an exercise for the reader.

Given the updated lemmas above, the proof of $\neg$I is exactly the same as it was before. Omitting the details of the proof here, it worth reviewing the proof given in Chapter 4.

Rule 3 ($\neg$I) $\Gamma_{n+1} \vDash \varphi_{n+1}$ if φ_{n+1} follows from Γ_{n+1} by the rule $\neg$I.

Despite how similar the introduction and elimination rules are for negation, a few minor amendments are required in order to extend the proof of **Rule 4** to hold for the wfss of $\mathcal{L}^=$.

Rule 4 ($\neg$E) $\Gamma_{n+1} \vDash \varphi_{n+1}$ if φ_{n+1} follows from Γ_{n+1} by the rule $\neg$E.

Proof: Assume φ_{n+1} follows from Γ_{n+1} by negation elimination ¬E. Thus there is some subproof on lines i-j where $i < j \leqslant n$ and $\varphi_i = \neg\varphi_{n+1}$, $\psi = \varphi_h$, and $\neg\psi = \varphi_k$ for $i \leqslant h \leqslant j$ and $i \leqslant k \leqslant j$. By parity of reasoning, we may assume that $h < k = j$. Thus we may represent the subproof as follows:

i		$\neg\varphi$	:AS for ¬E
h		ψ	
j		$\neg\psi$	
$n+1$	φ		:i–j ¬E

By hypothesis, $\Gamma_h \vDash \psi$ and $\Gamma_j \vDash \neg\psi$. With the exception of $\varphi_i = \neg\varphi$, the undischarged assumptions at lines h and j are also undischarged at line $n+1$. It follows that $\Gamma_h, \Gamma_j \subseteq \Gamma_{n+1} \cup \{\varphi_i\}$, and so $\Gamma_{n+1} \cup \{\varphi_i\} \vDash \psi$ and $\Gamma_{n+1} \cup \{\varphi_i\} \vDash \neg\psi$ by **Lemma 2.1**. Thus $\Gamma_{n+1} \cup \{\varphi_i\}$ is unsatisfiable by **Lemma 11.2**, and so $\Gamma_{n+1} \vDash \neg\varphi_i$ by **Lemma 11.3**. Equivalently, $\Gamma_{n+1} \vDash \neg\neg\varphi_{n+1}$. Given any model $\mathcal{M} = \langle \mathbb{D}, \mathcal{I} \rangle$ where $\mathcal{V}_{\mathcal{I}}(\chi) = 1$ for all $\chi \in \Gamma_{n+1}$, it follows that $\mathcal{V}_{\mathcal{I}}(\neg\neg\varphi_{n+1}) = 1$, and so $\mathcal{V}_{\mathcal{I}}^{\hat{a}}(\neg\neg\varphi_{n+1}) = 1$ for all $\hat{a}$. By two applications of the semantics for negation, it follows that $\mathcal{V}_{\mathcal{I}}^{\hat{a}}(\varphi_{n+1}) = 1$ all $\hat{a}$, and so $\mathcal{V}_{\mathcal{I}}(\varphi_{n+1}) = 1$. Thus we may conclude by generalizing on $\mathcal{M}$ that $\Gamma_{n+1} \vDash \varphi_{n+1}$ as desired. □

This proof is almost identical to the **Rule 3** except that an additional negation sign is introduced before eliminating the double negation by appealing to the semantics.

11.2.3 Conjunction and Disjunction

The following rule is established in much the same way as before save for minor details particular to the semantics for $\mathcal{L}^=$ and so has been left as an exercise for the reader.

Rule 5 (&I) $\Gamma_{n+1} \vDash \varphi_{n+1}$ if φ_{n+1} follows from Γ_{n+1} by the rule ∧I.

Proof: This proof is left as an exercise for the reader. □

Since the details for the following proof rule were omitted before, they will be provided here for completeness. Nevertheless, few changes are required for the proof to go through.

Rule 6 (&E) $\Gamma_{n+1} \vDash \varphi_{n+1}$ if φ_{n+1} follows from Γ_{n+1} by the rule ∧E.

Proof: Assuming φ_{n+1} follows from Γ_{n+1} by conjunction elimination $\wedge$E, there is some $i \leqslant n$ where either $\varphi_i = \varphi_{n+1} \wedge \psi$ or $\varphi_i = \psi \wedge \varphi_{n+1}$ is live at line $n+1$. By hypothesis, $\Gamma_i \vDash \varphi_i$ where $\Gamma_i \subseteq \Gamma_{n+1}$ by **Lemma 4.3**, and so $\Gamma_{n+1} \vDash \varphi_i$ by **Lemma 2.1**. Letting $\mathcal{M} = \langle \mathbb{D}, \mathcal{I} \rangle$ be a model of $\mathcal{L}^=$ where $\mathcal{V}_{\mathcal{I}}(\chi) = 1$ for all $\chi \in \Gamma_{n+1}$, it follows that $\mathcal{V}_{\mathcal{I}}(\varphi_i) = 1$, and so either $\mathcal{V}_{\mathcal{I}}(\varphi_{n+1} \wedge \psi) = 1$ or $\mathcal{V}_{\mathcal{I}}(\psi \wedge \varphi_{n+1}) = 1$. By **Lemma 9.2**, either $\mathcal{V}_{\mathcal{I}}^{\hat{a}}(\varphi_{n+1} \wedge \psi) = 1$ or $\mathcal{V}_{\mathcal{I}}^{\hat{a}}(\psi \wedge \varphi_{n+1}) = 1$ for some v.a. $\hat{a}$, and so either way $\mathcal{V}_{\mathcal{I}}^{\hat{a}}(\varphi_{n+1}) = 1$ by the semantics for conjunction. Thus $\mathcal{V}_{\mathcal{I}}(\varphi_{n+1}) = 1$ again by **Lemma 9.2**, and so $\Gamma_{n+1} \vDash \varphi_{n+1}$. □

Rule 7 ($\vee$I) $\Gamma_{n+1} \vDash \varphi_{n+1}$ if φ_{n+1} follows from Γ_{n+1} by the rule $\vee$I.

Proof: Assume that φ_{n+1} follows from Γ_{n+1} by disjunction introduction $\vee$I. Thus $\varphi_{n+1} = \varphi_i \vee \psi$ or $\varphi_{n+1} = \psi \vee \varphi_i$ for some line $i \leqslant n$ that is live at line $n+1$. By hypothesis, $\Gamma_i \vDash \varphi_i$ where $\Gamma_i \subseteq \Gamma_{n+1}$ by **Lemma 4.3**, and so $\Gamma_{n+1} \vDash \varphi_i$ by **Lemma 2.1**. Letting $\mathcal{M} = \langle \mathbb{D}, \mathcal{I} \rangle$ be a model where $\mathcal{V}_{\mathcal{I}}(\chi) = 1$ for all $\chi \in \Gamma_{n+1}$, it follows that $\mathcal{V}_{\mathcal{I}}(\varphi_i) = 1$, and so $\mathcal{V}_{\mathcal{I}}^{\hat{a}}(\varphi_i) = 1$ for some variable assignment $\hat{a}$ by **Lemma 9.2**. By the semantics for disjunction, both $\mathcal{V}_{\mathcal{I}}^{\hat{a}}(\varphi_i \vee \psi) = 1$ and $\mathcal{V}_{\mathcal{I}}^{\hat{a}}(\psi \vee \varphi_i) = 1$, and so $\mathcal{V}_{\mathcal{I}}^{\hat{a}}(\varphi_{n+1}) = 1$. Thus $\mathcal{V}_{\mathcal{I}}(\varphi_{n+1}) = 1$ again by **Lemma 9.2**, and so $\Gamma_{n+1} \vDash \varphi_{n+1}$ follows from generalizing on $\mathcal{M}$. □

Neither of the rule proofs above should surprise, amounting to little more than applications of the semantic clauses for conjunction and disjunction respectively. The only differences with the proofs given before concern the way that the wfss of $\mathcal{L}^=$ are assigned truth-values relative to models instead of interpretations and the way that variable assignments are negotiated throughout. Since the proof rules for PL do not appeal to variables, little turns on these extra details. Rather, they are included only to conform to the strict letter of the definitions.

Something similar may be said for the following proof though a little more care is required to keep track of all of the moving parts in the proof rule for disjunction elimination.

Rule 8 ($\vee$E) $\Gamma_{n+1} \vDash \varphi_{n+1}$ if φ_{n+1} follows from Γ_{n+1} by the rule $\vee$E.

Proof: Assume φ_{n+1} follows from Γ_{n+1} by disjunction elimination $\vee$I. Thus there is some line $\varphi_i = \varphi_j \vee \varphi_k$ which is live at $n+1$ and subproofs on lines j-h and k-l where $i < j, k, h, l \leqslant n$ and $\varphi_h = \varphi_l = \varphi_{n+1}$. By parity of reasoning, we may assume that $h < k$, and so represent the proof as follows:

$$\begin{array}{l|l|l}
i & \varphi \vee \psi & \\
j & \quad \varphi & \text{:AS for } \vee\text{E} \\
h & \quad \chi & \\
k & \quad \psi & \text{:AS for } \vee\text{E} \\
l & \quad \chi & \\
n+1 & \chi & :i,\ j\text{–}h,\ k\text{–}l\ \vee\text{E}
\end{array}$$

By hypothesis, $\Gamma_i \vDash \varphi_i$, $\Gamma_h \vDash \varphi_h$, and $\Gamma_l \vDash \varphi_l$ where $\Gamma_i \subseteq \Gamma_{n+1}$ all follow by **Lemma 4.3**, and so $\Gamma_{n+1} \vDash \varphi_i$ by **Lemma 2.1**. With the exception of $\varphi_j = \varphi$, every assumption that is undischarged at line h is also undischarged at line $n+1$, and so $\Gamma_h \subseteq \Gamma_{n+1} \cup \{\varphi_j\}$. Similarly, we may conclude that $\Gamma_l \subseteq \Gamma_{n+1} \cup \{\varphi_k\}$, and so $\Gamma_{n+1} \cup \{\varphi_j\} \vDash \varphi_h$ and $\Gamma_{n+1} \cup \{\varphi_k\} \vDash \varphi_l$ by **Lemma 2.1**.

Letting $\mathcal{M} = \langle \mathbb{D}, \mathcal{I} \rangle$ be any model where $\mathcal{V}_{\mathcal{I}}(\chi) = 1$ for all $\chi \in \Gamma_{n+1}$, it follows from above that $\mathcal{V}_{\mathcal{I}}(\varphi_i) = 1$. Equivalently, $\mathcal{V}_{\mathcal{I}}(\varphi_j \vee \varphi_k) = 1$, and so $\mathcal{V}_{\mathcal{I}}^{\hat{a}}(\varphi_j \vee \varphi_k) = 1$ for some v.a. $\hat{a}$ defined over $\mathbb{D}$ by **Lemma 9.2**. By the semantics for disjunction, $\mathcal{V}_{\mathcal{I}}^{\hat{a}}(\varphi_j) = 1$ or $\mathcal{V}_{\mathcal{I}}^{\hat{a}}(\varphi_k) = 1$, and so $\mathcal{V}_{\mathcal{I}}(\varphi_j) = 1$ or $\mathcal{V}_{\mathcal{I}}(\varphi_k) = 1$ by **Lemma 9.2**. If $\mathcal{V}_{\mathcal{I}}(\varphi_j) = 1$, then $\mathcal{V}_{\mathcal{I}}(\chi) = 1$ for all $\chi \in \Gamma_{n+1} \cup \{\varphi_j\}$, and so $\mathcal{V}_{\mathcal{I}}(\varphi_{n+1}) = 1$ since $\Gamma_{n+1} \cup \{\varphi_j\} \vDash \varphi_h$ and $\varphi_h = \varphi_{n+1}$. If $\mathcal{V}_{\mathcal{I}}(\varphi_k) = 1$, then $\mathcal{V}_{\mathcal{I}}(\chi) = 1$ for all $\chi \in \Gamma_{n+1} \cup \{\varphi_k\}$, and so $\mathcal{V}_{\mathcal{I}}(\varphi_{n+1}) = 1$ since $\Gamma_{n+1} \cup \{\varphi_k\} \vDash \varphi_l$ and $\varphi_l = \varphi_{n+1}$. Either way, $\mathcal{V}_{\mathcal{I}}(\varphi_{n+1}) = 1$, and so $\Gamma_{n+1} \vDash \varphi_{n+1}$ by generalizing on $\mathcal{M}$. □

As with the previous two proof rules for conjunction and disjunction, the proof above turns on little more than an application of the semantics for disjunction.

11.2.4 Conditional Rules

The elimination rules for the conditional and the biconditional are also straightforward applications of the semantics. By contrast, the introduction rules for the conditional and biconditional benefit from the following analogue of **Lemma 4.6** from Chapter 4. Since the proofs are very similar, the details have been left as an exercise for the reader.

Lemma 11.4 If $\Gamma \cup \{\varphi\} \vDash \psi$, then $\Gamma \vDash \varphi \rightarrow \psi$.

Proof: This proof is left as an exercise for the reader. □

The application of the previous lemma in the proof of the following rule is unchanged from before, and so the details of the proof will be omitted.

> **Rule 9 (→I)** $\Gamma_{n+1} \vDash \varphi_{n+1}$ if φ_{n+1} follows from Γ_{n+1} by the rule →I.

Having previously left the proof for the following proof rule as an exercise for the reader, we now provide the following details for completeness:

> **Rule 10 (→E)** $\Gamma_{n+1} \vDash \varphi_{n+1}$ if φ_{n+1} follows from Γ_{n+1} by the rule →E.

Proof: Assume φ_{n+1} follows from Γ_{n+1} by conditional introduction →E. Thus there are some lines $\varphi_i = \varphi_j \rightarrow \varphi_{n+1}$ and φ_j for $i, j \leqslant n$ which are live at $n+1$, and so $\Gamma_i, \Gamma_j \subseteq \Gamma_{n+1}$ by **Lemma 4.3**. By hypothesis, $\Gamma_i \vDash \varphi_i$ and $\Gamma_j \vDash \varphi_j$, and so both $\Gamma_{n+1} \vDash \varphi_i$ and $\Gamma_{n+1} \vDash \varphi_j$ by **Lemma 2.1**.

Letting $\mathcal{M} = \langle \mathbb{D}, \mathcal{I} \rangle$ be any model where $\mathcal{V}_{\mathcal{I}}(\chi) = 1$ for all $\chi \in \Gamma_{n+1}$, it follows that $\mathcal{V}_{\mathcal{I}}(\varphi_i) = \mathcal{V}_{\mathcal{I}}(\varphi_j) = 1$. It follows that $\mathcal{V}_{\mathcal{I}}^{\hat{a}}(\varphi_i) = \mathcal{V}_{\mathcal{I}}^{\hat{a}}(\varphi_j \rightarrow \varphi_{n+1}) = \mathcal{V}_{\mathcal{I}}^{\hat{a}}(\varphi_j) = 1$ for all variable assignment $\hat{a}$ over $\mathbb{D}$, and so for some $\hat{a}$ in particular.

By the semantics for the conditional, either $\mathcal{V}_{\mathcal{I}}^{\hat{a}}(\varphi_j) \neq 1$ or $\mathcal{V}_{\mathcal{I}}^{\hat{a}}(\varphi_{n+1}) = 1$. Given the above, we may conclude that $\mathcal{V}_{\mathcal{I}}^{\hat{a}}(\varphi_{n+1}) = 1$, and so $\mathcal{V}_{\mathcal{I}}(\varphi_{n+1}) = 1$ follows by **Lemma 9.2** . Generalizing on $\mathcal{M}$, it follows that $\Gamma_{n+1} \vDash \varphi_{n+1}$ as desired. □

> **Rule 11 (↔I)** $\Gamma_{n+1} \vDash \varphi_{n+1}$ if φ_{n+1} follows from Γ_{n+1} by the rule ↔I.

Proof: Assume φ_{n+1} follows from Γ_{n+1} by biconditional introduction ↔E. Thus there are some subproofs on lines i-j and h-k for some $i, j < h, k \leqslant n$ where $\varphi_i = \varphi_h = \varphi$, $\varphi_j = \varphi_k = \psi$, and either $\varphi_{n+1} = \varphi \leftrightarrow \psi$ or $\varphi_{n+1} = \psi \leftrightarrow \varphi$. By parity of reasoning, we may assume that $\varphi_{n+1} = \varphi \leftrightarrow \psi$. Thus we have:

i	$\quad\mid\ \varphi$	:AS for ∨E
j	$\quad\mid\ \psi$	
h	$\quad\mid\ \psi$	:AS for ∨E
k	$\quad\mid\ \varphi$	
$n+1$	$\varphi \leftrightarrow \psi$	:i–j, h–k ↔I

By hypothesis, $\Gamma_j \vDash \varphi_j$, $\Gamma_k \vDash \varphi_k$, and $\Gamma_{n+1} \vDash \varphi_{n+1}$. With the exception of φ_i, every assumption that is undischarged at line j is also undischarged at line $n+1$, and so $\Gamma_j \subseteq \Gamma_{n+1} \cup \{\varphi_i\}$. Similarly, we may conclude that $\Gamma_k \subseteq \Gamma_{n+1} \cup \{\varphi_h\}$, and so $\Gamma_{n+1} \cup \{\varphi_i\} \vDash \varphi_j$ and $\Gamma_{n+1} \cup \{\varphi_h\} \vDash \varphi_k$ by **Lemma 2.1**.

By **Lemma 11.4**, both $\Gamma_{n+1} \vDash \varphi_i \to \varphi_j$ and $\Gamma_{n+1} \vDash \varphi_h \to \varphi_k$. Equivalently, $\Gamma_{n+1} \vDash \varphi \to \psi$ and $\Gamma_{n+1} \vDash \psi \to \varphi$. Letting $\mathcal{M} = \langle \mathbb{D}, \mathcal{I} \rangle$ be any model where $\mathcal{V}_{\mathcal{I}}(\chi) = 1$ for all $\chi \in \Gamma_{n+1}$, it follows that $\mathcal{V}_{\mathcal{I}}(\varphi \to \psi) = \mathcal{V}_{\mathcal{I}}(\psi \to \varphi) = 1$ given the results above. Thus $\mathcal{V}_{\mathcal{I}}^{\hat{a}}(\varphi \to \psi) = \mathcal{V}_{\mathcal{I}}^{\hat{a}}(\psi \to \varphi) = 1$ for all variable assignments $\hat{a}$, and so for some $\hat{a}$ in particular. By the semantics for the conditional, $\mathcal{V}_{\mathcal{I}}^{\hat{a}}(\varphi) \neq 1$ or $\mathcal{V}_{\mathcal{I}}^{\hat{a}}(\psi) = 1$, and $\mathcal{V}_{\mathcal{I}}^{\hat{a}}(\psi) \neq 1$ or $\mathcal{V}_{\mathcal{I}}^{\hat{a}}(\varphi) = 1$.

