forallχ

An Introduction to Formal Logic

P.D. Magnus
University at Albany, State University of New York

Lorain County Remix by

J. Robert Loftis
Lorain County Community College

This is version 0.1 of the Lorain County Remix. It is current as of July 24, 2012.

This remix is based on Version 1.27 [090604] of For All X, which was current as of January 9, 2010 and is available at `www.fecundity.com/logic`

Typesetting was carried out entirely in LaTeX2ε. The style for typesetting proofs is based on fitch.sty (v0.4) by Peter Selinger, University of Ottawa.

Acknowledgement from For All *X by P.D. Magnus*

The author would like to thank the people who made this project possible. Notable among these are Cristyn Magnus, who read many early drafts; Aaron Schiller, who was an early adopter and provided considerable, helpful feedback; and Bin Kang, Craig Erb, Nathan Carter, Wes McMichael, and the students of Introduction to Logic, who detected various errors in previous versions of the book.

Acknowledgement for the Lorain County Remix

J. Robert Loftis would like to acknowledge P.D. Magnus for his work on the original *For All* X, Molly Hinshaw for copyediting a draft, and the students of all of his Introduction to Logic classes from 2009 onward, who probably often wished they had just been asked to shell out a hundred bucks for Hurley, rather than deal with an awkward work in progress.

Contents

Chapter 1

What Is Logic?

1.1 Logic and the Study of Reasoning

Logic is a part of the study of human reason, the ability we have think abstractly, solve problems, and infer true statements on the basis of evidence. Traditionally, logic has focused on the last of these items, the ability to make inferences on the basis of evidence. In a game of Clue, for instance, you might infer that the murder was committed with the candlestick on the basis of evidence that rules out the other murder weapons. In logic, we use the word "argument" to refer to the attempt to show that certain evidence supports a conclusion. This is very different from the sort of argument you might have with your family, which could involve screaming and throwing things. We are going to use the word "argument" a lot in this book, so you need to get used to thinking of it as a name for a rational process, and not a word that describes what happens when people disagree.

A logical argument is structured to give someone a reason to believe some conclusion. Here is one such argument:

1. In a game of Clue, the possible murder weapons are the knife, the candlestick, the revolver, the rope, the lead pipe, and the wrench.
2. The murder weapon was not the knife.
3. The murder weapon was not the revolver.
4. The murder weapon was not the rope, the lead pipe, or the wrench.
5. Therefore, the murder weapon was the candlestick.

The word "therefore" indicates that the final sentence is the *conclusion* of the argument. The other sentences are *premises* of the argument. If you believe the premises, then the argument provides you with a reason to believe the conclusion. We will explain this all in more detail in the next section.

LOGIC, then, is the study of argument. Other disciplines, like critical thinking, decision theory, and cognitive science, have dealt with other aspects of human reasoning, like abstract thinking and problem solving more generally. Logic, more specifically, is the business of evaluating arguments, sorting good ones from bad ones. The argument in the example above is a good one, based on the process of elimination. It is good because it leads to truth. If I've got all the premises right, the conclusion will also be right. So let's define logic this way: LOGIC is the study of virtue in argument, where an argument is considered virtuous if it helps us get to the truth.

Logic is different from *rhetoric*, which is the study of effective persuasion. An advertisement might convince you to buy a new truck by having a gravelly voiced announcer tell you it is "ram tough" and showing you a picture of the truck on top of a mountain, where it no doubt actually had to be airlifted. This sort of persuasion is often more effective at getting people to believe things than logical argument. We are not interested in this sort of persuasion because it is not helpful in getting us to the truth. We will leave the study of rhetoric to the politicians, spin doctors, and ad-men.

The key to studying argument is to set aside the subject being argued about and to focus on the *way* it is argued *for*. The example above was about a game of Clue. However the kind of reasoning used, process of elimination, could be applied to any subject. Suppose a group of friends is deciding which restaurant to eat at, and there aree six restaurants in town. If you could rule out five of the possibilities, you would use an argument just like the one above to decide where to eat. We will call this the *content neutrality* of logic: if we say an argument is good, then the same kind of argument applied to a different topic will also be good. If we say an argument is good for solving murders, we will say that the same kind of argument is good for deciding where to eat, what kind of disease is destroying your crops, or who to vote for.