As a result, $\mathcal{V}_{\mathcal{I}}^{\hat{a}}(\psi) = 1$ if $\mathcal{V}_{\mathcal{I}}^{\hat{a}}(\varphi) = 1$, and similarly, $\mathcal{V}_{\mathcal{I}}^{\hat{a}}(\varphi) = 1$ if $\mathcal{V}_{\mathcal{I}}^{\hat{a}}(\psi) = 1$, and so $\mathcal{V}_{\mathcal{I}}^{\hat{a}}(\varphi) = \mathcal{V}_{\mathcal{I}}^{\hat{a}}(\psi)$. Thus $\mathcal{V}_{\mathcal{I}}^{\hat{a}}(\varphi \leftrightarrow \psi) = 1$ by the semantics for the biconditional, and so $\mathcal{V}_{\mathcal{I}}(\varphi_{n+1})$ by **Lemma 9.2**. Generalizing on $\mathcal{M}$, we know $\Gamma_{n+1} \vDash \varphi_{n+1}$. □

Rule 12 (↔E) $\Gamma_{n+1} \vDash \varphi_{n+1}$ if φ_{n+1} follows from Γ_{n+1} by the rule ↔E.

Proof: Assume φ_{n+1} follows from Γ_{n+1} by biconditional introduction →E. Thus there are some lines $i, j \leqslant n$ that are live at $n+1$ where either $\varphi_i = \varphi_j \leftrightarrow \varphi_{n+1}$ or $\varphi_i = \varphi_{n+1} \leftrightarrow \varphi_j$. By parity of reasoning, we may assume that $\varphi_i = \varphi_j \leftrightarrow \varphi_{n+1}$ where $\Gamma_i, \Gamma_j \subseteq \Gamma_{n+1}$ follows by **Lemma 4.3**. By hypothesis, $\Gamma_i \vDash \varphi_i$ and $\Gamma_j \vDash \varphi_j$, and so $\Gamma_{n+1} \vDash \varphi_i$ and $\Gamma_{n+1} \vDash \varphi_j$ by **Lemma 2.1**.

Letting $\mathcal{M} = \langle \mathbb{D}, \mathcal{I} \rangle$ be any model where $\mathcal{V}_{\mathcal{I}}(\chi) = 1$ for all $\chi \in \Gamma_{n+1}$, it follows that $\mathcal{V}_{\mathcal{I}}(\varphi_i) = \mathcal{V}_{\mathcal{I}}(\varphi_j) = 1$. Thus $\mathcal{V}_{\mathcal{I}}^{\hat{a}}(\varphi_i) = \mathcal{V}_{\mathcal{I}}^{\hat{a}}(\varphi_j \leftrightarrow \varphi_{n+1}) = \mathcal{V}_{\mathcal{I}}^{\hat{a}}(\varphi_j) = 1$ for every v.a. $\hat{a}$ defined over $\mathbb{D}$, and so for some $\hat{a}$ in particular. By the semantics for the biconditional, $\mathcal{V}_{\mathcal{I}}^{\hat{a}}(\varphi_j) = \mathcal{V}_{\mathcal{I}}^{\hat{a}}(\varphi_{n+1})$, and so $\mathcal{V}_{\mathcal{I}}^{\hat{a}}(\varphi_{n+1}) = 1$. Thus $\mathcal{V}_{\mathcal{I}}(\varphi_{n+1}) = 1$ by **Lemma 9.2**, and so $\Gamma_{n+1} \vDash \varphi_{n+1}$ follows by generalizing on $\mathcal{M}$. □

These final results complete the last of the proofs for all of the rules included in PL. Given **Rule 1** – **Rule 12**, we may report the following preliminary result:

PL RULES: If $\Gamma_k \vDash \varphi_k$ for every $k \leqslant n$ and φ_{n+1} follows by the proof rules for PL, then $\Gamma_{n+1} \vDash \varphi_{n+1}$.

It remains to extend this result to include the remaining proof rules in FOL$^=$. In order to do so, the following section will prove two important supporting lemmas. Whereas the lemmas above merely adapted the lemmas already given in Chapter 4, the lemmas proven in the following section are entirely novel to $\mathcal{L}^=$. We will then put these lemmas to work in order to establish FOL$^=$ RULES in the following section.

11.3 Substitution and Model Lemmas

This section establishes two closely related results, both of which show that the truth-value of a wff is preserved by specific changes to that wff or to the model in which it is evaluated. These results will play a crucial role in proving the lemmas that we will need to show that the remaining proof rules that belong to FOL$^=$ preserve logical consequence.

In slightly greater detail, the following lemma shows that replacing α with β in a wff φ does not effect its truth-value when evaluated at a model and variable assignment so long as α and β refer to the same element of the domain on that model and variable assignment.

Lemma 11.5 $\mathcal{V}_{\mathcal{I}}^{\hat{a}}(\varphi) = \mathcal{V}_{\mathcal{I}}^{\hat{a}}(\varphi[\beta/\alpha])$ if $\boldsymbol{v}_{\mathcal{I}}^{\hat{a}}(\alpha) = \boldsymbol{v}_{\mathcal{I}}^{\hat{a}}(\beta)$ and β is free for α in φ.

Proof: The proof goes by induction on the wff of $\mathcal{L}^=$.

Base: Let φ be a wff of $\mathcal{L}^=$ where $\texttt{Comp}(\varphi) = 0$. Assume that $\boldsymbol{v}_{\mathcal{I}}^{\hat{a}}(\alpha) = \boldsymbol{v}_{\mathcal{I}}^{\hat{a}}(\beta)$. It follows that φ is either $\mathcal{F}^n\alpha_1, \ldots \alpha_n$ or $\alpha_1 = \alpha_2$. If φ is $\mathcal{F}^n\alpha_1, \ldots \alpha_n$ where $\gamma_i = \beta$ if $\alpha_i = \alpha$ and otherwise $\gamma_i = \alpha_i$, then we have:

$$\begin{array}{rll}
\mathcal{V}_{\mathcal{I}}^{\hat{a}}(\varphi) = 1 & \textit{iff} & \mathcal{V}_{\mathcal{I}}^{\hat{a}}(\mathcal{F}^n\alpha_1, \ldots, \alpha_n) = 1 \\
& \textit{iff} & \langle \boldsymbol{v}_{\mathcal{I}}^{\hat{a}}(\alpha_1), \ldots, \boldsymbol{v}_{\mathcal{I}}^{\hat{a}}(\alpha_n) \rangle \in \mathcal{I}(\mathcal{F}^n) \\
(\star) & \textit{iff} & \langle \boldsymbol{v}_{\mathcal{I}}^{\hat{a}}(\gamma_1), \ldots, \boldsymbol{v}_{\mathcal{I}}^{\hat{a}}(\gamma_n) \rangle \in \mathcal{I}(\mathcal{F}^n) \\
& \textit{iff} & \mathcal{V}_{\mathcal{I}}^{\hat{a}}(\mathcal{F}^n\gamma_1, \ldots, \gamma_n) = 1 \\
& \textit{iff} & \mathcal{V}_{\mathcal{I}}^{\hat{a}}(\varphi[\beta/\alpha]) = 1.
\end{array}$$

Whenever $\alpha_i = \alpha$, it follows that $\boldsymbol{v}_{\mathcal{I}}^{\hat{a}}(\alpha_i) = \boldsymbol{v}_{\mathcal{I}}^{\hat{a}}(\alpha) = \boldsymbol{v}_{\mathcal{I}}^{\hat{a}}(\beta)$ by assumption. Since $\boldsymbol{v}_{\mathcal{I}}^{\hat{a}}(\beta) = \boldsymbol{v}_{\mathcal{I}}^{\hat{a}}(\gamma_i)$ by definition, we may conclude that $\boldsymbol{v}_{\mathcal{I}}^{\hat{a}}(\alpha_i) = \boldsymbol{v}_{\mathcal{I}}^{\hat{a}}(\gamma_i)$. If $\alpha_i \neq \alpha$, then $\alpha_i = \gamma_i$ by definition, and so $\boldsymbol{v}_{\mathcal{I}}^{\hat{a}}(\alpha_i) = \boldsymbol{v}_{\mathcal{I}}^{\hat{a}}(\gamma_i)$ is immediate. It follows that $\boldsymbol{v}_{\mathcal{I}}^{\hat{a}}(\alpha_i) = \boldsymbol{v}_{\mathcal{I}}^{\hat{a}}(\gamma_i)$ for all $1 \leqslant i \leqslant n$, thereby justifying $(\star)$. The other biconditionals hold by definition or the semantics for atomic wffs of $\mathcal{L}^=$.

If instead φ is $\alpha_1 = \alpha_n$, then assuming for all $1 \leqslant i \leqslant n$ as before that $\gamma_i = \beta$ if $\alpha_i = \alpha$ and otherwise $\gamma_i = \alpha_i$, we have the following biconditionals:

$$\begin{array}{rll}
\mathcal{V}_{\mathcal{I}}^{\hat{a}}(\varphi) = 1 & \textit{iff} & \mathcal{V}_{\mathcal{I}}^{\hat{a}}(\alpha_1 = \alpha_2) = 1 \\
& \textit{iff} & \boldsymbol{v}_{\mathcal{I}}^{\hat{a}}(\alpha_1) = \boldsymbol{v}_{\mathcal{I}}^{\hat{a}}(\alpha_n) \\
(*) & \textit{iff} & \boldsymbol{v}_{\mathcal{I}}^{\hat{a}}(\gamma_1) = \boldsymbol{v}_{\mathcal{I}}^{\hat{a}}(\gamma_n) \\
& \textit{iff} & \mathcal{V}_{\mathcal{I}}^{\hat{a}}(\gamma_1 = \gamma_n) = 1 \\
& \textit{iff} & \mathcal{V}_{\mathcal{I}}^{\hat{a}}(\varphi[\beta/\alpha]) = 1.
\end{array}$$

We may justify $(*)$ in an analogous manner to $(\star)$, where the justifications for the other biconditionals is the same as before. It follows that $\mathcal{V}_{\mathcal{I}}^{\hat{a}}(\varphi) = \mathcal{V}_{\mathcal{I}}^{\hat{a}}(\varphi[\beta/\alpha])$ whenever $\boldsymbol{v}_{\mathcal{I}}^{\hat{a}}(\alpha) = \boldsymbol{v}_{\mathcal{I}}^{\hat{a}}(\beta)$ and $\texttt{Comp}(\varphi) = 0$.

Induction: Assume that if $\texttt{Comp}(\varphi) \leqslant n$, then $\mathcal{V}_{\mathcal{I}}^{\hat{a}}(\varphi) = \mathcal{V}_{\mathcal{I}}^{\hat{a}}(\varphi[\beta/\alpha])$ whenever $\boldsymbol{v}_{\mathcal{I}}^{\hat{a}}(\alpha) = \boldsymbol{v}_{\mathcal{I}}^{\hat{a}}(\beta)$. Letting $\texttt{Comp}(\varphi) = n+1$, there are seven cases to consider corresponding to the operators $\neg, \wedge, \vee, \rightarrow, \leftrightarrow, \forall\gamma$, and $\exists\gamma$.

Case 1: Assume that $\varphi = \neg\psi$ where $\boldsymbol{v}_{\mathcal{I}}^{\hat{a}}(\alpha) = \boldsymbol{v}_{\mathcal{I}}^{\hat{a}}(\beta)$. Since $\texttt{Comp}(\varphi) = n+1$ and $\texttt{Comp}(\neg\psi) = \texttt{Comp}(\psi) + 1$, it follows that $\texttt{Comp}(\psi) \leqslant n$. It follows by hypothesis that $\mathcal{V}_{\mathcal{I}}^{\hat{a}}(\psi) = \mathcal{V}_{\mathcal{I}}^{\hat{a}}(\psi[\beta/\alpha])$, and so $\mathcal{V}_{\mathcal{I}}^{\hat{a}}(\neg\psi) = \mathcal{V}_{\mathcal{I}}^{\hat{a}}(\neg\psi[\beta/\alpha])$ by the semantics for negation. Thus $\mathcal{V}_{\mathcal{I}}^{\hat{a}}(\varphi) = \mathcal{V}_{\mathcal{I}}^{\hat{a}}(\varphi[\beta/\alpha])$ as desired.

Case 6: Assume $\varphi = \forall\gamma\psi$ where $\boldsymbol{v}_{\mathcal{I}}^{\hat{a}}(\alpha) = \boldsymbol{v}_{\mathcal{I}}^{\hat{a}}(\beta)$. If $\gamma = \alpha$, if follows that α is not free in φ, and so trivially $\varphi = \varphi[\beta/\alpha]$. Thus $\mathcal{V}_{\mathcal{I}}^{\hat{a}}(\varphi) = \mathcal{V}_{\mathcal{I}}^{\hat{a}}(\varphi[\beta/\alpha])$ is immediate. Assume instead that $\gamma \neq \alpha$ and consider the following biconditionals:

$$
\begin{array}{rll}
\mathcal{V}_{\mathcal{I}}^{\hat{a}}(\varphi) = 1 & \textit{iff} & \mathcal{V}_{\mathcal{I}}^{\hat{a}}(\forall\gamma\psi) = 1 \\
 & \textit{iff} & \mathcal{V}_{\mathcal{I}}^{\hat{e}}(\psi) = 1 \text{ for all } \gamma\text{-variants } \hat{e} \text{ of } \hat{a} \\
(\dagger) & \textit{iff} & \mathcal{V}_{\mathcal{I}}^{\hat{e}}(\psi[\beta/\alpha]) = 1 \text{ for all } \gamma\text{-variants } \hat{e} \text{ of } \hat{a} \\
 & \textit{iff} & \mathcal{V}_{\mathcal{I}}^{\hat{a}}(\forall\gamma\psi[\beta/\alpha]) = 1 \\
 & \textit{iff} & \mathcal{V}_{\mathcal{I}}^{\hat{a}}(\varphi[\beta/\alpha]) = 1.
\end{array}
$$

Let $\hat{e}$ be an arbitrary γ-variant of $\hat{a}$. Since $\gamma \neq \alpha$, it follows that $\hat{e}(\alpha) = \hat{a}(\alpha)$ if α is a variable, and so $\boldsymbol{v}_{\mathcal{I}}^{\hat{e}}(\alpha) = \boldsymbol{v}_{\mathcal{I}}^{\hat{a}}(\alpha)$ regardless of whether α is a variable or a constant. Given the starting assumption, $\boldsymbol{v}_{\mathcal{I}}^{\hat{e}}(\alpha) = \boldsymbol{v}_{\mathcal{I}}^{\hat{a}}(\beta)$. Since β is free for α in φ, we know that $\gamma \neq \beta$. If β is a variable, then $\hat{e}(\beta) = \hat{a}(\beta)$ since $\hat{e}$ is a γ-variant of $\hat{a}$, and so $\boldsymbol{v}_{\mathcal{I}}^{\hat{a}}(\beta) = \boldsymbol{v}_{\mathcal{I}}^{\hat{e}}(\beta)$ regardless of whether β is a variable or a constant. Thus $\boldsymbol{v}_{\mathcal{I}}^{\hat{e}}(\alpha) = \boldsymbol{v}_{\mathcal{I}}^{\hat{e}}(\beta)$. As in *Case 1*, $\texttt{Comp}(\psi) \leqslant n$, and so $\mathcal{V}_{\mathcal{I}}^{\hat{e}}(\psi) = \mathcal{V}_{\mathcal{I}}^{\hat{e}}(\psi[\beta/\alpha])$ by hypothesis. Since $\hat{e}$ was any γ-variant of $\hat{a}$, it follows that $\mathcal{V}_{\mathcal{I}}^{\hat{e}}(\psi) = \mathcal{V}_{\mathcal{I}}^{\hat{e}}(\psi[\beta/\alpha])$ for all γ-variants $\hat{e}$ of $\hat{a}$, thereby establishing (†). The other biconditionals follow from the definitions and the semantics for the universal quantifier.

The cases for $\wedge, \vee, \rightarrow, \leftrightarrow$, and $\exists\gamma$ are similar and so will be left as exercises for the reader. It follows that $\mathcal{V}_{\mathcal{I}}^{\hat{a}}(\varphi) = \mathcal{V}_{\mathcal{I}}^{\hat{a}}(\varphi[\beta/\alpha])$ whenever $\boldsymbol{v}_{\mathcal{I}}^{\hat{a}}(\alpha) = \boldsymbol{v}_{\mathcal{I}}^{\hat{a}}(\beta)$ and $\texttt{Comp}(\varphi) = n+1$. Thus the lemma follows by induction. □

The proof above works by induction on complexity where the only tricky cases are for the quantifiers. In a similar manner to **Lemma 9.1**, we avoided assuming the antecedent of the claim to be proved at the outset so that the induction hypothesis took a general form. In particular, $\mathcal{V}_{\mathcal{I}}^{\hat{a}}(\varphi) = \mathcal{V}_{\mathcal{I}}^{\hat{a}}(\varphi[\beta/\alpha])$ whenever $\boldsymbol{v}_{\mathcal{I}}^{\hat{a}}(\alpha) = \boldsymbol{v}_{\mathcal{I}}^{\hat{a}}(\beta)$. Since this holds for any variable assignment $\hat{a}$, we were able to apply the induction hypothesis in order to prove (†).

The next lemma proves something similar, this time holding the wff φ fixed and varying the model. In particular, any model that agrees with $\mathcal{M}$ on all constants and predicates which occur in φ will yield the same truth-value at any given variable assignment.

Lemma 11.6 If $\mathcal{M} = \langle \mathbb{D}, \mathcal{I} \rangle$ and $\mathcal{M}' = \langle \mathbb{D}, \mathcal{I}' \rangle$ share the domain $\mathbb{D}$ where $\mathcal{I}(\mathcal{F}^n) = \mathcal{I}'(\mathcal{F}^n)$ and $\mathcal{I}(\alpha) = \mathcal{I}'(\alpha)$ for every n-place predicate $\mathcal{F}^n$ and constant α that occurs in a wff φ, then $\mathcal{V}_{\mathcal{I}}^{\hat{a}}(\varphi) = \mathcal{V}_{\mathcal{I}'}^{\hat{a}}(\varphi)$ for any variable assignment $\hat{a}$ over $\mathbb{D}$.

Proof: Assume that $\mathcal{M} = \langle \mathbb{D}, \mathcal{I} \rangle$ and $\mathcal{M}' = \langle \mathbb{D}, \mathcal{I}' \rangle$ where $\mathcal{I}(\mathcal{F}^n) = \mathcal{I}'(\mathcal{F}^n)$ and $\mathcal{I}(\alpha) = \mathcal{I}'(\alpha)$ for every n-place predicate $\mathcal{F}^n$ and constant α that occurs in φ. The proof goes by induction on the complexity of φ.

Base: Assume $\texttt{Comp}(\varphi) = 0$ where $\hat{a}$ is any variable assignment over $\mathbb{D}$. It follows that φ is either $\mathcal{F}^n\alpha_1, \ldots \alpha_n$ or $\alpha_1 = \alpha_2$. Consider the following cases:

$$\begin{aligned} \mathcal{V}_{\mathcal{I}}^{\hat{a}}(\mathcal{F}^n\alpha_1, \ldots, \alpha_n) = 1 \quad & \textit{iff} \;\; \langle v_{\mathcal{I}}^{\hat{a}}(\alpha_1), \ldots, v_{\mathcal{I}}^{\hat{a}}(\alpha_n) \rangle \in \mathcal{I}(\mathcal{F}^n) \\ (\star) \quad & \textit{iff} \;\; \langle v_{\mathcal{I}'}^{\hat{a}}(\alpha_1), \ldots, v_{\mathcal{I}'}^{\hat{a}}(\alpha_n) \rangle \in \mathcal{I}'(\mathcal{F}^n) \\ & \textit{iff} \;\; \mathcal{V}_{\mathcal{I}'}^{\hat{a}}(\mathcal{F}^n\alpha_1, \ldots, \alpha_n) = 1. \end{aligned}$$

$$\begin{aligned} \mathcal{V}_{\mathcal{I}}^{\hat{a}}(\alpha_1 = \alpha_n) = 1 \quad & \textit{iff} \;\; v_{\mathcal{I}}^{\hat{a}}(\alpha_1) = v_{\mathcal{I}}^{\hat{a}}(\alpha_n) \\ (*) \quad & \textit{iff} \;\; v_{\mathcal{I}'}^{\hat{a}}(\alpha_1) = v_{\mathcal{I}'}^{\hat{a}}(\alpha_n) \\ & \textit{iff} \;\; \mathcal{V}_{\mathcal{I}'}^{\hat{a}}(\alpha_1 = \alpha_n) = 1. \end{aligned}$$

Whereas $\mathcal{I}(\mathcal{F}^n) = \mathcal{I}'(\mathcal{F}^n)$ is immediate from the assumption, given any $1 \leqslant i \leqslant n$, observe that $v_{\mathcal{I}}^{\hat{a}}(\alpha_i) = \mathcal{I}(\alpha_i) = \mathcal{I}'(\alpha_i) = v_{\mathcal{I}'}^{\hat{a}}(\alpha_i)$ if α_i is a constant. If instead α_i is a variable, then $v_{\mathcal{I}}^{\hat{a}}(\alpha_i) = \hat{a}(\alpha_i) = v_{\mathcal{I}'}^{\hat{a}}(\alpha_i)$, thereby establishing $(\star)$ and $(*)$. It follows that $\mathcal{V}_{\mathcal{I}}^{\hat{a}}(\varphi) = \mathcal{V}_{\mathcal{I}'}^{\hat{a}}(\varphi)$ for any variable assignment $\hat{a}$ over $\mathbb{D}$ if $\texttt{Comp}(\varphi) = 0$.

Induction: Assume that if $\texttt{Comp}(\varphi) \leqslant n$, then $\mathcal{V}_{\mathcal{I}}^{\hat{a}}(\varphi) = \mathcal{V}_{\mathcal{I}'}^{\hat{a}}(\varphi)$ for all variable assignments $\hat{a}$ over $\mathbb{D}$. Letting $\texttt{Comp}(\varphi) = n+1$, there are seven cases to consider corresponding to the operators $\neg, \wedge, \vee, \rightarrow, \leftrightarrow, \forall\gamma$, and $\exists\gamma$.

Case 1: Assume $\varphi = \neg\psi$. Since $\texttt{Comp}(\varphi) = n+1$ and $\texttt{Comp}(\neg\psi) = \texttt{Comp}(\psi)+1$, it follows that $\texttt{Comp}(\psi) \leqslant n$. By hypothesis, we know $\mathcal{V}_{\mathcal{I}}^{\hat{a}}(\psi) = \mathcal{V}_{\mathcal{I}'}^{\hat{a}}(\psi)$ for all variable assignments $\hat{a}$ over $\mathbb{D}$, and so by the semantics for negation, $\mathcal{V}_{\mathcal{I}}^{\hat{a}}(\neg\psi) = \mathcal{V}_{\mathcal{I}'}^{\hat{a}}(\neg\psi)$ for all variable assignments $\hat{a}$ over $\mathbb{D}$. Equivalently, $\mathcal{V}_{\mathcal{I}}^{\hat{a}}(\varphi) = \mathcal{V}_{\mathcal{I}'}^{\hat{a}}(\varphi)$ for all variable assignments $\hat{a}$ over $\mathbb{D}$ as desired. The cases for $\wedge, \vee, \rightarrow$, and $\leftrightarrow$ are similar.

Case 6: Assume $\varphi = \forall\gamma\psi$. For the same reasons given above, $\texttt{Comp}(\psi) \leqslant n$. We may then consider the following biconditionals:

$$\begin{aligned} \mathcal{V}_{\mathcal{I}}^{\hat{a}}(\forall\gamma\psi) = 1 \quad & \textit{iff} \;\; \mathcal{V}_{\mathcal{I}}^{\hat{e}}(\psi) = 1 \text{ for all } \gamma\text{-variants } \hat{e} \text{ of } \hat{a} \\ (\dagger) \quad & \textit{iff} \;\; \mathcal{V}_{\mathcal{I}'}^{\hat{e}}(\psi) = 1 \text{ for all } \gamma\text{-variants } \hat{e} \text{ of } \hat{a} \\ & \textit{iff} \;\; \mathcal{V}_{\mathcal{I}'}^{\hat{a}}(\forall\gamma\psi) = 1. \end{aligned}$$

By hypothesis, $\mathcal{V}_{\mathcal{I}}^{\hat{e}}(\psi) = \mathcal{V}_{\mathcal{I}'}^{\hat{e}}(\psi)$ for any variable assignment $\hat{e}$, thereby establishing $(\dagger)$. The other biconditionals follow from the semantics for the universal quantifier. Thus $\mathcal{V}_{\mathcal{I}}^{\hat{a}}(\varphi) = \mathcal{V}_{\mathcal{I}'}^{\hat{a}}(\varphi)$ for all variable assignments $\hat{a}$ over $\mathbb{D}$.

Since the cases for $\wedge, \vee, \rightarrow, \leftrightarrow$, and $\exists\gamma$ are similar to those above, they will be left as exercises for the reader. It follows that $\mathcal{V}_{\mathcal{I}}^{\hat{a}}(\varphi) = \mathcal{V}_{\mathcal{I}'}^{\hat{a}}(\varphi)$ for all variable assignments $\hat{a}$ if $\texttt{Comp}(\varphi) = n+1$. Thus the lemma follows by induction. □

Although by no means surprising, the lemma above plays a crucial role in a number of the proofs below. We may now turn to consider the remaining proof rules for FOL$^=$.

11.4 FOL$^=$ Rules

By drawing on the previous lemmas, we may prove a number of much more usable results. In particular, the following lemma provides a semantic analogue for universal introduction whereby we may assert the logical consequence of a universal claim given only the logical consequence of a sufficiently arbitrary instance.

11.4.1 Universal Quantifier Rules

Lemma 11.7 For any constant β that does not occur in $\forall\alpha\varphi$ or in any sentence $\chi \in \Gamma$, if $\Gamma \vDash \varphi[\beta/\alpha]$, then $\Gamma \vDash \forall\alpha\varphi$.

Proof: Assume $\Gamma \vDash \varphi[\beta/\alpha]$ where β is a constant that does not occur in $\forall\alpha\varphi$ or in any sentence $\chi \in \Gamma$. Assume for contradiction that $\Gamma \nvDash \forall\alpha\varphi$, and so there is some model $\mathcal{M} = \langle \mathbb{D}, \mathcal{I} \rangle$ where $\mathcal{V}_{\mathcal{I}}(\chi) = 1$ for all $\chi \in \Gamma$ but $\mathcal{V}_{\mathcal{I}}(\forall\alpha\varphi) = 1$. Thus $\mathcal{V}_{\mathcal{I}}^{\hat{a}}(\forall\alpha\varphi) \neq 1$ for some v.a. $\hat{a}$. By the semantics for the universal quantifier, $\mathcal{V}_{\mathcal{I}}^{\hat{c}}(\varphi) \neq 1$ for some α-variant $\hat{c}$ of $\hat{a}$. Let $\mathcal{M}'$ be the same as $\mathcal{M}$ with the exception that $\mathcal{I}'(\beta) = \hat{c}(\alpha)$. The following biconditionals hold for every $\psi \in \Gamma$:

$$\begin{aligned} \mathcal{V}_{\mathcal{I}}(\psi) = 1 \quad & \textit{iff} \;\; \mathcal{V}_{\mathcal{I}}^{\hat{e}}(\psi) = 1 \text{ for every variable assignment } \hat{e} \\ (\star) \quad & \textit{iff} \;\; \mathcal{V}_{\mathcal{I}'}^{\hat{e}}(\psi) = 1 \text{ for every variable assignment } \hat{e} \\ & \textit{iff} \;\; \mathcal{V}_{\mathcal{I}'}(\psi) = 1. \end{aligned}$$

By construction, $\mathcal{M}$ and $\mathcal{M}'$ have the same domain $\mathbb{D}$ where $\mathcal{I}(\mathcal{F}^n) = \mathcal{I}'(\mathcal{F}^n)$ and $\mathcal{I}(\alpha) = \mathcal{I}'(\alpha)$ for every n-place predicate $\mathcal{F}^n$ and every constant $\alpha \neq \beta$. Since β does not occur in any $\psi \in \Gamma$, we know $(\star)$ follows from **Lemma 11.6**. Thus $\mathcal{V}_{\mathcal{I}}(\psi) = \mathcal{V}_{\mathcal{I}'}(\psi)$ for all $\psi \in \Gamma$, and so $\mathcal{V}_{\mathcal{I}'}(\chi) = 1$ for all $\chi \in \Gamma$. By the starting assumption, $\mathcal{V}_{\mathcal{I}'}(\varphi[\beta/\alpha]) = 1$, and so $\mathcal{V}_{\mathcal{I}'}^{\hat{g}}(\varphi[\beta/\alpha]) = 1$ for every v.a. defined over $\mathbb{D}$. It follows that $\mathcal{V}_{\mathcal{I}'}^{\hat{c}}(\varphi[\beta/\alpha]) = 1$ in particular.

Recall $\mathcal{V}_{\mathcal{I}}^{\hat{c}}(\varphi) \neq 1$ from above. Since β does not occur in $\forall\alpha(\varphi)$, it follows that β does not occur in φ, and so $\mathcal{V}_{\mathcal{I}'}^{\hat{c}}(\varphi) \neq 1$ follows by **Lemma 11.6**. However, $\hat{c}(\alpha) = \mathcal{I}'(\beta)$ where β is a constant, and so $\boldsymbol{v}_{\mathcal{I}'}^{\hat{c}}(\alpha) = \boldsymbol{v}_{\mathcal{I}'}^{\hat{c}}(\beta)$ where β is free for α in φ. Thus $\mathcal{V}_{\mathcal{I}'}^{\hat{c}}(\varphi) = \mathcal{V}_{\mathcal{I}'}^{\hat{c}}(\varphi[\beta/\alpha])$ follows from **Lemma 11.5**, and so $\mathcal{V}_{\mathcal{I}'}^{\hat{c}}(\varphi[\beta/\alpha]) \neq 1$, contradicting the above. We may then conclude that $\Gamma \vDash \forall\alpha\varphi$. □

Given the lemma above, it easy to prove that the universal introduction proof rule preserves logical consequence in a similar manner to proof rules above. Consider the following proof.

Rule 13 (∀I) $\Gamma_{n+1} \vDash \varphi_{n+1}$ if φ_{n+1} follows from Γ_{n+1} by the rule ∀I.

Proof: Assume that φ_{n+1} follows from Γ_{n+1} by universal introduction $\forall$I. Thus there is some $i \leqslant n$ where $\varphi_i = \varphi[\beta/\alpha]$ is live at $n+1$ and β does not occur in $\varphi_{n+1} = \forall\alpha\varphi$ or in undischarged assumptions in Γ_{n+1}. By **Lemma 4.3**, $\Gamma_i \subseteq \Gamma_{n+1}$ where $\Gamma_i \vDash \varphi_i$ by hypotheses, and so $\Gamma_{n+1} \vDash \varphi_i$ by **Lemma 2.1**. Equivalently, $\Gamma_{n+1} \vDash \varphi[\beta/\alpha]$. Since β does not occur in $\forall\alpha\varphi$ or any undischarged assumptions in Γ_{n+1}, it follows by **Lemma 11.7** that $\Gamma_{n+1} \vDash \forall\alpha\varphi$, and so $\Gamma_{n+1} \vDash \varphi_{n+1}$. □

This proof amounts to little more than an application of **Lemma 11.7**. In particular, there is no mention of the semantics for the universal quantifiers in the proof of **Rule 13** since all of these details are already contained in the supporting lemma. The following lemma will play an analogous role for universal elimination.

Lemma 11.8 $\forall\alpha\varphi \vDash \varphi[\beta/\alpha]$ where α is a variable and $\varphi[\beta/\alpha]$ is a wfs of $\mathcal{L}^=$.

Proof: Let $\mathcal{M} = \langle \mathbb{D}, \mathcal{I} \rangle$ be any model where $\mathcal{V}_{\mathcal{I}}(\forall\alpha\varphi) = 1$. Thus $\mathcal{V}_{\mathcal{I}}^{\hat{a}}(\forall\alpha\varphi) = 1$ for every v.a. $\hat{a}$ defined over $\mathbb{D}$, and so $\mathcal{V}_{\mathcal{I}}^{\hat{c}}(\varphi) = 1$ for every α-variant $\hat{c}$ of $\hat{a}$ by the semantics for the universal quantifier. Letting $\hat{e}$ be an α-variant of $\hat{a}$ where $\hat{e}(\alpha) = \mathcal{I}(\beta)$, it follows that $\boldsymbol{v}_{\mathcal{I}}^{\hat{e}}(\alpha) = \boldsymbol{v}_{\mathcal{I}}^{\hat{e}}(\beta)$. Since there are no free variables in $\varphi[\beta/\alpha]$, we know that β is a constant, and so β is free for α in φ. Thus $\mathcal{V}_{\mathcal{I}}^{\hat{e}}(\varphi) = \mathcal{V}_{\mathcal{I}}^{\hat{e}}(\varphi[\beta/\alpha])$ follows by **Lemma 11.5**, and so $\mathcal{V}_{\mathcal{I}}(\varphi[\beta/\alpha]) = 1$ by **Lemma 9.2** since $\varphi[\beta/\alpha]$ is a wfs of $\mathcal{L}^=$. It follows that $\forall\alpha\varphi \vDash \varphi[\beta/\alpha]$. □

Whereas **Lemma 11.7** made use of the particular constraints that must hold for the universal introduction rule to be applied, the lemma above is much less constrained. This corresponds to the fact that universal claims entail all of their substitution instances.

We now turn to provide another supporting lemma which will help further streamline the proof for the universal elimination rule as well as a number of other proofs below.

Lemma 11.9 If $\Gamma \vDash \varphi$ and $\Sigma \cup \{\varphi\} \vDash \psi$, then $\Gamma \cup \Sigma \vDash \psi$.

Proof: Assume $\Gamma \vDash \varphi$ and $\Sigma \cup \{\varphi\} \vDash \psi$. Let $\mathcal{M} = \langle \mathbb{D}, \mathcal{I} \rangle$ be a model where $\mathcal{V}_{\mathcal{I}}(\chi) = 1$ for all $\chi \in \Gamma \cup \Sigma$. Since $\mathcal{V}_{\mathcal{I}}(\chi) = 1$ for all $\chi \in \Gamma$, we know by the starting assumptions that $\mathcal{V}_{\mathcal{I}}(\varphi) = 1$. Since $\mathcal{V}_{\mathcal{I}}(\chi) = 1$ for all $\chi \in \Sigma$, it follows that $\mathcal{V}_{\mathcal{I}}(\chi) = 1$ for all $\chi \in \Sigma \cup \{\varphi\}$, and so $\mathcal{V}_{\mathcal{I}}(\psi) = 1$ follows be the starting assumptions. Generalizing on $\mathcal{M}$, we may conclude that $\Gamma \cup \Sigma \vDash \psi$. □

The proof above is a semantic analogue of a metarule that goes by the name 'Cut' since it allows us to cut out intermediaries. This will play a helpful role in the following proof.

Rule 14 ($\forall$E) $\Gamma_{n+1} \vDash \varphi_{n+1}$ if φ_{n+1} follows from Γ_{n+1} by the rule $\forall$E.

Proof: Assume that φ_{n+1} follows from Γ_{n+1} by universal elimination $\forall$E. Thus there is some $i \leqslant n$ where $\varphi_i = \forall\alpha\varphi$ is live at $n+1$ and $\varphi_{n+1} = \varphi[\beta/\alpha]$ for some variable α and constant β. By **Lemma 4.3**, $\Gamma_i \subseteq \Gamma_{n+1}$ where $\Gamma_i \vDash \varphi_i$ by hypotheses, and so $\Gamma_{n+1} \vDash \varphi_i$ by **Lemma 2.1**. Equivalently, $\Gamma_{n+1} \vDash \forall\alpha\varphi$. By **Lemma 11.8** that $\forall\alpha\varphi \vDash \varphi[\beta/\alpha]$, and so $\Gamma_{n+1} \vDash \varphi_{n+1}$ by **Lemma 11.9**. $\square$

This proof turns on **Lemma 11.8** where the other lemmas only play a supporting role.

11.4.2 Existential Quantifier Rules

Just as universal elimination is an easier rule to apply with fewer constraints, something similar may be said for existential introduction. Nevertheless, the following lemma will help to show that the proof rule for existential introduction preserves logical consequence.

Lemma 11.10 $\varphi[\beta/\alpha] \vDash \exists\alpha\varphi$ where α is a variable and $\varphi[\beta/\alpha]$ is a wfs of $\mathcal{L}^=$.

Proof: Let $\mathcal{M} = \langle \mathbb{D}, \mathcal{I} \rangle$ be a model where $\mathcal{V}_{\mathcal{I}}(\varphi[\beta/\alpha]) = 1$, and so $\mathcal{V}_{\mathcal{I}}^{\hat{a}}(\varphi[\beta/\alpha]) = 1$ for every v.a. $\hat{a}$ defined over $\mathbb{D}$. Letting $\hat{c}$ be a v.a. where $\hat{c}(\alpha) = \mathcal{I}(\beta)$, it follows that $v_{\mathcal{I}}^{\hat{c}}(\alpha) = v_{\mathcal{I}}^{\hat{c}}(\beta)$ where β is free for α in φ, and so $\mathcal{V}_{\mathcal{I}}^{\hat{c}}(\varphi[\beta/\alpha]) = \mathcal{V}_{\mathcal{I}}^{\hat{c}}(\varphi)$ by **Lemma 11.5**. By the semantics for the existential quantifier, $\mathcal{V}_{\mathcal{I}}^{\hat{c}}(\exists\alpha\varphi) = 1$ since $\hat{c}$ is an α-variant of itself. Since $\varphi[\beta/\alpha]$ is a wfs $\mathcal{L}^=$, at most α is free in φ, and so $\exists\alpha\varphi$ is a wfs. Hence $\mathcal{V}_{\mathcal{I}}(\exists\alpha\varphi) = 1$ by **Lemma 9.2**, and so $\varphi[\beta/\alpha] \vDash \exists\alpha\varphi$. $\square$

This lemma follows easily from **Lemma 11.5** where most of the work was already accomplished save for one critical appeal to the semantics for the existential quantifier. We may now turn to provide a proof for the existential introduction rule given below:

Rule 15 ($\exists$I) $\Gamma_{n+1} \vDash \varphi_{n+1}$ if φ_{n+1} follows from Γ_{n+1} by the rule $\exists$I.