This book is specifically about formal logic, not informal logic. The difference between formal and informal logic is how we go about setting aside the content of an argument. In formal logic we get content neutrality by replacing parts of the argument we are studying with abstract symbols. For instance, we could turn the argument above into a formal argument like this:

(1) There are six possibilities: A, B, C, D, E, and F.
(2) A is false.
(3) B is false.
(4) D, E, and F are false.
$\therefore$ The correct answer is C.

Here we have replaced the concrete possibilities in the first argument with abstract letters that could stand for anything. We have also replaced the English word "therefore" with the symbol that means "therefore" ($\therefore$). This lets us see the formal structure of the argument, which is why it works in any domain you can think of. Technically, we can define FORMAL LOGIC as the method for studying argument that uses abstract notation to identify the formal structure of argument.

In the chapters to come, you will learn more precise ways to replace English words with abstract letters and symbols to see their formal structure. In many ways, these letters and symbols will function a lot like mathematical symbols. Also, to a large extent, our paradigm of a good argument will be a mathematical proof. As a result, much of what we will study in this book will have the same puzzle solving character that one finds in mathematics.

The contrast to formal logic is informal logic, where you keep arguments in ordinary language and draw extensively on your knowledge of the world to evaluate them. You can think of the difference between formal logic and informal logic as the difference between a laboratory science and a field science. Informal logic is the field science: you go out and study arguments in their natural habitats, like newspapers, courtrooms, and scientific journal articles. Like studying mice scurrying around the field, the process takes patience, but it lets you see how things work in the real world. Formal logic, the subject of this book, takes arguments out of their natural habitat and performs experiments on them to see what they are capable of. Some of our arguments will wind up looking like the genetically engineered obese mice scientists use to study dieting. They wouldn't survive in the wild, but they tell us a lot about the inner workings of the process.

This chapter discusses some basic logical notions that apply to arguments in ordinary English. In the next chapter, we will start replacing the ordinary English with the symbols of formal logic so that we can begin experimenting with them. For now, however, we want to be sure the words we use to evaluate arguments in formal logic correspond to the words we use to evaluate arguments in ordinary English.

1.2 Sentence, Argument, Premise, Conclusion

Before we can get anywhere in studying logic, we need to say exactly what an argument is. Arguments are made out of sentences. In logic, we define a SENTENCE as a unit of language that can be true or false. All of the items in figure 1.1 are sentences.

Notice that when we say a sentence is something that can be true *or* false, that

a. *Tyrannosaurus rex* went extinct 65 million years ago.

b. *Tyrannosaurus rex*went extinct last week.

c. On this exact spot, 100 million years ago, a *T. rex* laid a clutch of eggs.

d. George W. Bush is the king of Jupiter.

e. Murder is wrong.

f. Abortion is murder.

g. Abortion is a woman's right.

h. Lady Gaga is pretty.

i. The slithy toves did gyre and gimble in the wabe.

j. The murder of logician Richard Montague was never solved.

Figure 1.1: Examples of sentences in logic.

includes true statements like (a) and false statements like (b). It also includes statements that must either be true or false, but we don't know which, like (c). It includes statements that are completely silly, like (d). Sentences in logic include statements about morality, like (e), and things that in other contexts might be called "statements of opinion," like (f) and (g). People disagree strongly about whether (f) or (g) are true, but it is definitely possible for one of them to be true. The same is true about (h), although it is a less important issue than (f) and (g). Sentences can include nonsense words like (i), because we don't really need to know what the sentence is about to see that it is the sort of thing that can be true or false. All of this relates back to the content neutrality of logic. The sentences we study can be about dinosaurs, abortion, Lady Gaga, and even the history of logic itself, as in sentence (j), which is true.