Proof: Assume that φ_{n+1} follows from Γ_{n+1} by existential introduction $\exists$I. Thus there is some $i \leqslant n$ where $\varphi_i = \varphi[\beta/\alpha]$ is live at $n+1$ and $\varphi_{n+1} = \exists\alpha\varphi$ for some variable α and constant β. By **Lemma 4.3**, $\Gamma_i \subseteq \Gamma_{n+1}$ where $\Gamma_i \vDash \varphi_i$ by hypotheses, and so $\Gamma_{n+1} \vDash \varphi_i$ by **Lemma 2.1**. Equivalently, $\Gamma_{n+1} \vDash \varphi[\beta/\alpha]$. Since $\varphi[\beta/\alpha] \vDash \exists\alpha\varphi$ by **Lemma 11.10**, $\Gamma_{n+1} \vDash \varphi_{n+1}$ by **Lemma 11.9**. $\square$

Like the proof for universal elimination, this proof amounts to little more than an application of **Lemma 11.10** where most of the work was already completed there. Whereas universal elimination and existential introduction are easy to apply and relatively unconstrained, the existential elimination rule is much more restricted. Accordingly, the following lemma makes use of these restrictions in order to establish a semantic analogue of the existential elimination rule in a similar manner to the supporting lemma for universal introduction.

Lemma 11.11 For any constant β that does not occur in $\exists\alpha\varphi$, ψ, or in any sentence $\chi \in \Gamma$, if $\Gamma \vDash \exists\alpha\varphi$ and $\Gamma \cup \{\varphi[\beta/\alpha]\} \vDash \psi$, then $\Gamma \vDash \psi$.

Proof: Assume $\Gamma \vDash \exists\alpha\varphi$ and $\Gamma \cup \{\varphi[\beta/\alpha]\} \vDash \psi$ where β is a constant that does not occur in $\exists\alpha\varphi$, ψ, or in any sentence $\chi \in \Gamma$. Let $\mathcal{M} = \langle \mathbb{D}, \mathcal{I} \rangle$ be a model where $\mathcal{V}_{\mathcal{I}}(\chi) = 1$ for all $\chi \in \Gamma$. It follows that $\mathcal{V}_{\mathcal{I}}(\exists\alpha\varphi) = 1$, and so $\mathcal{V}_{\mathcal{I}}^{\hat{a}}(\exists\alpha\varphi) = 1$ for some v.a. $\hat{a}$ defined over $\mathbb{D}$ by **Lemma 9.2**. Thus $\mathcal{V}_{\mathcal{I}}^{\hat{c}}(\varphi) = 1$ for some α-variant $\hat{c}$ of $\hat{a}$ by the semantics for the existential quantifier.

Let $\mathcal{M}'$ be the same as $\mathcal{M}$ with the only possible exception being that $\mathcal{I}'(\beta) = \hat{c}(\alpha)$ so that $\boldsymbol{v}_{\mathcal{I}'}^{\hat{c}}(\beta) = \boldsymbol{v}_{\mathcal{I}'}^{\hat{c}}(\alpha)$. Letting $\chi \in \Gamma$, we know that $\mathcal{V}_{\mathcal{I}}(\chi) = 1$, and so $\mathcal{V}_{\mathcal{I}}^{\hat{e}}(\chi) = 1$ for some v.a. $\hat{e}$ by **Lemma 9.2**. Given the assumptions about β, it follows from **Lemma 11.6** that $\mathcal{V}_{\mathcal{I}'}^{\hat{e}}(\chi) = 1$, and so $\mathcal{V}_{\mathcal{I}'}(\chi) = 1$ again by **Lemma 9.2**. By generalizing on χ, we may conclude that $\mathcal{V}_{\mathcal{I}'}(\chi) = 1$ for all $\chi \in \Gamma$.

Since β does not occur in $\exists\alpha\varphi$, it follows that β does not occur in φ, and so $\mathcal{V}_{\mathcal{I}}^{\hat{c}}(\varphi) = \mathcal{V}_{\mathcal{I}'}^{\hat{c}}(\varphi)$ by **Lemma 11.6**. Moreover, $\boldsymbol{v}_{\mathcal{I}'}^{\hat{c}}(\beta) = \boldsymbol{v}_{\mathcal{I}'}^{\hat{c}}(\alpha)$ where β is free for α in φ on account of being a constant, and so $\mathcal{V}_{\mathcal{I}'}^{\hat{c}}(\varphi) = \mathcal{V}_{\mathcal{I}'}^{\hat{c}}(\varphi[\beta/\alpha])$ by **Lemma 11.5**. Given the identities above, $\mathcal{V}_{\mathcal{I}'}^{\hat{c}}(\varphi[\beta/\alpha]) = 1$, and so $\mathcal{V}_{\mathcal{I}'}(\varphi[\beta/\alpha]) = 1$ since $\varphi[\beta/\alpha]$ is a wfs of $\mathcal{L}^=$. Thus $\mathcal{V}_{\mathcal{I}'}(\chi) = 1$ for all $\chi \in \Gamma \cup \{\varphi[\beta/\alpha]\}$.

It follows by the starting assumption that $\mathcal{V}_{\mathcal{I}'}(\psi) = 1$, and so $\mathcal{V}_{\mathcal{I}'}^{\hat{g}}(\psi) = 1$ for every v.a. $\hat{g}$ defined over $\mathbb{D}$. Since β does not occur in ψ, we may conclude by **Lemma 11.6** that $\mathcal{V}_{\mathcal{I}}^{\hat{g}}(\psi) = 1$ for every v.a. $\hat{g}$ defined over $\mathbb{D}$. Thus $\mathcal{V}_{\mathcal{I}}(\psi) = 1$, and so $\Gamma \vDash \psi$ follows by generalizing on $\mathcal{M}$. □

Given a model $\mathcal{M}$ which makes all of the premises in Γ, it follows that $\exists\alpha\varphi$ is true in $\mathcal{M}$ on some variable assignment $\hat{c}$ by **Lemma 9.2**. The proof then draws on **Lemma 11.6** in order to introduce a model variant $\mathcal{M}'$ which assigns the constant β to whatever the variable α happens to be assigned by $\hat{c}$. The variable α in the wff φ is then replaced with β where **Lemma 11.5** is used to show that the truth-value of $\varphi[\beta/\alpha]$ remains unaffected in the model variant and variable assignment in question.

Since β does not occur in the premises, the premises are also true on the model variant, and since $\varphi[\beta/\alpha]$ is true in the model variant, ψ is true in in the model variant given the starting assumption. Since β does not occur in the conclusion ψ, we may conclude by **Lemma 11.6** that ψ is true in the original model $\mathcal{M}$. Generalizing on $\mathcal{M}$ completes the proof.

Given the previous lemma, we may proceed to show that the proof rule for existential elimination preserves logical consequence as desired.

Rule 16 (∃E) $\Gamma_{n+1} \vDash \varphi_{n+1}$ if φ_{n+1} follows from Γ_{n+1} by the rule ∃E.

Proof: Assume that φ_{n+1} follows from Γ_{n+1} by existential elimination ∃E. Thus there is some $i < j < k \leqslant n$ where $\varphi_i = \exists\alpha\varphi$ is live at $n+1$, $\varphi_j = \varphi[\beta/\alpha]$ for some constant β that does not occur in φ_i, φ_k, or any $\psi \in \Gamma_i$. Thus we have:

$$
\begin{array}{l|lll}
i & \exists\alpha\varphi & & \\
j & & \left|\ \underline{\varphi[\beta/\alpha]}\right. & \text{:AS for } \exists\text{E} \\
 & & \left|\ \vdots\right. & \\
k & & \left|\ \psi\right. & \\
n+1 & \psi & & :i,\ j\text{–}k\ \exists\text{E}
\end{array}
$$

By hypothesis, $\Gamma_i \vDash \varphi_i$ and $\Gamma_k \vDash \varphi_k$ where $\Gamma_i \subseteq \Gamma_{n+1}$ by **Lemma 4.3**. With the exception of φ_j, every assumption that is undischarged at line k is also undischarged at line $n+1$, and so $\Gamma_k \subseteq \Gamma_{n+1} \cup \{\varphi_j\}$. It follows by **Lemma 2.1** that $\Gamma_{n+1} \vDash \varphi_i$ and $\Gamma_{n+1} \cup \{\varphi_j\} \vDash \varphi_k$, and so $\Gamma_{n+1} \vDash \exists\alpha\varphi$ and $\Gamma_{n+1} \cup \{\varphi[\beta/\alpha]\} \vDash \psi$. Thus $\Gamma_{n+1} \vDash \psi$ by **Lemma 11.11**, and so $\Gamma_{n+1} \vDash \varphi_{n+1}$. □

Since **Lemma 11.11** already does most of the heavy lifting, the proof above is the result of carefully setting up a generic scenario in which the existential elimination rule is applied, using the lemmas cited above to draw out the resulting consequences.

11.4.3 Identity Rules

Recall from the proof of **Lemma 11.1** that we have already considered identity introduction in the case of a one line proof. All that remains is to generalize this proof to the present setting where the $n+1$ line is the result of identity introduction.

Rule 17 (=I) $\Gamma_{n+1} \vDash \varphi_{n+1}$ if φ_{n+1} follows from Γ_{n+1} by the rule =I.

Proof: Assume that φ_{n+1} follows from Γ_{n+1} by existential introduction =I. Thus φ_{n+1} is $\alpha = \alpha$ for some constant α. Letting $\mathcal{M} = \langle \mathbb{D}, \mathcal{I} \rangle$ be any model, it follows that $\mathcal{I}(\alpha) = \mathcal{I}(\alpha)$, and so $\boldsymbol{v}_{\mathcal{I}}^{\hat{a}}(\alpha) = \boldsymbol{v}_{\mathcal{I}}^{\hat{a}}(\alpha)$ for any variable assignment $\hat{a}$. By the semantics for identity, $\mathcal{V}_{\mathcal{I}}^{\hat{a}}(\alpha = \alpha) = 1$, and so $\vDash \alpha = \alpha$ by generalizing on $\mathcal{M}$. Equivalently $\vDash \varphi_{n+1}$, and so $\Gamma_{n+1} \vDash \varphi_{n+1}$ follows by **Lemma 2.1**. □

The proof above follows the same line of reasoning given in **Lemma 11.1**. In order to provide a proof for identity elimination, the following lemma establishes a semantic analogue of the identity elimination rule where this proof will draw on the substitution lemma given above.

Lemma 11.12 If α and β are constants, then $\varphi[\alpha/\gamma], \alpha = \beta \vDash \varphi[\beta/\gamma]$.

Proof: Let $\mathcal{M} = \langle \mathbb{D}, \mathcal{I} \rangle$ be a model where $\mathcal{V}_{\mathcal{I}}(\varphi[\alpha/\gamma]) = \mathcal{V}_{\mathcal{I}}(\alpha = \beta) = 1$ where α and β are both constants. By **Lemma 9.2**, $\mathcal{V}_{\mathcal{I}}^{\hat{a}}(\varphi[\alpha/\gamma]) = 1$ for some variable assignment $\hat{a}$ over $\mathbb{D}$, where $\mathcal{V}_{\mathcal{I}}^{\hat{c}}(\alpha = \beta) = 1$ for all variable assignments $\hat{c}$ over $\mathbb{D}$, and so $\mathcal{V}_{\mathcal{I}}^{\hat{a}}(\alpha = \beta) = 1$ in particular. Thus $\boldsymbol{v}_{\mathcal{I}}^{\hat{a}}(\alpha) = \boldsymbol{v}_{\mathcal{I}}^{\hat{a}}(\beta)$.

Since β is a constant, β is free for α in $\varphi[\alpha/\gamma]$, and so it follows by **Lemma 11.5** that $\mathcal{V}_{\mathcal{I}}^{\hat{a}}(\varphi[\alpha/\gamma]) = \mathcal{V}_{\mathcal{I}}^{\hat{a}}((\varphi[\alpha/\gamma])[\beta/\alpha])$. However, $(\varphi[\alpha/\gamma])[\beta/\alpha] = \varphi[\beta/\gamma]$, and so $\mathcal{V}_{\mathcal{I}}^{\hat{a}}(\varphi[\beta/\gamma]) = 1$. Since $\varphi[\beta/\gamma]$ is a wfs of $\mathcal{L}^=$, $\mathcal{V}_{\mathcal{I}}(\varphi[\beta/\gamma]) = 1$. By generalizing on $\mathcal{M}$ we may conclude that $\varphi[\alpha/\gamma], \alpha = \beta \vDash \varphi[\beta/\gamma]$. □

This lemma amounts to little more than an application of **Lemma 11.5** together with the observation that $(\varphi[\alpha/\gamma])[\beta/\alpha] = \varphi[\beta/\gamma]$. We may then provide the following proof:

Rule 18 (=E) $\Gamma_{n+1} \vDash \varphi_{n+1}$ if φ_{n+1} follows from Γ_{n+1} by the rule =E.

Proof: Assume that φ_{n+1} follows from Γ_{n+1} by existential elimination =E. Thus there are some live lines $i, j \leqslant n$ at $n+1$ where φ_i is $\alpha = \beta$ for some constants α and β and either $\varphi_j = \varphi[\alpha/\gamma]$ and $\varphi_{n+1} = \varphi[\beta/\gamma]$ or else $\varphi_j = \varphi[\beta/\gamma]$ and $\varphi_{n+1} = \varphi[\beta/\gamma]$. By parity of reasoning, we may assume that $\varphi_j = \varphi[\alpha/\gamma]$ and $\varphi_{n+1} = \varphi[\beta/\gamma]$ which we may represent as follows:

i	$\alpha = \beta$	
j	$\varphi[\alpha/\gamma]$	
$n+1$	$\varphi[\beta/\gamma]$	$:i,\ j$ =E

By **Lemma 4.3**, $\Gamma_i, \Gamma_j \subseteq \Gamma_{n+1}$ where $\Gamma_i \vDash \varphi_i$ and $\Gamma_j \vDash \varphi_j$ by hypotheses, and so $\Gamma_{n+1} \vDash \varphi_i$ and $\Gamma_{n+1} \vDash \varphi_j$ by **Lemma 2.1**. Equivalently, $\Gamma_{n+1} \vDash \alpha = \beta$ and

$\Gamma_{n+1} \vDash \varphi[\alpha/\gamma]$. Since α and β are constants, we know by **Lemma 11.12** that $\varphi[\alpha/\gamma], \alpha = \beta \vDash \varphi[\beta/\gamma]$. By two applications of **Lemma 11.9**, we may conclude that $\Gamma_{n+1} \vDash \varphi[\beta/\gamma]$, or equivalently, $\Gamma_{n+1} \vDash \varphi_{n+1}$. □

Since **Lemma 11.12** does most of the work above and **Lemma 11.5** made it easy to prove **Lemma 11.12**, identity elimination can be viewed as an application of **Lemma 11.5**. Put otherwise, **Lemma 11.5** is what explains why the identity elimination rule preserves validity.

11.5 Conclusion

Given PL Rules together with **Rule 13** – **Rule 18**, we may now assert the following:

FOL$^=$ Rules: If $\Gamma_k \vDash \varphi_k$ for every $k \leqslant n$ and φ_{n+1} follows by the proof rules for FOL$^=$, then $\Gamma_{n+1} \vDash \varphi_{n+1}$.

Having done most of work required, we are now in a position to establish the induction lemma cited in the proof of FOL$^=$ Soundness above.

Lemma 11.13 (Induction Step) $\Gamma_{n+1} \vDash \varphi_{n+1}$ if $\Gamma_k \vDash \varphi_k$ for every $k \leqslant n$.

Assume that $\Gamma_k \vDash \varphi_k$ for every $k \leqslant n$. It remains to show that $\Gamma_{n+1} \vDash \varphi_{n+1}$. By the definition of a proof in FOL$^=$, we know that φ_{n+1} is either a premise or follows by one of the proof rules for FOL$^=$. If φ_{n+1} is a premise, then $\varphi_{n+1} \in \Gamma_{n+1}$ and so $\Gamma_{n+1} \vDash \varphi_{n+1}$ is immediate. If φ_{n+1} follows from the previous lines by one of the proof rules for FOL$^=$, then given our starting assumption that $\Gamma_k \vDash \varphi_k$ for every $k \leqslant n$, it follows from FOL$^=$ Rules that $\Gamma_{n+1} \vDash \varphi_{n+1}$. Thus $\Gamma_{n+1} \vDash \varphi_{n+1}$ in either case. Discharging our assumption completes the proof. □

Not only does the soundness of FOL$^=$ tell us that we can rely on our natural deduction systems in order to construct valid arguments in which the conclusion is a logical consequence of the premises, soundness begins to close the gap between two very different approaches to logic. Whereas the logical consequence relation $\vDash$ for $\mathcal{L}^=$ describes what follows from what in virtue of logical form by quantifying over all models of $\mathcal{L}^=$, the derivation relation $\vdash$ aims to directly encode natural patterns of reasoning in $\mathcal{L}^=$. What soundness shows is that our purely proof-theoretic descriptions of logical reasoning in $\mathcal{L}^=$ does not diverge from our model-theoretic descriptions of logical reasoning in $\mathcal{L}^=$.

In the following chapter, we will consider the converse, showing that in addition to being sound, FOL$^=$ is also complete. By contrast with soundness which one might insist any proof system must satisfy over a reasonable semantics, completeness is a powerful and deeply surprising result. As before, our approach will be to build on what we have already established.

Chapter 12

The Completeness of $\text{FOL}^=$

12.1 Introduction

Consider the calculator from before that can compute basic arithmetical operations. If that calculator sometimes gave false answers we probably shouldn't call it a calculator at all. Although it may often given the right results, there would be no way to know if it was giving us the right result or not, and so could not be relied upon. Put otherwise, the calculator is not *sound* with respect to the truths of arithmetic since some of its answers are false. For an analogous reason, it was important to show that $\text{FOL}^=$ was sound over the semantics for $\mathcal{L}^=$ so that we could rely on $\text{FOL}^=$ to conduct valid reasoning. If φ is derivable from Γ, we know by soundness that φ is also a logical consequence of Γ. Put formally: if $\Gamma \vdash \varphi$, then $\Gamma \vDash \varphi$.

Completeness asserts the converse so that we may conclude that φ is derivable from Γ whenever φ is a logical consequence of Γ, or more compactly: if $\Gamma \vDash \varphi$, then $\Gamma \vdash \varphi$. You might recall that our calculator from before is not complete. In fact, no calculator is complete for purely material reasons: no matter how much memory a calculator may have, there are numbers big enough that will exhaust its memory. For instance, raising one large number to another large number will quickly use up the memory. As you may have observed in elementary school, there are some arithmetical operations that the calculator simply cannot compute, yielding 'ERROR' as a result. So long as the calculator doesn't spit out any false answers— i.e., it is sound— it is still be of considerable use despite its incompleteness.

A separate question is whether, in principle, there could be an effective procedure which yields the right answers to any arithmetical operations. By 'effective procedure' we do not mean a material computing device but rather an abstract method which could be fully specified with precise rules which one could in principle follow to compute the result of any arithmetical operations. It turns out that there is no such effective procedure for arithmetic. Put otherwise, arithmetic is incomplete. $\text{FOL}^=$ does not share this same fate. Rather, we will show in the following section that whenever a conclusion is a logical consequence of some premises, that conclusion is derivable from those premises in $\text{FOL}^=$.