Other logic textbooks describe the components of argument as "statements," "propositions," "assertions," or any number of other things. There is actually a great deal of disagreement about what the differences between all of these things are and which is best used to describe arguments. However none of that makes a difference for this textbook. We could have used any of the other terms in this text, and it wouldn't change anything, and if you read something in another basic logic textbook about "statements" or "propositions," it will probably also apply to what we are calling "sentences."

"Sentences" in this text does *not* include three things that you might have been taught to call sentences in grade school: questions, commands, and exclamations. *Questions* like "Does the grass need to be mowed?" might be called interoggative sentences in a linguistics class, but for our purposes they are not sentences. A person who is asking the question is not claiming that something is true or false. Generally, *questions* will not count as sentences, but *answers* will. "What is this course about?" is not a sentence. "No one knows what this course is about" is a sentence.

premise indicators: because, as, for, since, given that, for the reason that

conclusion indicators: therefore, thus, hence, so, consequently, it follows that, in conclusion, as a result, then, must, accordingly, this implies that, this entails that, we may infer that

Figure 1.2: Premise and conclusion indicators.

For the same reason *commands* do not count as sentences for us, although sometimes they get called “imperative sentences.” If someone bellows “Mow the grass, now!” they are not saying whether the grass has been mowed or not. You might infer that they believe the lawn has not been mowed, but then again maybe they think the lawn is fine and just want to see you exercise. Note, however, that commands are not always phrased as imperatives. “You will respect my authority” *is* either true or false—either you will or you will not—and so it counts as a sentence in the logical sense.

Again, “Ouch!” is sometimes called an exclamatory sentence, but it is 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.

We use sentences to build arguments. An ARGUMENT is a connected series of sentences designed to convince an audience of another sentence. The sentence that someone is trying to convince the audience of is called the CONCLUSION and the sentences that do the convincing are called the PREMISES. When people mean to give arguments, they typically use words like “therefore” and “because.” These are meant to signal to the audience that what is coming is either a premise or a conclusion in an argument. Words and phrases like “because” signal that a premise is coming, so we call these premise indicators. Similarly, words and phrases like “therefore” signal a conclusion and are called conclusion indicators. Figure 1.2 is an incomplete list of indicator words and phrases in English.

One of the first things you have to learn to do in logic is to identify and analyze arguments. Premise and conclusion indicators are good clues to this. In the passage below, Aristotle lets you know he is giving an argument by using the premise indicator word “for.”

> The earth is spherical in shape. For the night sky looks different in the northern and southern parts of the earth, and this would be so if the earth were spherical in shape (Aristotle, *Physics.*)

In the next passage, Justice Blackmun lets you know he is giving an argument by using the conclusion indicator word “Consequently.”

> Mortality rates for women undergoing early abortions, where the procedure is legal, appear to be as low as or lower than the rates for normal childbirth. Consequently, any interest of the State in protecting the woman from an inherently hazardous procedure, except when it would be equally dangerous for her to forgo it, has largely disappeared. (Harry Blackmun, *Roe v. Wade*)

To show that you have identified arguments, and to make their logical structure more clear, it helps to write them in canonical form. An argument written in CANONICAL FORM has each premise numbered and written on a separate line. Indicator words and other unnecessary material should be removed from the premises. Although you can shorten the premises and conclusion, you need to be sure to keep them all complete sentences with the same meaning, so that they can be true or false. You then draw a line and write the conclusion preceded by the $\therefore$.. In canonical form, Aristotole's argument that the earth is round looks like this.

1. The night sky looks different in the northern and southern parts of the earth.
2. This would be so if the earth were spherical in shape.

$\therefore$ The earth is spherical in shape.

Blackmun's argument only has one long premise, and a long conclusion, so it would look like this.

1. Mortality rates for women undergoing early abortions, where the procedure is legal, appear to be as low as or lower than the rates for normal childbirth.

$\therefore$ Any interest of the State in protecting the woman from an inherently hazardous procedure, except when it would be equally dangerous for her to forgo it, has largely disappeared.