Much will be as before where instead of beginning with $\Gamma \vDash \varphi$ as an assumption and arguing to the conclusion $\Gamma \vdash \varphi$, we will focus on establishing a closely related result:

Theorem 12.1 Every consistent set of $\mathcal{L}^=$ sentences Γ is satisfiable.

Recall that a set of $\mathcal{L}^=$ sentences Γ is inconsistent if $\bot$ is derivable from Γ, and consistent otherwise. Assuming $\Gamma \vDash \varphi$, we know that $\Gamma \cup \{\neg\varphi\}$ is unsatisfiable by **Lemma 2.3**, and so inconsistent by **Theorem 12.1**. It follows that $\Gamma \cup \{\neg\varphi\} \vdash \bot$, and so we may derive a contradiction from $\Gamma \cup \{\neg\varphi\}$ in FOL$^=$. Given Γ as premises and $\neg\varphi$ as an assumption, it follows by negation elimination that $\Gamma \vdash \varphi$, establishing completeness (see **Corollary 12.1**).

It remains to establish **Theorem 12.1**. The proof will proceed in a number of stages. Whereas PL only concerned the wfs of $\mathcal{L}^{\text{PL}}$ which build up complex sentences from sentence letters and the truth-functional operators, we must now take into consideration the wffs that can be constructed from predicates, constants, and variables, building up more complex wffs with the truth-functional operators and quantifiers which bind the variables that fall within their scope. Accordingly, we will extend our language to include a countably infinite number of new constants. For simplicity, we will use the set of natural numbers $\mathbb{N}$, calling our new language $\mathcal{L}^=_{\mathbb{N}}$. Just as we added a countably infinite number of constants $\zeta_0, \zeta_1, \ldots$ for each lower case letter $\zeta \in \{a, b, \ldots, t\}$, we have effectively added one more countably infinite stock of constants to use to build sentences. Assuming that Γ is consistent in $\mathcal{L}^=$, we will show in §12.2.1 that Γ is also consistent in $\mathcal{L}^=_{\mathbb{N}}$. This completes the first stage of the proof.

You might be wondering what difference a few extra constants will make to our language given how many constants we had to begin with. The reason for this addition is that we would like to extend Γ to a bigger set of sentences which is guaranteed to include instances of every existential claim. We will refer to these instances as *witnesses*. That is, if $\exists x Fxa$ is a sentence in Γ, then we would like to include an instance such as $F1a$ where our instantiating constant is guaranteed not to conflict with anything else in Γ. An easy way to do this is to draw on our set of new constants. Since Γ is still the same set of wfss from $\mathcal{L}^=$, none of our new constants occur in any sentence in Γ. For instance 1 does not occur in any sentence in Γ, and so our instance $F1a$ will not conflict with what belongs to Γ. Extending Γ to include witnesses for all existential claims constitutes the second stage of the proof where we will refer to this larger set Σ_Γ as *saturated*. As we will show in §12.2.2, Σ_Γ is also consistent.

The next stage will be familiar from before where we will extend Σ_Γ to includes every sentence or its negation but not both, calling this *maximal* set Δ_{Σ_Γ}, or just Δ for short. We will show in §12.2.3 that Δ is consistent, where it follows that Δ is *deductively closed* insofar as it contains every sentence that is derivable from Δ. Deductive closure is a very important and convenient property which will play a critical role in the later stages of the proof.

Having extended Γ to a much bigger set of sentences Δ that is saturated, maximal, consistent, and deductively closed, we will proceed to use this set to construct a model that shows Δ is satisfiable, and so Γ is satisfiable as a result. Accordingly, it will be convenient to say that a model $\mathcal{M}$ SATISFIES a set Λ of wfss just in case $\mathcal{V}_{\mathcal{I}}(\chi) = 1$ for all $\chi \in \Lambda$.

As in the proof of PL Completeness, we will use the set of wfss Γ to build a model that satisfies Γ. However strange it may be to use the symbols that make up the syntax of a language to provide a model of that language, there is no circularity. Rather, to interpret a first-order language like $\mathcal{L}^=$ and $\mathcal{L}^=_{\mathbb{N}}$, all we need is a domain where any nonempty set will suffice. In what follows, we will interpret $\mathcal{L}^=_{\mathbb{N}}$ over the domain $\mathbb{D}_\Delta$ whose members are sets of constants in our extended language $\mathcal{L}^=_{\mathbb{N}}$ which we will define below. Although these might be obscure objects to think about, nothing prevents us from using them to interpret $\mathcal{L}^=_{\mathbb{N}}$.

Given $\mathbb{D}_\Delta$, we will specify referents and extensions for the constants and predicates of our language in such a way that the resulting model $\mathcal{M}_\Delta$ satisfies all and only the sentences that belong to Δ. As a result, $\mathcal{M}_\Delta$ satisfies Δ, and since $\Gamma \subseteq \Delta$, we may conclude that Γ is satisfiable. As before, we will refer to this cleverly constructed model $\mathcal{M}_\Delta$ as a Henkin model after Leon Henkin who developed this proof strategy in 1949.

This provides a rough overview of the proof strategy that will be deployed below. If you find that you get lost along the way, it can help to return to this overview to regain your bearings and keep track of what is happening and why. Slogging on in the dark is rarely advisable.

12.2 Extensions

Assume Γ is a consistent set of $\mathcal{L}^=$ sentences. We will maintain this assumption throughout the following sections in order to show that Γ is satisfiable. The following section begins by constructing an extension of Γ called Δ— i.e., where $\Gamma \subseteq \Delta$— which we will show is saturated, maximal, and consistent. These properties will enable us to show that Δ is deductively closed insofar as $\varphi \in \Delta$ whenever $\Delta \vdash \varphi$. Deductive closure will play a critical role in constructing a Henkin model $\mathcal{M}_\Delta$ which satisfies Δ, and so satisfies Γ as a consequence.

12.2.1 Witnesses

Let $\mathcal{L}^=_{\mathbb{N}}$ be a language like $\mathcal{L}^=$ except for including the natural numbers $\mathbb{N}$ as an additional set of constants. Even though Γ is consistent in $\mathcal{L}^=$, it does not immediately follow that Γ is consistent in $\mathcal{L}^=_{\mathbb{N}}$. In general, adding expressive resources to a language can provide the grounds for new derivations and so we need to check that a contradiction cannot be derived in $\mathcal{L}^=_{\mathbb{N}}$ from Γ. In order to rule out this possibility out, we will prove the following lemma. Assuming that β is free for α in every line of a proof X, it will be convenient to take $X[\beta/\alpha]$ to be the result of replacing β with α in every line of X.

Lemma 12.1 If α is a constant and X is an FOL$^=$ derivation in which the constant β does not occur, then $X[\beta/\alpha]$ is also an FOL$^=$ derivation.

Proof: This proof is left as an exercise for the reader. □

Whereas the rules of FOL$^=$ were defined for the wfss of $\mathcal{L}^=$, they may just as easily be defined for $\mathcal{L}^=_{\mathbb{N}}$, referring to the resulting proof system as FOL$^=_{\mathbb{N}}$. Recalling the definition of a FOL$^=$ DERIVATION from Chapter 10, we may similarly define the analogue for FOL$^=_{\mathbb{N}}$ to be just like it was before while drawing on the wider range of wfss of $\mathcal{L}^=_{\mathbb{N}}$.

> A DERIVATION (or PROOF) of φ from Γ in FOL$^=_{\mathbb{N}}$ is any finite sequence of wfss of $\mathcal{L}^=_{\mathbb{N}}$ ending in φ where every wfs in the sequence is either: (1) a premise in Γ; (2) an assumption which is eventually discharged; or (3) follows from previous lines by a natural deduction rule for FOL$^=_{\mathbb{N}}$ besides AS.

Recall that consistency was relative to a proof system. Whereas a set of wfss Γ of $\mathcal{L}^=$ is CONSISTENT in FOL$^=$ just in case $\Gamma \nvdash \bot$ in FOL$^=$, we may also say that a set of wfss Γ of $\mathcal{L}^=_{\mathbb{N}}$ is CONSISTENT in FOL$^=_{\mathbb{N}}$ just in case $\Gamma \nvdash \bot$ in FOL$^=_{\mathbb{N}}$. We may then prove the following.

Lemma 12.2 If Γ is consistent in FOL$^=$, then Γ is also consistent in FOL$^=_{\mathbb{N}}$.

Proof: Assume that Γ is a consistent in FOL$^=$. Assume for contradiction that Γ is inconsistent in FOL$^=_{\mathbb{N}}$, and so $\Gamma \vdash A \wedge \neg A$ in FOL$^=_{\mathbb{N}}$. Thus there is a derivation X of $A \wedge \neg A$ from Γ where every line of X is a wfs of $\mathcal{L}^=_{\mathbb{N}}$. Since every proof is finite, there are at most finitely many constants that occur in X, and so at most finitely many constants in X that belong to $\mathbb{N}$.

Letting $\vec{n} = \langle n_1, \ldots, n_m \rangle$ include all constants in $\mathbb{N}$ that occur in X, we may take $\vec{\alpha} = \langle \alpha_1, \ldots, \alpha_m \rangle$ to be a sequence of $\mathcal{L}^=$ constants where α_i is the i^{th} $\mathcal{L}^=$ constant not to occur in X. By defining $X[\vec{\alpha}/\vec{n}] := X[\alpha_1/n_1]\ldots[\alpha_m/n_m]$ to be the result of substituting α_i for n_i in for all $1 \leqslant i \leqslant n$ in every line of X, it follows by m applications of **Lemma 12.1** that $X[\vec{\alpha}/\vec{n}]$ is a proof of $(A \wedge \neg A)[\vec{\alpha}/\vec{n}]$ from $\Gamma[\vec{\alpha}/\vec{n}]$ in FOL$^=$ since every line in $X[\vec{\alpha}/\vec{n}]$ is a $\mathcal{L}^=$ wfs.

Since Γ is consistent in FOL$^=$, we know that ψ is a $\mathcal{L}^=$ wfs for every $\psi \in \Gamma$, and so $\Gamma[\vec{\alpha}/\vec{n}] = \Gamma$. Similarly, $(A \wedge \neg A)[\vec{\alpha}/\vec{n}] = (A \wedge \neg A)$ which is a wfs of $\mathcal{L}^=$. As a result, $X[\vec{\alpha}/\vec{n}]$ is a proof of $A \wedge \neg A$ from Γ in FOL$^=$, and so Γ is not consistent in FOL$^=$, contradicting the above. Thus Γ is consistent in FOL$^=_{\mathbb{N}}$. □

Although the proof of **Lemma 12.2** is not immediate, it is hardly surprising that merely adding new constants would enable the derivation of a contradiction from Γ when no contradiction is derivable from Γ without those additional constants. Since Γ is consistent in $\mathcal{L}^=$, we may conclude that Γ is consistent in $\mathcal{L}^=_{\mathbb{N}}$. We may now proceed to extend Γ further, using all of the expressive resources of the extended language $\mathcal{L}^=_{\mathbb{N}}$. In particular, the following section will begin by finding a constant from $\mathbb{N}$ to witness each existential claim that may occur in Γ so that we never end up in a situation where an existential claim is true, but no particular instance is true. Nevertheless, this is one of the more opaque portions of the proof where it will only become clear later on why the definitions given here are needed.

12.2.2 Saturation

We will now move to extend Γ to a saturated set of wfss. Letting $\varphi(\alpha)$ be a wff of $\mathcal{L}^=_\mathbb{N}$ with at most one free variable α, we may take a set of wfss Σ to be SATURATED in $\mathcal{L}^=_\mathbb{N}$ just in case for each wff $\varphi(\alpha)$ of $\mathcal{L}^=_\mathbb{N}$, there is a constant β in $\mathcal{L}^=_\mathbb{N}$ where $(\exists\alpha\varphi \to \varphi[\beta/\alpha]) \in \Sigma$. In order to extend Γ to a saturated set Σ, fix an enumeration $\varphi_1(\alpha_1), \varphi_2(\alpha_2), \varphi_3(\alpha_3), \ldots$ of all wffs of $\mathcal{L}^=_\mathbb{N}$ with at most one free variable. We may then provide the following recursive definition:

θ-Base: $\theta_1 = (\exists\alpha_1\varphi_1 \to \varphi_1[n_1/\alpha_1])$ where $n_1 \in \mathbb{N}$ is the first constant not in φ_1.

θ-Recursion: $\theta_{k+1} = (\exists\alpha_{k+1}\varphi_{k+1} \to \varphi_{k+1}[n_{k+1}/\alpha_{k+1}])$ where $n_{k+1} \in \mathbb{N}$ is the first constant not in φ_{k+1} or θ_j for any $j \leqslant k$.

Given the infinite supply of new constants $\mathbb{N}$, we may always find an unused constant at each stage k in the process of constructing θ_{k+1}. We may then extend Γ to the saturated set Σ_Γ:

$$\begin{aligned}\Sigma_0 &= \Gamma\\ \Sigma_{n+1} &= \Sigma_n \cup \{\theta_n\}\\ \Sigma_\Gamma &= \bigcup_{i\in\mathbb{N}} \Sigma_n.\end{aligned}$$

Equivalently, $\Sigma_\Gamma = \Gamma \cup \{\theta_i : i \in \mathbb{N}\}$. The reason we did not define Σ_Γ in this way is to ease the exposition of the proof that Σ_Γ is consistent which goes by induction on the stages of the construction of Σ_Γ. Accordingly, you can think of the recursive definition of Σ_Γ as little more than a convenient notation reminiscent of the constructions used Chapter 5.

A similar sort of recursive construction will come up again when we introduce a maximal consistent extension of Σ_Γ. The reason these constructions are used both here and while proving PL Completeness before is that in each case they help us to establish the consistency of a set of wfss. Whereas in Chapter 5 this happened just once, now we will have two stages of consistent extension. Since we will continue to speak of consistency throughout may of the results that follow, it will become cumbersome to specify that we have consistency in FOL$^=_\mathbb{N}$ each time. Accordingly, we often speak of sets of wfss as being consistent full stop, where it is to be understood that we have FOL$^=_\mathbb{N}$ in mind.

Before showing that Σ_Γ is consistent, we will begin by proving the following lemma.

Lemma 12.3 If $\Lambda \cup \{\varphi\}$ is inconsistent, then $\Lambda \vdash \neg\varphi$.

Proof: Identical to **Lemma 5.1**. □

It is worth reviewing the proof of **Lemma 5.1** if you cannot remember the details. As with so many proofs, a small technique can make the difference between being able to explain why something is true even if you have convinced yourself of its truth.

Lemma 12.4 If $\Lambda \vdash \varphi$ and $\Pi \cup \{\varphi\} \vdash \psi$, then $\Lambda \cup \Pi \vdash \psi$.

Proof: Assume that $\Lambda \vdash \varphi$ and $\Pi \cup \varphi \vdash \psi$. It follows that there is a proof of φ from Λ as well as a proof Y of ψ from Π. Let Z be the result of replacing the line in which φ occurs as a premise with X. Since Z proves ψ from the premises Λ in X together with the premises $\Pi \cup \{\varphi\}$ in Y with the exception of φ, we may conclude that Z proves φ from $\Lambda \cup \Pi$, and so $\Lambda \cup \Pi \vdash \psi$. □

Lemma 12.5 If $\Lambda \vdash \varphi$ and $\Lambda \vdash \neg\varphi$, then Λ is inconsistent.

Proof: Identical to **Lemma 5.2**. □

Lemma 12.6 If Γ is consistent, then Σ_Γ is consistent and saturated in $\mathcal{L}^=_\mathbb{N}$.

Proof: Recall the enumeration of wffs of $\mathcal{L}^=_\mathbb{N}$ with one free variable from before. Letting $\varphi(\alpha)$ be any wff of $\mathcal{L}^=_\mathbb{N}$ with one free variable, $\varphi(\alpha) = \varphi_i(\alpha_i)$ for some $i \in \mathbb{N}$. By construction $\theta_i \in \Sigma_\Gamma$ where $\theta_i = (\exists\alpha_i\varphi_i \to \varphi_i[n_i/\alpha_i])$. Thus there is some constant n_i where $(\exists\alpha\varphi \to \varphi[n_i/\alpha]) \in \Sigma_\Gamma$, and so Σ_Γ is saturated.

The proof that Σ_Γ is consistent goes by induction on its construction where the consistency of Σ_0 follows from its definition given the starting assumption. Assume Σ_m is consistent. To show that Σ_{m+1} is consistent, assume for contradiction that Σ_{m+1} is not consistent. Since $\Sigma_{m+1} = \Sigma_m \cup \{\theta_{m+1}\}$, we know that $\Sigma_m \vdash \neg\theta_{m+1}$ by **Lemma 12.3**, and so $\Sigma_m \vdash \neg(\exists\alpha_{m+1}\varphi_{m+1} \to \varphi_{m+1}[n_{m+1}/\alpha_{m+1}])$.

Given that the derived rules for PL are also derived rules in FOL$^=$, it follows by **Lemma 12.4** that $\Sigma_m \vdash \exists\alpha_{m+1}\varphi_{m+1}$ and $\Sigma_m \vdash \neg\varphi_{m+1}[n_{m+1}/\alpha_{m+1}]$. In particular, we may derive the rules $\neg(\varphi \to \psi) \vdash \varphi$ and $\neg(\varphi \to \psi) \vdash \neg\psi$, where applications of these rules together with **Lemma 12.4** justifies the claims indicated in the previous sentence.

Since n_{m+1} does not occur in φ_{m+1} or in Σ_m, it follows by universal introduction $\forall$I that $\Sigma_m \vdash \forall\alpha_{m+1}\neg\varphi_{m+1}$. Given that $\forall\alpha_{m+1}\neg\varphi_{m+1} \vdash \neg\exists\alpha_{m+1}\varphi_{m+1}$ by the quantifier exchange rule ($\forall\neg$) derived in §10.6, we know that $\Sigma_m \vdash \neg\exists\alpha_{m+1}\varphi_{m+1}$ by **Lemma 12.4**, and so Σ_m is inconsistent by **Lemma 12.5**, contradicting our assumption. Thus we may conclude that Σ_{m+1} is consistent, and so it follows by induction that Σ_k is consistent for all $k \in \mathbb{N}$ as desired.