Notice that in the original passages, Aristotle put the conclusion first, while Blackmun put it last. In ordinary English, people can put the conclusion of their argument where ever they want. However, when we write the argument in canonical form, the conclusion goes last.

Unfortunately, indicator words aren't a perfect guide to when people are giving an argument. Look at this passage from a newspaper:

> The new budget underscores the consistent and paramount importance of tax cuts in the Bush philosophy. His first term cuts affected more money than any other initiative undertaken in his presidency, including the costs thus far of the war in Iraq. All told,

> including tax incentives for health care programs and the extension of other tax breaks that are likely to be taken up by Congress, the White House budget calls for nearly $300 billion in tax cuts over the next five years, and $1.5 trillion over the next 10 years. (Robin Toner, In Budget, Bush Holds Fast to a Policy of Tax Cutting, *New York Times*, February 7, 2006)

Although there are no indicator words, this is in fact an argument. The writer wants you to believe something about George Bush: tax cuts are his number one priority. The next two sentences in the paragraph give you reasons to believe this. You can write the argument in canonical form like this.

1. Bush's first term cuts affected more money than any other initiative undertaken in his presidency, including the costs thus far of the war in Iraq.
2. The White House budget calls for nearly $300 billion in tax cuts over the next five years, and $1.5 trillion over the next 10 years.

∴ Tax cuts are of consistent and paramount importance of in the Bush philosophy.

The ultimate test of whether something is an argument is simply whether some of the sentences provide reason to believe another one of the sentences. If some sentences support others, you are looking at an argument. The speakers in these two cases use indicator phrases to let you know they are trying to give an argument.

Notice that the definition of argument is quite general. Consider this example:

1. There is coffee in the coffee pot.
2. There is a dragon playing bassoon on the armoire.

∴ Salvador Dali was a poker player.

It may seem odd to call this an argument, but that is because it would be a terrible argument. The two premises have nothing at all to do with the conclusion. Nevertheless, given our definition, it still counts as an argument—albeit a bad one.

Practice Exercises

Throughout the book, you will find a series of practice problems that review and explore the material covered in the chapter. There is no substitute for actually working through some problems, because logic is more about a way of thinking than it is about memorizing facts.

Part A Which of the following are sentences in the logical sense?

1. England is smaller than China.
2. Greenland is south of Jerusalem.
3. Is New Jersey east of Wisconsin?
4. The atomic number of helium is 2.
5. The atomic number of helium is π.
6. I hate overcooked noodles.
7. Blech! Overcooked noodles!
8. Overcooked noodles are disgusting.
9. Take your time.
10. This is the last question.

Part B Which of the following are sentences in the logical sense?

1. Is this a question?
2. Nineteen out of the 20 known species of Eurasian elephants are extinct.
3. The government of the United Kingdom has formally apologized for the way it treated the logician Alan Turing.
4. Texting while driving
5. Texting while driving is dangerous.
6. Insanity ran in the family of logician Bertrand Russell, and he had a lifelong fear of going mad.
7. For the love of Pete, put that thing down before someone gets hurt!
8. Don't try to make too much sense of this.
9. Never look a gift horse in the mouth.
10. The physical impossibility of death in the mind of someone living

Part C Rewrite each of the following arguments in canonical form. Be sure to remove all indicator words and keep the premises and conclusion as complete sentences. Write the indicator words and phrases separately and state whether they are premise or conclusion indicators.

1. There is no reason to fear death. Why? Because death is a time when you will not exist, just like the time before you were born. You are not troubled by the fact that you didn't exist before you were born, and the time that you won't exist after you are dead is no different.
2. You cannot both oppose abortion and support the death penalty, unless you think there is a difference between fetuses and felons. Steve opposes abortion and supports the death penalty. Therefore Steve thinks there is a difference between fetuses and felons.