Assume for contradiction that Σ_Γ is inconsistent. By definition $\Sigma_\Gamma \vdash A \wedge \neg A$, and so there is a proof X of $A \wedge \neg A$ from the premises Σ_Γ. Since every proof is finite, at most finitely many premises in Σ_Γ are cited in X, and so there is some $m \in \mathbb{N}$ where every premise cited in X occurs in Σ_m. As a result, $\Sigma_m \vdash A \wedge \neg A$, and so Σ_m is inconsistent, contradicting the above. Thus Σ_Γ is consistent. □

12.2.3 Maximization

A set of wfss Δ is MAXIMAL in $\mathcal{L}^{=}_{\mathbb{N}}$ just in case either $\psi \in \Delta$ or $\neg\psi \in \Delta$ for every sentence ψ of $\mathcal{L}^{=}_{\mathbb{N}}$. Having shown that Σ_Γ is consistent if Γ is consistent, we may now maximize Σ_Γ by adding every sentence that we can consistently. Whereas before we enumerated all wffs which contain a single free variable, we will now enumerate all wfss $\psi_0, \psi_1, \psi_2, \ldots$ of $\mathcal{L}^{=}_{\mathbb{N}}$ whatsoever in order to present the following recursive construction:

$$\Delta_0 = \Sigma$$

$$\Delta_{n+1} = \begin{cases} \Delta_n \cup \{\psi_n\} & \text{if } \Gamma_n \cup \{\psi_n\} \text{ is consistent} \\ \Delta_n \cup \{\neg\psi_n\} & \text{otherwise.} \end{cases}$$

$$\Delta_\Sigma = \bigcup_{i \in \mathbb{N}} \Delta_n.$$

If Σ_Γ is consistent, we may show that Δ_{Σ_Γ} is both consistent and maximal where it follows as a result that Δ_{Σ_Γ} is deductively closed. Moreover, we may show $\Gamma \subseteq \Sigma_\Gamma \subseteq \Delta_{\Sigma_\Gamma}$ where Δ_{Σ_Γ} is saturated on account of including Σ_Γ. These properties will form the basis upon which the Henkin model is constructed in section §12.3 and then shown to satisfy Γ. In order to establish these results, we will begin by relabeling the following supporting lemma.

Lemma 12.7 If $\Lambda \cup \{\varphi\}$ and $\Lambda \cup \{\neg\varphi\}$ are both inconsistent, then Λ is inconsistent.

Proof: Identical to **Lemma 5.6**.

Lemma 12.8 If Γ is consistent, then Δ_{Σ_Γ} is maximal in $\mathcal{L}^{=}_{\mathbb{N}}$ and consistent.

Proof: Let φ be any wfs of $\mathcal{L}^{=}_{\mathbb{N}}$. Thus $\varphi = \psi_i$ for some $i \in \mathbb{N}$ given the enumeration above where either $\psi_i \in \Delta_{i+1}$ or $\neg\psi_i \in \Delta_{i+1}$. Since $\Delta_{i+1} \subseteq \Delta_{\Sigma_\Gamma}$, either $\varphi \in \Delta_{\Sigma_\Gamma}$ or $\neg\varphi \in \Delta_{\Sigma_\Gamma}$, and so Δ_{Σ_Γ} is maximal in $\mathcal{L}^{=}_{\mathbb{N}}$.

The proof that Δ_{Σ_Γ} is consistent goes by induction on the construction of Δ_{Σ_Γ}, where we know by **Lemma 12.6** that $\Sigma_\Gamma = \Delta_0$ is consistent. Assume for weak induction that Δ_n is consistent. There are two cases to consider.

Case 1: $\Delta_n \cup \{\psi_n\}$ is consistent, and so $\Delta_{n+1} = \Delta_n \cup \{\psi_n\}$ is consistent.

Case 2: $\Delta_n \cup \{\psi_n\}$ is not consistent, and so $\Delta_{n+1} = \Delta_n \cup \{\neg\psi_n\}$. Assume for contradiction that $\Delta_n \cup \{\neg\psi_n\}$ is not consistent. By **Lemma 12.7**, Δ_n is inconsistent, contradicting the hypothesis. Thus Γ_{n+1} is consistent.

Since Γ_{n+1} is consistent, it follows by induction that Γ_n is consistent for all $n \in \mathbb{N}$. Assume for contradiction that Δ_{Σ_Γ} is inconsistent. Thus $\Delta_{\Sigma_\Gamma} \vdash \bot$, and so there is a derivation Y of $\bot$ from Δ_{Σ_Γ} in FOL$^{=}_{\mathbb{N}}$. Since Y is finite, there is a finite number of premises cited in Y, and so there is some $k \in \mathbb{N}$ where every premise cited in Y belongs to Δ_k. As a result, Y is also a proof of $\bot$ from Δ_k, and so Δ_k is inconsistent, contradicting the above. Thus Δ_{Σ_Γ} is consistent. □

Lemma 12.9 $\Gamma \subseteq \Sigma_\Gamma \subseteq \Delta_{\Sigma_\Gamma}$ where Δ_{Σ_Γ} is saturated.

Proof: By definition, $\Gamma = \Sigma_0$ where $\Sigma_0 \subseteq \Sigma_\Gamma$, and $\Sigma_\Gamma = \Delta_0$ where $\Delta_0 \subseteq \Delta_{\Sigma_\Gamma}$. Thus $\Gamma \subseteq \Delta_{\Sigma_\Gamma}$. Moreover, Δ_{Σ_Γ} is saturated since otherwise there would be some wff $\varphi(\alpha)$ of $\mathcal{L}^{=}_{\mathbb{N}}$ with one free variable but no constant n where $(\exists\alpha\varphi \to \varphi[n/\alpha]) \in \Delta_{\Sigma_\Gamma}$. Since $\Sigma_\Gamma \subseteq \Delta_{\Sigma_\Gamma}$, there would be no constant n where $(\exists\alpha\varphi \to \varphi[n/\alpha]) \in \Sigma_\Gamma$, and so Σ_Γ would not be saturated, contradicting **Lemma 12.6**. □

As brought out in the proof of PL COMPLETENESS before, maximal consistent sets of wfss have some nice properties where principle among these is deductive closure.

12.2.4 Deductive Closure

Maximal consistent sets of wfss are deductively closed insofar as they contain every wfs derivable from that set as a member. Put formally, Δ is DEDUCTIVELY CLOSED just in case $\varphi \in \Delta$ whenever $\Delta \vdash \varphi$. Since $\Delta \vdash \varphi$ whenever $\varphi \in \Delta$, deductively closed sets of wfss are identical to the set of wfss which they derive. In order to show that Δ_{Σ_Γ} is deductively closed, we will begin by relabeling the following general purpose lemma from before.

Lemma 12.10 If Δ is maximal in $\mathcal{L}^{=}_{\mathbb{N}}$ and consistent, then $\varphi \in \Delta$ whenever $\Delta \vdash \varphi$.

Proof: Assume Δ is maximal in $\mathcal{L}^{=}_{\mathbb{N}}$ and consistent where $\Delta \vdash \varphi$. If $\Delta \vdash \neg\varphi$, then Δ is inconsistent by **Lemma 12.5**, contradicting the assumption. Thus $\Delta \nvdash \neg\varphi$, and so $\neg\varphi \notin \Delta$ since otherwise $\Delta \vdash \neg\varphi$. Since Δ is maximal, $\varphi \in \Delta$. □

This completes the setup for the construction of the Henkin Model presented in the following section. Whereas saturation was required to make sure every existential claim has a witness, maximal consistency is familiar from Chapter 5.

Although certain similarities will persist, the Henkin model that we will define over Δ_{Σ_Γ} will differ considerably from the Henkin interpretation that we introduced before. In particular, constructing the domain will require some care so as to ensure that any two constants that name the same individual are assigned to the same element of the domain of our Henkin model. None of these considerations occurred before, and indeed would be considerably simpler in a language without identity. These are the costs incurred by expanding the expressive power of our language to include not just predicates and the quantifiers but also a designated identity predicate for which we provided a semantics.

12.3 Henkin Model

Having extended the consistent set of sentences Γ in $\mathcal{L}^=$ to a saturated maximal consistent set of sentences Δ_{Σ_Γ} in $\mathcal{L}^=_\mathbb{N}$ which was shown to be deductively closed, we may proceed to use Δ_{Σ_Γ} to construct a Henkin model that satisfies Δ_{Σ_Γ}, and so satisfies Γ as a result. For ease of exposition, we will often drop the subscripts, assuming $\Delta = \Delta_{\Sigma_\Gamma}$ throughout.

We begin by letting $\mathbb{C}$ be the set of all constants in $\mathcal{L}^=_\mathbb{N}$ where this includes all the typical constants included in $\mathcal{L}^=$ together with the natural numbers $\mathbb{N}$ which we added. Since more than one constant can refer to the same element in a domain, we will model the elements of the domain as equivalence classes of co-referring constants. Consider the following:

Element: $[\alpha]_\Delta = \{\beta \in \mathbb{C} : \alpha = \beta \in \Delta\}$.

Domain: $\mathbb{D}_\Delta = \{[\alpha]_\Delta \subseteq \mathbb{C} : \alpha \in \mathbb{C}\}$.

The equivalence class $[\alpha]_\Delta$ is the set of constants in $\mathbb{C}$ which includes β just in case $\alpha = \beta$ belongs to Δ. In order to show that $[\alpha]_\Delta \neq \varnothing$ for any constant $\alpha \in \mathbb{C}$, we begin by relabeling the weakening principle from before which will be of general utility throughout the proof:

Lemma 12.11 If $\Lambda \vdash \varphi$, then $\Lambda \cup \Pi \vdash \varphi$.

Proof: Identical to **Lemma 5.6**. □

Every constant $\alpha \in \mathbb{C}$ generates an element $[\alpha]_\Delta \in \mathbb{D}_\Delta$ which includes α as a member:

Lemma 12.12 $\alpha \in [\alpha]_\Delta$ for any constant $\alpha \in \mathbb{C}$.

Proof: Let $\alpha \in \mathbb{C}$ be an arbitrary constant. Since $\vdash \alpha = \alpha$ by identity introduction =I, it follows that $\Delta \vdash \alpha = \alpha$ by **Lemma 12.11**, and so $\alpha = \alpha \in \Delta$ by **Lemma 12.10**. Thus $\alpha \in [\alpha]_\Delta$ for any constant $\alpha \in \mathbb{C}$. □

Next we may show that the elements of $\mathbb{D}_\Delta$ are well defined with the following:

Lemma 12.13 If $\alpha = \beta \in \Delta$, then $[\alpha]_\Delta = [\beta]_\Delta$.

Proof: Assume $\alpha = \beta \in \Delta$. Letting $\gamma \in [\alpha]_\Delta$, it follows that $\alpha = \gamma \in \Delta$. Since $\alpha = \beta, \alpha = \gamma \vdash \beta = \gamma$ by identity elimination =E, we know that $\Delta \vdash \beta = \gamma$ by **Lemma 12.11**, and so $\beta = \gamma \in \Delta$ by **Lemma 12.10**. Thus $\gamma \in [\beta]_\Delta$, and so generalising on γ, it follows that $[\alpha]_\Delta \subseteq [\beta]_\Delta$. By parity of reasoning, we may conclude that $[\beta]_\Delta \subseteq [\alpha]_\Delta$, and so $[\alpha]_\Delta = [\beta]_\Delta$ as desired. □

This shows that it does not matter which element $\alpha \in [\alpha]_\Delta$ we choose to represent the element $[\alpha]_\Delta$. For instance, if $\beta \in [\alpha]_\Delta$, then $\alpha = \beta \in \Delta$ and so we could have written '$[\beta]_\Delta$' in place of '$[\alpha]_\Delta$' since $[\alpha]_\Delta = [\beta]_\Delta$. As a result, the elements in $\mathbb{D}_\Delta$ are well-defined.

Having constructed the domain, we may proceed to specify an interpretation of the constants and predicates included in $\mathcal{L}^=_\mathbb{N}$. Rather than specifying any interpretation at all, we will make a number of especially convenient choices in order to guarantee that the resulting model satisfies all of the wfss in Δ. In particular, consider the following definitions:

Constants: $\mathcal{I}_\Delta(\alpha) = [\alpha]_\Delta$ for all constants $\alpha \in \mathbb{C}$.

Predicates: $\mathcal{I}_\Delta(\mathcal{F}^n) = \{\langle [\alpha_1]_\Delta, \ldots, [\alpha_n]_\Delta \rangle \in \mathbb{D}^n_\Delta : \mathcal{F}^n\alpha_1, \ldots, \alpha_n \in \Delta\}$.

Whereas every constant α is assigned to the element $[\alpha]_\Delta$ it generates, the extension of any n-place predicate $\mathcal{F}^n$ includes all and only the ordered tuples $\langle [\alpha_1]_\Delta, \ldots, [\alpha_n]_\Delta \rangle$ for which $\mathcal{F}^n\alpha_1, \ldots, \alpha_n \in \Delta$. Given that $[\alpha]_\Delta = [\beta]_\Delta$ may hold for distinct constants α and β, we must check that there is no ensuing conflict among the atomic sentences included in Δ. Put otherwise, we must show that the extensions of predicates are well-defined as follows:

Lemma 12.14 If $\alpha_i = \beta_i \in \Delta$, then $\mathcal{F}^n\alpha_1, \ldots, \alpha_n \in \Delta$ just in case $\mathcal{F}^n\alpha_1, \ldots, \alpha_n[\beta_i/\alpha_i] \in \Delta$.

Proof: Assume that $\alpha_i = \beta_i \in \Delta$ for some $\alpha_i, \beta_i \in \mathbb{C}$ where $\mathcal{F}^n\alpha_1, \ldots, \alpha_n \in \Delta$. It follows that $\Delta \vdash \mathcal{F}^n\alpha_1, \ldots, \alpha_n[\beta_i/\alpha_i]$ by identity elimination =E, and so $\mathcal{F}^n\alpha_1, \ldots, \alpha_n[\beta_i/\alpha_i] \in \Delta$ by **Lemma 12.10**. By parity of reasoning, we may conclude that $\mathcal{F}^n\alpha_1, \ldots, \alpha_n \in \Delta$ just in case $\mathcal{F}^n\alpha_1, \ldots, \alpha_n[\beta_i/\alpha_i] \in \Delta$. □

Focusing on the first index, suppose that $\alpha_1 = \beta_1 \in \Delta$ where $\mathcal{F}^n\alpha_1, \ldots, \alpha_n \in \Delta$. We know by *Predicates* that $\langle [\alpha_1]_\Delta, \ldots, [\alpha_n]_\Delta \rangle \in \mathcal{I}_\Delta(\mathcal{F}^n)$, and so $\langle [\beta_1]_\Delta, \ldots, [\alpha_n]_\Delta \rangle \in \mathcal{I}_\Delta(\mathcal{F}^n)$ since $[\alpha_1]_\Delta = [\beta_1]_\Delta$ by **Lemma 12.13**. Thus $\mathcal{F}^n\beta_1, \ldots, \alpha_n \in \Delta$ again by *Predicates*. More generally, **Lemma 12.14** shows that the same considerations apply no matte the index. It follows that the extension $\mathcal{I}_\Delta(\mathcal{F}^n)$ for any predicate $\mathcal{F}^n$ is well-defined since their is no possibility of disagreement about whether $\langle [\alpha_1]_\Delta, \ldots, [\alpha_n]_\Delta \rangle \in \mathcal{I}_\Delta(\mathcal{F}^n)$ by merely changing the representative α_i for the element $[\alpha_i]_\Delta$ to β_i whenever $\alpha_i = \beta_i$.

Observe that $\mathcal{M}_\Delta = \langle \mathbb{D}_\Delta, \mathcal{I}_\Delta \rangle$ satisfies the definition of a $\mathcal{L}^=_\mathbb{N}$ model. Since this construction is due to Leon Henkin (1949), we will refer to $\mathcal{M}_\Delta$ as the HENKIN MODEL for Γ, recalling that $\Delta = \Delta_{\Sigma_\Gamma}$. It remains to show that $\mathcal{M}_\Delta$ satisfies Δ, and so satisfies Γ as a result.

Much will turn on the details that we have presented so far. In order to gather a few more intuitions for how Δ determines what $\mathbb{D}_\Delta$ includes before pressing on, let $\mathbb{C}^3 = \{a, b, c\}$ where $\Gamma_3 = \{a \neq b, a \neq c, a \neq b\}$ and Δ^3 is defined from Γ_3 in a similar manner as above. It is easy to show that $\mathbb{D}_{\Delta^3} = \{\{a\}, \{b\}, \{c\}\}$. Were we to let $\Gamma_2 = \{a = b, a \neq c, a \neq b\}$ where Δ^2 is defined from Γ_2, then $\mathbb{D}_{\Delta^2} = \{\{a, b\}, \{c\}\}$ since $[a]_{\Delta^2} = [b]_{\Delta^2} = \{a, b\}$. More generally, the more identities included in Δ^2, the fewer elements in the domain for the constants to name.

12.4 Satisfiability

We turn now to present some of the more substantial lemmas upon which the completeness of FOL⁼ will ultimately depend. To being with, consider the following quantifier lemmas.

Lemma 12.15 $\mathcal{V}^{\hat{a}}_{\mathcal{I}_\Delta}(\exists\alpha\varphi) = 1$ just in case $\mathcal{V}^{\hat{a}}_{\mathcal{I}_\Delta}(\varphi[\beta/\alpha]) = 1$ for some constant $\beta \in \mathbb{C}$.

Proof: Let $\hat{a}$ be a variable assignment defined over $\mathbb{D}_\Delta$ where $\mathcal{V}^{\hat{a}}_{\mathcal{I}_\Delta}(\exists\alpha\varphi) = 1$. It follows that $\mathcal{V}^{\hat{c}}_{\mathcal{I}_\Delta}(\varphi) = 1$ for some α-variant $\hat{c}$ of $\hat{a}$ by the semantics for the existential quantifier. Given that $\hat{c}(\alpha) \in \mathbb{D}_\Delta$, we know that $\hat{c}(\alpha) = [\beta]_\Delta$ for some constant $\beta \in \mathbb{C}$. Moreover, we know that $\mathcal{I}_\Delta(\beta) = [\beta]_\Delta$ and so $\hat{c}(\alpha) = \mathcal{I}_\Delta(\beta)$. Thus $\mathfrak{v}^{\hat{c}}_{\mathcal{I}}(\alpha) = \mathfrak{v}^{\hat{c}}_{\mathcal{I}}(\beta)$, and so $\mathcal{V}^{\hat{c}}_{\mathcal{I}_\Delta}(\varphi) = \mathcal{V}^{\hat{c}}_{\mathcal{I}_\Delta}(\varphi[\beta/\alpha])$ by **Lemma 11.5**.

Since α does not occur in $\varphi[\beta/\alpha]$ and $\hat{c}$ is a α-variant of $\hat{a}$, we know $\hat{c}(\gamma) = \hat{a}(\gamma)$ for all variables γ that occur in $\varphi[\beta/\alpha]$. Thus $\mathcal{V}^{\hat{c}}_{\mathcal{I}_\Delta}(\varphi[\beta/\alpha]) = \mathcal{V}^{\hat{a}}_{\mathcal{I}_\Delta}(\varphi[\beta/\alpha])$ by **Lemma 9.1**, and so $\mathcal{V}^{\hat{a}}_{\mathcal{I}_\Delta}(\varphi[\beta/\alpha]) = 1$ for some $\beta \in \mathbb{C}$ given the identities above.

Assume instead that $\mathcal{V}^{\hat{a}}_{\mathcal{I}_\Delta}(\varphi[\beta/\alpha]) = 1$ for some constant $\beta \in \mathbb{C}$. Letting $\hat{c}$ be the α-variant of $\hat{a}$ where $\hat{c}(\alpha) = \mathcal{I}_\Delta(\beta)$, it follows that $\mathfrak{v}^{\hat{c}}_{\mathcal{I}}(\alpha) = \mathfrak{v}^{\hat{c}}_{\mathcal{I}}(\beta)$, and so $\mathcal{V}^{\hat{c}}_{\mathcal{I}_\Delta}(\varphi) = \mathcal{V}^{\hat{c}}_{\mathcal{I}_\Delta}(\varphi[\beta/\alpha])$ by **Lemma 11.5**. Thus $\mathcal{V}^{\hat{c}}_{\mathcal{I}_\Delta}(\varphi) = 1$, and so $\mathcal{V}^{\hat{a}}_{\mathcal{I}_\Delta}(\exists\alpha\varphi) = 1$ follows by the semantics for the existential quantifier. □

Lemma 12.16 $\mathcal{V}^{\hat{a}}_{\mathcal{I}_\Delta}(\forall\alpha\varphi) = 1$ just in case $\mathcal{V}^{\hat{a}}_{\mathcal{I}_\Delta}(\varphi[\beta/\alpha]) = 1$ for all constants $\beta \in \mathbb{C}$.

Proof: Let $\hat{a}$ be defined over $\mathbb{D}_\Delta$ where $\mathcal{V}^{\hat{a}}_{\mathcal{I}_\Delta}(\forall\alpha\varphi) = 1$. It follows that $\mathcal{V}^{\hat{c}}_{\mathcal{I}_\Delta}(\varphi) = 1$ for every α-variant $\hat{c}$ of $\hat{a}$ by the semantics for the universal quantifier. Let $\beta \in \mathbb{C}$ be any constant. Since $[\beta]_\Delta \in \mathbb{D}_\Delta$, we may let $\hat{c}$ be the α-variant of $\hat{a}$ where $\hat{c}(\alpha) = [\beta]_\Delta$. Given that $\mathcal{I}_\Delta(\beta) = [\beta]_\Delta$, it follows that $\hat{c}(\alpha) = \mathcal{I}_\Delta(\beta)$. As a result, $\mathfrak{v}^{\hat{c}}_{\mathcal{I}}(\alpha) = \mathfrak{v}^{\hat{c}}_{\mathcal{I}}(\beta)$, and so $\mathcal{V}^{\hat{c}}_{\mathcal{I}_\Delta}(\varphi) = \mathcal{V}^{\hat{c}}_{\mathcal{I}_\Delta}(\varphi[\beta/\alpha])$ by **Lemma 11.5**.

Since α does not occur in $\varphi[\beta/\alpha]$ and $\hat{c}$ is a α-variant of $\hat{a}$, we know $\hat{c}(\gamma) = \hat{a}(\gamma)$ for all variables γ that occur in $\varphi[\beta/\alpha]$. Thus $\mathcal{V}^{\hat{c}}_{\mathcal{I}_\Delta}(\varphi[\beta/\alpha]) = \mathcal{V}^{\hat{a}}_{\mathcal{I}_\Delta}(\varphi[\beta/\alpha])$ by **Lemma 9.1**, and so $\mathcal{V}^{\hat{a}}_{\mathcal{I}_\Delta}(\varphi[\beta/\alpha]) = 1$ by the identities above. Generalizing on $\beta \in \mathbb{C}$, it follows that $\mathcal{V}^{\hat{a}}_{\mathcal{I}_\Delta}(\varphi[\beta/\alpha]) = 1$ for all constants $\beta \in \mathbb{C}$.

Assume instead that $\mathcal{V}^{\hat{a}}_{\mathcal{I}_\Delta}(\varphi[\beta/\alpha]) = 1$ for every constant $\beta \in \mathbb{C}$. Letting $\hat{c}$ be any α-variant of $\hat{a}$, it follows that $\hat{c}(\alpha) \in \mathbb{D}_\Delta$, and so $\hat{c}(\alpha) = [\beta]_\Delta$ for some constant $\beta \in \mathbb{C}$. Thus $\mathfrak{v}^{\hat{c}}_{\mathcal{I}}(\alpha) = \mathfrak{v}^{\hat{c}}_{\mathcal{I}}(\beta)$, and so $\mathcal{V}^{\hat{c}}_{\mathcal{I}_\Delta}(\varphi) = \mathcal{V}^{\hat{c}}_{\mathcal{I}_\Delta}(\varphi[\beta/\alpha])$ by **Lemma 11.5**. Since α does not occur in $\varphi[\beta/\alpha]$ and $\hat{c}$ is a α-variant of $\hat{a}$, we know $\hat{c}(\gamma) = \hat{a}(\gamma)$ for all variables γ that occur in $\varphi[\beta/\alpha]$. Thus $\mathcal{V}^{\hat{c}}_{\mathcal{I}_\Delta}(\varphi[\beta/\alpha]) = \mathcal{V}^{\hat{a}}_{\mathcal{I}_\Delta}(\varphi[\beta/\alpha])$ by **Lemma 9.1**. By assumption, $\mathcal{V}^{\hat{a}}_{\mathcal{I}_\Delta}(\varphi[\beta/\alpha]) = 1$, and so $\mathcal{V}^{\hat{c}}_{\mathcal{I}_\Delta}(\varphi) = 1$ follows from the identities established above. Generalizing on $\hat{c}$, we may conclude that $\mathcal{V}^{\hat{a}}_{\mathcal{I}_\Delta}(\forall\alpha\varphi) = 1$ by the semantics for the universal quantifier. □

Although the lemmas above are of limited significance on their own, they play a critical role in the proof of the following lemma which establishes that the Henkin Model $\mathcal{M}_\Delta$ has the desired property of satisfying Δ. Instead of proving this claim directly, it will be easier to establish a slightly stronger claim that Δ satisfies all and only the wfss in Δ. That Δ and so Γ are satisfiable will then follow as an immediate result.

Whereas the lemmas so far have been relatively easy to establish, all the pieces that we have developed will come together in the following lemma. Since the lemma goes by induction on complexity, there are a number of cases to check, resulting in a much longer proof. As always, take care to refer back to the beginning of the proof if you get lost and need to regain your bearings. Given the importance of this lemma, all but one case has been included in full.

Lemma 12.17 If Δ is a saturated maximal consistent set of $\mathcal{L}^=_{\mathbb{N}}$ sentences and φ is any $\mathcal{L}^=_{\mathbb{N}}$ sentence, then $\mathcal{M}_\Delta$ satisfies φ just in case $\varphi \in \Delta$.

Proof: Let Δ be a saturated maximal consistent set of $\mathcal{L}^=_{\mathbb{N}}$ sentences and φ is any $\mathcal{L}^=_{\mathbb{N}}$ sentence. Letting $\hat{a}$ be an arbitrary variable assignment defined over $\mathbb{D}_\Delta$, we will show that $\mathcal{V}^{\hat{a}}_{\mathcal{I}_\Delta}(\varphi) = 1$ just in case $\varphi \in \Delta$ by induction on the complexity of the wfss of $\mathcal{L}^=_{\mathbb{N}}$. There are two base cases and seven induction cases.

Base: Assume $\texttt{Comp}(\varphi) = 0$ and so either φ is $\mathcal{F}^n\alpha_1, \ldots, \alpha_n$ or $\alpha_1 = \alpha_n$ for some constants $\alpha_1, \ldots, \alpha_n \in \mathbb{C}$. Consider the following biconditionals:

$$\begin{aligned}
\mathcal{V}^{\hat{a}}_{\mathcal{I}_\Delta}(\mathcal{F}^n\alpha_1, \ldots, \alpha_n) = 1 \quad & \textit{iff} \quad \langle v^{\hat{a}}_{\mathcal{I}_\Delta}(\alpha_1), \ldots, v^{\hat{a}}_{\mathcal{I}_\Delta}(\alpha_n)\rangle \in \mathcal{I}_\Delta(\mathcal{F}^n) \\
& \textit{iff} \quad \langle \mathcal{I}_\Delta(\alpha_1), \ldots, \mathcal{I}_\Delta(\alpha_n)\rangle \in \mathcal{I}_\Delta(\mathcal{F}^n) \\
& \textit{iff} \quad \langle [\alpha_1]_\Delta, \ldots, [\alpha_n]_\Delta\rangle \in \mathcal{I}_\Delta(\mathcal{F}^n) \\
& \textit{iff} \quad \mathcal{F}^n\alpha_1, \ldots, \alpha_n \in \Delta.
\end{aligned}$$

Whereas the final biconditional follows by the definition of $\mathcal{I}_\Delta$, all of the other biconditionals are immediate from the definitions together with the assumptions. Something similar may be observed for identity sentences in $\mathcal{L}^=_{\mathbb{N}}$:

$$\begin{aligned}
\mathcal{V}^{\hat{a}}_{\mathcal{I}_\Delta}(\alpha_1 = \alpha_n) = 1 \quad & \textit{iff} \quad v^{\hat{a}}_{\mathcal{I}_\Delta}(\alpha_1) = v^{\hat{a}}_{\mathcal{I}_\Delta}(\alpha_n) \\
& \textit{iff} \quad \mathcal{I}_\Delta(\alpha_1) = \mathcal{I}_\Delta(\alpha_n) \\
& \textit{iff} \quad [\alpha_1]_\Delta = [\alpha_n]_\Delta \\
(*) \quad & \textit{iff} \quad \alpha_1 = \alpha_n \in \Delta.
\end{aligned}$$

In support of the final biconditional, assume $[\alpha_1]_\Delta = [\alpha_n]_\Delta$. By **Lemma 12.12**, we know that $\alpha_n \in [\alpha_n]_\Delta$, and so $\alpha_n \in [\alpha_1]_\Delta$. By definition, $\alpha_1 = \alpha_n \in \Delta$. Together with **Lemma 12.13**, we may conclude that $(*)$ holds where the other biconditionals follow from the definitions and the assumption that $\alpha_1, \alpha_n \in \mathbb{C}$. It follows that $\mathcal{V}^{\hat{a}}_{\mathcal{I}_\Delta}(\varphi) = 1$ just in case $\varphi \in \Delta$ whenever $\texttt{Comp}(\varphi) = 0$.

Induction: Assume for induction that $\mathcal{V}^{\hat{a}}_{\mathcal{I}_\Delta}(\varphi) = 1$ just in case $\varphi \in \Delta$ whenever $\texttt{Comp}(\varphi) \leqslant n$. Let φ be a sentence of $\mathcal{L}^{=}_{\mathbb{N}}$ where $\texttt{Comp}(\varphi) = n + 1$.

Case 1: Assume $\varphi = \neg\psi$. Since $\texttt{Comp}(\neg\psi) = \texttt{Comp}(\psi) + 1$ and $\texttt{Comp}(\varphi) = n + 1$, it follows that $\texttt{Comp}(\psi) = n$. We may then reason as follows:

$$\begin{array}{rrl} \mathcal{V}^{\hat{a}}_{\mathcal{I}_\Delta}(\varphi) = 1 & \textit{iff} & \mathcal{V}^{\hat{a}}_{\mathcal{I}_\Delta}(\neg\psi) = 1 \\ & \textit{iff} & \mathcal{V}^{\hat{a}}_{\mathcal{I}_\Delta}(\psi) \neq 1 \\ (\star) & \textit{iff} & \psi \notin \Delta \\ (\neg) & \textit{iff} & \neg\psi \in \Delta \\ & \textit{iff} & \varphi \in \Delta. \end{array}$$

Assuming $\psi \notin \Delta$, it follows by the maximality assumed of Δ that $\neg\psi \in \Delta$. Conversely, if $\neg\psi \in \Delta$, then $\psi \notin \Delta$ since otherwise $\Delta \vdash \bot$ by EFQ in §4.4.8, making Δ inconsistent contrary to assumption. This establishes $(\neg)$ where $(\star)$ holds by hypothesis and the other biconditionals follow from the semantics for negation together and the case assumption.

Case 2: Assume $\varphi = \psi \wedge \chi$. Since $\texttt{Comp}(\psi \wedge \chi) = \texttt{Comp}(\psi) + \texttt{Comp}(\chi) + 1$ and $\texttt{Comp}(\varphi) = n + 1$, it follows that $\texttt{Comp}(\psi), \texttt{Comp}(\chi) \leqslant n$. Thus we have:

$$\begin{array}{rrl} \mathcal{V}^{\hat{a}}_{\mathcal{I}_\Delta}(\varphi) = 1 & \textit{iff} & \mathcal{V}^{\hat{a}}_{\mathcal{I}_\Delta}(\psi \wedge \chi) = 1 \\ & \textit{iff} & \mathcal{V}^{\hat{a}}_{\mathcal{I}_\Delta}(\psi) = \mathcal{V}^{\hat{a}}_{\mathcal{I}_\Delta}(\chi) = 1 \\ (\star) & \textit{iff} & \psi, \chi \in \Delta \\ (\wedge) & \textit{iff} & \psi \wedge \chi \in \Delta \\ & \textit{iff} & \varphi \in \Delta. \end{array}$$

Assuming that $\psi, \chi \in \Delta$, we know that $\Delta \vdash \psi \wedge \chi$ by conjunction introduction $\wedge$I, and so $\psi \wedge \chi \in \Delta$ by **Lemma 12.10**. Assuming instead that $\psi \wedge \chi \in \Delta$, it follows that $\Delta \vdash \psi$ and $\Delta \vdash \chi$ by conjunction elimination $\wedge$E, and so $\psi, \chi \in \Delta$ by **Lemma 12.10**. This establishes $(\wedge)$.

Additionally, $(\star)$ holds by hypothesis, and the other biconditionals follow from the semantics for conjunction along with the case assumption.

Case 3: Assume $\varphi = \psi \vee \chi$. Since $\texttt{Comp}(\psi \vee \chi) = \texttt{Comp}(\psi) + \texttt{Comp}(\chi) + 1$ and $\texttt{Comp}(\varphi) = n + 1$, it follows that $\texttt{Comp}(\psi), \texttt{Comp}(\chi) \leqslant n$. Thus we have:

$$\begin{array}{rrl} \mathcal{V}^{\hat{a}}_{\mathcal{I}_\Delta}(\varphi) = 1 & \textit{iff} & \mathcal{V}^{\hat{a}}_{\mathcal{I}_\Delta}(\psi \vee \chi) = 1 \\ & \textit{iff} & \mathcal{V}^{\hat{a}}_{\mathcal{I}_\Delta}(\psi) = 1 \text{ or } \mathcal{V}^{\hat{a}}_{\mathcal{I}_\Delta}(\chi) = 1 \\ (\star) & \textit{iff} & \psi \in \Delta \text{ or } \chi \in \Delta \\ (\vee) & \textit{iff} & \psi \vee \chi \in \Delta \\ & \textit{iff} & \varphi \in \Delta. \end{array}$$

Assuming that $\psi \in \Delta$, we know that $\Delta \vdash \psi \vee \chi$ by disjunction introduction $\vee$I, and so $\psi \vee \chi \in \Delta$ by **Lemma 12.10**. Analogous reasoning shows that $\psi \vee \chi \in \Delta$ if $\chi \in \Delta$, and so $\psi \vee \chi \in \Delta$ if either $\psi \in \Delta$ or $\chi \in \Delta$.

Assume instead that $\psi \vee \chi \in \Delta$. If $\psi \in \Delta$, then either $\psi \in \Delta$ or $\chi \in \Delta$. If $\psi \notin \Delta$, then $\neg\psi \in \Delta$ by the maximality assumed of Δ, and so $\Delta \vdash \chi$ by DS from §4.4.3. Thus $\chi \in \Delta$ by **Lemma 12.10**, and so either $\psi \in \Delta$ or $\chi \in \Delta$. It follows that $\psi \in \Delta$ or $\chi \in \Delta$ if $\psi \vee \chi \in \Delta$ which together with the above establishes $(\vee)$.

Additionally, $(\star)$ holds by hypothesis, and the other biconditionals follow from the semantics for disjunction along with the case assumption.

Case 4: Assume $\varphi = \psi \to \chi$. Since $\texttt{Comp}(\psi \to \chi) = \texttt{Comp}(\psi) + \texttt{Comp}(\chi) + 1$ and $\texttt{Comp}(\varphi) = n + 1$, it follows that $\texttt{Comp}(\psi), \texttt{Comp}(\chi) \leqslant n$. Thus we have:

$$\begin{array}{rrl} \mathcal{V}^{\hat{a}}_{\mathcal{I}_\Delta}(\varphi) = 1 & \textit{iff} & \mathcal{V}^{\hat{a}}_{\mathcal{I}_\Delta}(\psi \to \chi) = 1 \\ & \textit{iff} & \mathcal{V}^{\hat{a}}_{\mathcal{I}_\Delta}(\psi) \neq 1 \text{ or } \mathcal{V}^{\hat{a}}_{\mathcal{I}_\Delta}(\chi) = 1 \\ (\star) & \textit{iff} & \psi \notin \Delta \text{ or } \chi \in \Delta \\ (\to) & \textit{iff} & \psi \to \chi \in \Delta \\ & \textit{iff} & \varphi \in \Delta. \end{array}$$

Assuming that $\psi \notin \Delta$, we know that $\neg\psi \in \Delta$ by the maximality of Δ. Moreover, it is easy to derive $\neg\psi \vdash \psi \to \chi$ since given $\neg\psi$ as a premise, we may use the assumption rule AS to write ψ on a second line, deriving χ by EFQ from §4.4.8 and using conditional introduction $\to$I to discharge the assumption. It follows that $\Delta \vdash \psi \to \chi$ by **Lemma 12.11**, and so $\psi \to \chi \in \Delta$ by **Lemma 12.10**.

Assuming instead that $\chi \in \Delta$, we may derive $\chi \vdash \psi \to \chi$ since given χ as a premise, we may use the assumption rule AS to write ψ on a second line. By then using the reiteration rule R, we may rewrite the premise χ, discharging our assumption with conditional introduction $\to$I in order to derive $\psi \to \chi$ from χ. Thus $\Delta \vdash \psi \to \chi$ by **Lemma 12.11**, and so $\psi \to \chi \in \Delta$ by **Lemma 12.10**. We may then conclude that $\psi \to \chi \in \Delta$ if either $\psi \notin \Delta$ or $\chi \in \Delta$.

Assume instead that $\psi \to \chi \in \Delta$. If $\psi \notin \Delta$, then $\psi \notin \Delta$ or $\chi \in \Delta$. If $\psi \in \Delta$, then $\Delta \vdash \chi$ by conditional elimination $\to$E, and so $\chi \in \Delta$ by **Lemma 12.10**. Thus $\psi \notin \Delta$ or $\chi \in \Delta$ if $\psi \to \chi \in \Delta$ which, given the above, establishes $(\to)$.

Additionally, $(\star)$ holds by hypothesis, and the other biconditionals follow from the semantics for the conditional along with the case assumption.

Case 5: Assume $\varphi = \psi \leftrightarrow \chi$. (Exercise for the reader.)

Case 6: Assume $\varphi = \exists\alpha\psi$. Since $\texttt{Comp}(\exists\alpha\psi) = \texttt{Comp}(\psi) + 1$ and $\texttt{Comp}(\varphi) = n + 1$, it follows that $\texttt{Comp}(\psi) = n$. We may then reason as follows:

$$\begin{aligned}
\mathcal{V}^{\hat{a}}_{\mathcal{I}_\Delta}(\varphi) = 1 \quad & \textit{iff} \;\; \mathcal{V}^{\hat{a}}_{\mathcal{I}_\Delta}(\exists\alpha\psi) = 1 \\
(*) \quad & \textit{iff} \;\; \mathcal{V}^{\hat{a}}_{\mathcal{I}_\Delta}(\psi[\beta/\alpha]) = 1 \text{ for some constant } \beta \in \mathbb{C} \\
(\star) \quad & \textit{iff} \;\; \psi[\beta/\alpha] \in \Delta \text{ for some constant } \beta \in \mathbb{C} \\
(\exists) \quad & \textit{iff} \;\; \exists\alpha\psi \in \Delta \\
& \textit{iff} \;\; \varphi \in \Delta.
\end{aligned}$$

Assume $\psi[\beta/\alpha] \in \Delta$ for some constant $\beta \in \mathbb{C}$. Thus $\Delta \vdash \exists\alpha\psi$ by existential introduction $\exists$I, and so $\exists\alpha\psi \in \Delta$ by **Lemma 12.10**. Assuming $\exists\alpha\psi \in \Delta$ instead, we know that ψ has at most one free variable α, and so $\psi = \varphi_i(\alpha_i)$ for some $i \in \mathbb{N}$ where $\alpha_i = \alpha$ by the enumeration given in §12.2.2. Thus $\exists\alpha_i\varphi_i \to \varphi_i[n_i/\alpha_i] \in \Delta$ by the saturation assumed of Δ. Since $n_i \in \mathbb{C}$, it follows that $\exists\alpha\psi \to \psi[\beta/\alpha] \in \Delta$ for some $\beta \in \mathbb{C}$, and so $\Delta \vdash \psi[\beta/\alpha]$ by conditional elimination $\to$E. We may then conclude by **Lemma 12.10** that $\psi[\beta/\alpha] \in \Delta$, thereby establishing $(\exists)$.

Additionally, $(\star)$ holds by hypothesis, $(*)$ is given by **Lemma 12.15**, and the other biconditionals follow from the case assumption.

Case 7: Assume $\varphi = \forall\alpha\psi$. Since $\texttt{Comp}(\forall\alpha\psi) = \texttt{Comp}(\psi) + 1$ and $\texttt{Comp}(\varphi) = n + 1$, it follows that $\texttt{Comp}(\psi) = n$. We may then reason as follows:

$$\begin{aligned}
\mathcal{V}^{\hat{a}}_{\mathcal{I}_\Delta}(\varphi) = 1 \quad & \textit{iff} \;\; \mathcal{V}^{\hat{a}}_{\mathcal{I}_\Delta}(\forall\alpha\psi) = 1 \\
(*) \quad & \textit{iff} \;\; \mathcal{V}^{\hat{a}}_{\mathcal{I}_\Delta}(\psi[\beta/\alpha]) = 1 \text{ for every constant } \beta \in \mathbb{C} \\
(\star) \quad & \textit{iff} \;\; \psi[\beta/\alpha] \in \Delta \text{ for every constant } \beta \in \mathbb{C} \\
(\forall) \quad & \textit{iff} \;\; \forall\alpha\psi \in \Delta \\
& \textit{iff} \;\; \varphi \in \Delta.
\end{aligned}$$

Assuming $\forall\alpha\psi \in \Delta$ and letting $\beta \in \mathbb{C}$ be arbitrary, it follows that $\Delta \vdash \psi[\beta/\alpha]$ by universal elimination $\forall$E, and so $\psi[\beta/\alpha] \in \Delta$ by **Lemma 12.10**.

Assume instead that $\forall\alpha\psi \notin \Delta$. Since Δ is maximal, $\neg\forall\alpha\psi \in \Delta$. By $(\neg\forall)$ from §10.6, we know that $\neg\forall\alpha\psi \vdash \exists\alpha\neg\psi$, and so $\Delta \vdash \exists\alpha\neg\psi$ by **Lemma 12.11**. Thus $\exists\alpha\neg\psi \in \Delta$ by **Lemma 12.10**.

Given that $\neg\psi$ has at most one free variable α, it follows that $\psi = \varphi_i(\alpha_i)$ for some $i \in \mathbb{N}$ where $\alpha_i = \alpha$ by the enumeration given in §12.2.2. Since Δ is saturated, it follows that $\exists\alpha_i\varphi_i \to \varphi_i[n_i/\alpha_i] \in \Delta$ where $n_i \in \mathbb{C}$, and so $\exists\alpha\neg\psi \to \neg\psi[\beta/\alpha] \in \Delta$ for some $\beta \in \mathbb{C}$. By conditional elimination $\to$E, we know $\Delta \vdash \neg\psi[\beta/\alpha]$.

If $\psi[\beta/\alpha] \in \Delta$, then $\Delta \vdash \psi[\beta/\alpha]$, and so it would follow by **Lemma 12.5** that Δ is inconsistent contrary to assumption. Thus $\psi[\beta/\alpha] \notin \Delta$ for some $\beta \in \mathbb{C}$. By contraposition, we may conclude that if $\psi[\beta/\alpha] \in \Delta$ for all $\beta \in \mathbb{C}$, then $\forall\alpha\psi \in \Delta$. Together with the above, this establishes $(\forall)$.

Additionally, $(\star)$ holds by hypothesis, $(*)$ is given by **Lemma 12.16**, and the other biconditionals follow from the case assumption.

Conclusion: It follows by induction that $\mathcal{V}^{\hat{a}}_{\mathcal{I}_\Delta}(\varphi) = 1$ just in case $\varphi \in \Delta$ where the variable assignment $\hat{a}$ and sentence φ in $\mathcal{L}^=_{\mathbb{N}}$ where both arbitrary. Thus we have:

$$\begin{array}{rll} \mathcal{M}_\Delta \text{ satisfies } \varphi & \textit{iff} & \mathcal{V}_{\mathcal{I}_\Delta}(\varphi) = 1 \\ & \textit{iff} & \mathcal{V}^{\hat{a}}_{\mathcal{I}_\Delta}(\varphi) = 1 \text{ for all v.a. } \hat{a} \\ (\star) & \textit{iff} & \varphi \in \Delta. \end{array}$$

Whereas the induction argument presented above was required to establish $(\star)$, the other biconditionals follow from the definitions. This completes the proof. □

Most of the work is done. All that remains is to connect the dots in order to establish $\text{FOL}^=$ COMPLETENESS. In particular, the following section extend the results that we have established for $\mathcal{L}^=_{\mathbb{N}}$ to apply to our original language $\mathcal{L}^=$. Since the details of the lemmas given above are independent of the global structure of the proof, it is worth reviewing the way that the proof was set up in order to regain perspective of the whole before continuing.

12.5 Restriction

It follows immediately from **Lemma 12.17** that $\mathcal{M}_\Delta$ satisfies Δ and so $\mathcal{M}_\Delta$ satisfies Γ since $\Gamma \subseteq \Delta$ by **Lemma 12.9**. Since $\mathcal{M}_\Delta$ is a model of $\mathcal{L}^=_{\mathbb{N}}$ and not $\mathcal{L}^=$, it remains to show that Γ is satisfied by a model of $\mathcal{L}^=$. Thus we will restrict $\mathcal{M}_\Delta$ to $\mathcal{L}^=$ as follows:

Restriction: $\mathcal{I}'_\Delta(\alpha) = [\alpha]_\Delta$ for every constant α in $\mathcal{L}^=$.

$\mathcal{I}'_\Delta(\mathcal{F}^n) = \mathcal{I}_\Delta(\mathcal{F}^n)$ for all n-place predicates $\mathcal{F}^n$ and $n \in \mathbb{N}$.

Since the predicates in $\mathcal{L}^=_{\mathbb{N}}$ are the same as those in $\mathcal{L}^=$, no change to the predicate extensions is required. By contrast, the set of constants in $\mathcal{L}^=$ is a proper subset of the set of constants in $\mathcal{L}^=_{\mathbb{N}}$. Given that our aim is to restrict consideration to the expressions in $\mathcal{L}^=$, it doesn't matter that the elements in $\mathbb{D}_\Delta$ over which we are interpreting $\mathcal{L}^=$ may contain constants that do not belong to $\mathcal{L}^=$. Rather, we simply need some nonempty set of elements over which to interpret $\mathcal{L}^=$. Since nothing requires the domain by which we interpret $\mathcal{L}^=$ to only include constants that belong to $\mathcal{L}^=$, interpreting $\mathcal{L}^=$ over the domain $\mathbb{D}_\Delta$ will suffice.

Given these considerations, we may take $\mathbb{D}'_\Delta = \mathbb{D}_\Delta$ as before, letting $\mathcal{M}'_\Delta = \langle \mathbb{D}'_\Delta, \mathcal{I}'_\Delta \rangle$ be the restriction of $\mathcal{M}$ to $\mathcal{L}^=$ as defined above. Since every wfs of $\mathcal{L}^=_{\mathbb{N}}$ is also a wfs of $\mathcal{L}^=$, it is easy to show that $\mathcal{M}'_\Delta$ and $\mathcal{M}_\Delta$ satisfy the same $\mathcal{L}^=$ sentences, and so $\mathcal{M}'_\Delta$ satisfies Γ where $\mathcal{M}'_\Delta$ is a model of $\mathcal{L}^=$. Thus we may conclude that Γ is satisfiable as desired.

In support of this conclusion, we begin by drawing on **Lemma 11.6** in order to establish the following result without having to indulge in another induction proof on the complexity.

Lemma 12.18 For all $\mathcal{L}^=$ sentences φ, $\mathcal{M}'_\Delta$ satisfies φ just in case $\mathcal{M}_\Delta$ satisfies φ.

Proof: Since $\mathcal{M}' = \langle \mathbb{D}_\Delta, \mathcal{I}'_\Delta \rangle$ and $\mathcal{M} = \langle \mathbb{D}_\Delta, \mathcal{I}_\Delta \rangle$ share the same domain $\mathbb{D}_\Delta$ where $\mathcal{I}_\Delta$ and $\mathcal{I}'_\Delta$ agree about every n-place predicate $\mathcal{F}^n$ and constant α that occurs in any wfs φ of $\mathcal{L}^=$, we know that $\mathcal{V}^{\hat{a}}_{\mathcal{I}'_\Delta}(\varphi) = \mathcal{V}^{\hat{a}}_{\mathcal{I}_\Delta}(\varphi)$ for any variable assignment $\hat{a}$ defined over $\mathbb{D}_\Delta$ by **Lemma 11.6**. We may then reason as follows:

$$\begin{array}{rll} \mathcal{M}'_\Delta \text{ satisfies } \varphi & \textit{iff} & \mathcal{V}_{\mathcal{I}'_\Delta}(\varphi) = 1 \\ & \textit{iff} & \mathcal{V}^{\hat{a}}_{\mathcal{I}'_\Delta}(\varphi) = 1 \text{ for all v.a. } \hat{a} \\ (\star) & \textit{iff} & \mathcal{V}^{\hat{a}}_{\mathcal{I}_\Delta}(\varphi) = 1 \text{ for all v.a. } \hat{a} \\ & \textit{iff} & \mathcal{V}_{\mathcal{I}_\Delta}(\varphi) = 1 \\ & \textit{iff} & \mathcal{M}_\Delta \text{ satisfies } \varphi. \end{array}$$

Whereas $(\star)$ follows from **Lemma 11.6**, the other biconditionals follow from the definitions. This completes the proof. □

Since $\mathcal{M}_\Delta$ satisfies Γ where every $\varphi \in \Gamma$ is a sentence of $\mathcal{L}^=$, it follows from **Lemma 12.18** that $\mathcal{M}'_\Delta$ satisfies Γ where $\mathcal{M}'_\Delta$ is a $\mathcal{L}^=$ model. Thus Γ is satisfiable with respect to the models of $\mathcal{L}^=$. Since Γ was any consistent set, we may draw the following conclusion:

Theorem 12.1 Every consistent set of $\mathcal{L}^=$ sentences Γ is satisfiable.

Proof: Let Γ be a consistent set of $\mathcal{L}^=$ sentences in FOL$^=$. By **Lemma 12.2**, Γ is a set of $\mathcal{L}^=_\mathbb{N}$ sentences that is consistent in FOL$^=_\mathbb{N}$, and so Σ_Γ is consistent and saturated in $\mathcal{L}^=_\mathbb{N}$ by **Lemma 12.6**. Given **Lemma 12.8** and **Lemma 12.9**, Δ_{Σ_Γ} is a saturated maximal consistent set of sentences in $\mathcal{L}^=_\mathbb{N}$ where $\Gamma \subseteq \Delta_{\Sigma_\Gamma}$. Letting $\Delta = \Delta_{\Sigma_\Gamma}$, **Lemma 12.17** shows that the Henkin model $\mathcal{M}_\Delta$ satisfies φ just in case $\varphi \in \Delta$, and so $\mathcal{M}_\Delta$ satisfies Δ. Having shown that $\Gamma \subseteq \Delta$, we know that $\mathcal{M}_\Delta$ satisfies Γ. Since Γ is a set of $\mathcal{L}^=$ sentences, it follows by **Lemma 12.18** that there is a model $\mathcal{M}'_\Delta$ of $\mathcal{L}^=$ that satisfies Γ. Thus Γ is satisfiable. □

Whereas we began with the assumption that Γ is consistent in FOL$^=$ which is made up of rules defined for the language $\mathcal{L}^=$, **Lemma 12.2** established the consistency of Γ in FOL$^=_\mathbb{N}$ since Γ is also a set of wfss of $\mathcal{L}^=_\mathbb{N}$. Throughout the remainder of the lemma, consistency is defined with respect to FOL$^=_\mathbb{N}$ rather than FOL$^=$ in order to show that Γ is satisfied by a model of $\mathcal{L}^=_\mathbb{N}$. It is not until **Lemma 12.18** that consideration is returned to $\mathcal{L}^=$, showing that Γ is also satisfied by a model of $\mathcal{L}^=$, and so is satisfiable in the desired sense.

Given the previous result, the completeness of FOL$^=$ over the semantics for $\mathcal{L}^=$ follows as a corollary. The proof below formalizes the reasoning that we provided early on in order to motivate our approach to ultimately establishing the following result.

Corollary 12.1 (FOL$^=$ COMPLETENESS) If $\Gamma \vDash \varphi$, then $\Gamma \vdash \varphi$.

Proof: Assume $\Gamma \vDash \varphi$ and let $\mathcal{M} = \langle \mathbb{D}, \mathcal{I} \rangle$ be a model that satisfies Γ. It follows that $\mathcal{M}$ satisfies φ, and so $\mathcal{V}_{\mathcal{I}}^{\hat{a}}(\varphi) = 1$ for every variable assignment $\hat{a}$. Given the semantics for negation, $\mathcal{V}_{\mathcal{I}}^{\hat{a}}(\neg\varphi) \neq 1$ for every variable assignment $\hat{a}$, and so $\mathcal{V}_{\mathcal{I}}(\neg\varphi) \neq 1$. Thus $\mathcal{M}$ does not satisfy $\neg\varphi$. By generalising on $\mathcal{M}$, no model that satisfies Γ also satisfies $\neg\varphi$, and so $\Gamma \cup \{\varphi\}$ is unsatisfiable. By **Theorem 12.1**, $\Gamma \cup \{\neg\varphi\}$ is inconsistent, and so $\Gamma \vdash \neg\neg\varphi$ by **Lemma 12.3**. Since $\neg\neg\varphi \vdash \varphi$ by DN from §4.4.8, we may conclude that $\Gamma \vdash \varphi$ by **Lemma 12.4**. $\square$

Completeness may seems like a good property for any proof system to have. In analogy, if we could build a complete calculator that could compute any arithmetical operation, that would seem like a good thing. With respect to the semantics we provided for $\mathcal{L}^=$, the completeness of FOL$^=$ shows that there is no (extensionally) better proof system which allows us to derive a valid inference that FOL$^=$ leaves out. However, there is another perspective which takes completeness to describe a certain limitation on what sorts of logical consequences hold between the sentences in $\mathcal{L}^=$, calling the notion of logical consequence that we provided for $\mathcal{L}^=$ into question. We will close by briefly reflecting on the status of FOL$^=$.

12.6 Compactness

In order to discipline the following reflections we may begin with an important consequence of the completeness of FOL$^=$, mirroring a related result that we established for PL.

Corollary 12.2 If $\Gamma \vDash \varphi$, then there is a finite subset $\Lambda \subseteq \Gamma$ where $\Lambda \vDash \varphi$.

Proof: Assume $\Gamma \vDash \varphi$. Thus $\Gamma \vdash \varphi$ by FOL$^=$ COMPLETENESS, and so there is a derivation X of φ from Γ. Letting Γ_X be the set of premises which appear in X, it follows that $\Gamma_X \vdash \varphi$, and so $\Gamma_X \vDash \varphi$ by FOL$^=$ SOUNDNESS. Since X is finite, Γ_X is also finite, and so there is a finite subset $\Lambda \subseteq \Gamma$ where $\Lambda \vDash \varphi$. $\square$

Corollary 12.3 (Compactness) Γ is satisfiable if every finite subset $\Lambda \subseteq \Gamma$ is satisfiable.

Proof: Assume for contraposition that Γ is unsatisfiable. It follows vacuously that $\Gamma \vDash \bot$, and so $\Lambda \vDash \bot$ by **Corollary 12.2** for some finite subset $\Lambda \subseteq \Gamma$. Thus there is some finite subset $\Lambda \subseteq \Gamma$ that is unsatisfiable. By contraposition, if every finite subset $\Lambda \subseteq \Gamma$ is satisfiable, then Γ is satisfiable. $\square$

This property is referred to as COMPACTNESS. Although compactness may seems like a nice property, it demonstrates that there cannot be wfs in $\mathcal{L}^=$ which are only logical consequences of infinite sets of wfss of $\mathcal{L}^=$. However, there would seems to be some natural examples. For instance, let $\Gamma_\infty = \{\exists_{\geqslant n} xFx : n \in \mathbb{N}\}$ be the set of wfs which say that at least n things are F for every natural number n. Although it would seem that it is a logical consequence of Γ_∞ that infinitely many things are F, this logical consequence cannot hold by compactness. More precisely, there is no wfs of $\mathcal{L}^=$ which asserts that infinitely many things are F.

In order to see this, suppose that there were a wfs A_∞ of $\mathcal{L}^=$ that is satisfied by all and only the models in which infinitely many things are F. It follows that $\Gamma_\infty \vDash A_\infty$, and so $\Lambda \vDash A_\infty$ for some finite subset $\Lambda \subseteq \Gamma_\infty$ by compactness. However, since every finite subset $\Lambda \subseteq \Gamma_\infty$ will have a finite model, A_∞ must have a finite model given that $\Lambda \vDash A_\infty$. But this contradicts the assumption that A_∞ is only satisfied by models in which infinitely many things are F. As a result, there are no wfs of $\mathcal{L}^=$ such as A_∞ that only have infinite models.

These conclusions do not tell against infinity, but rather expose a limitation of our present semantics. Although this is a limitation that we can accept, it suggests that there are stronger notions of logical consequence that we may wish to consider. These semantic theories will not be compact, and so will not admit of complete logics since otherwise we could construct a similar argument to what was given above. From this perspective, completeness describes a limitation of our semantics for $\mathcal{L}^=$ rather than a virtue. Even though $\mathcal{L}^=$ along with its semantics and proof system FOL$^=$ is extremely useful for a wide range of applications, logic does not end here. Rather, the systems that we have covered are just the beginning.