3. We know that whenever people from one planet invade another, they always wind up being killed by the local diseases, because in 1938, when Martians invaded the Earth, they were defeated because they lacked immunity to Earth's diseases. Also, in 1942, when Hitler's forces landed on the Moon, they were killed by Moon diseases.
4. If you have slain the Jabberwock, my son, it will be a frabjous day. The Jabberwock lies there dead, its head cleft with your vorpal sword. This is truly a fabjous day.
5. Miss Scarlett was jealous that Professor Plum would not leave his wife to be with her. Therefore she must be the killer, because she is the only one with a motive.

Part D Rewrite each of the following arguments in canonical form. Be sure to remove all indicator words and keep the premises and conclusion as complete sentences. Write the indicator words and phrases separately and state whether they are premise or conclusion indicators.

1. Hillary Clinton should drop out of the race for Democratic Presidential nominee. For every day she stays in the race, McCain gets a day free from public scrutiny and the members of the Democratic party get to fight one another.
2. You have to be smart to understand the rules of Dungeons and Dragons. Most smart people are nerds. So, I bet most people who play D&D are nerds.
3. Any time the public receives a tax rebate, consumer spending increases. Since the public just received a tax rebate, consumer spending will increase.
4. Caroline is taller than Joey. Kate must also be taller than Joey, because she is taller than Caroline.
5. The economy has been in trouble recently. And it's certainly true that cell phone use has been rising during that same period. So, I suspect increasing cell phone use is bad for the economy.

1.3 Two Ways That Arguments Can Go Wrong

Arguments are supposed to lead us to the truth, but they don't always succeed. There are two ways they can fail in their mission. First, they can simply start out wrong, using false premises. Consider the following argument.

1. It is raining heavily.
2. If you do not take an umbrella, you will get soaked.

$\therefore$ You should take an umbrella.

If premise (1) is false—if it is sunny outside—then the argument gives you no reason to carry an umbrella.The argument has failed its job. Premise (2) could also be false: Even if it is raining outside, you might not need an umbrella. You might wear a rain poncho or keep to covered walkways and still avoid getting soaked. Again, the argument fails because a premise is false.

Even if an argument has all true premises, there is still a second way it can fail. Suppose for a moment that both the premises in the argument above are true. You do not own a rain poncho. You need to go places where there are no covered walkways. Now does the argument show you that you should take an umbrella? Not necessarily. Perhaps you enjoy walking in the rain, and you would like to get soaked. In that case, even though the premises were true, the conclusion would be false. The premises, although true, do not *support* the conclusion.

The act of coming to believe a conclusion on the basis of some set of premises is called an INFERENCE. Inferences are like argument glue: they hold the premises and conclusion together. When an argument goes wrong because the premises do not support the conclusion, we say there is something wrong with the inference. When there is something wrong with the inference, that means there is something wrong with the *logical form* of the argument: Premises of the kind given do not necessarily lead to a conclusion of the kind given. We will be interested primarily in the logical form of arguments. We will learn to identify bad inferences by identifying bad logical forms.

Consider another example:

1. You are reading this book.
2. This is a logic book.

$\therefore$ You are a logic student.

This is not a terrible argument. Most people who read this book are logic students. Yet, it is possible for someone besides a logic student to read this book. If your roommate picked up the book and thumbed through it, they would not immediately become a logic student. So the premises of this argument, even though they are true, do not guarantee the truth of the conclusion. Its logical form is less than perfect.

Again, for any argument, there are two ways that it could fail. First, one or more of the premises might be false. Second, the premises might fail to support the conclusion. Even if the premises were true, the form of the argument might be weak, meaning the inference is bad.

1.4 Valid, Sound, Strong, Cogent

In logic, we are mostly concerned with evaluating the quality of inferences. The truth of various premises will be a matter of whatever specific topic we are arguing about, and, as we have said, logic is content neutral. The strongest inference possible would be one where the premises, if true, would force the conclusion to be true. This kind of inference is called valid. An argument is VALID if and only if it is impossible for the premises to be true and the conclusion false. If an argument is not perfect in this way, it is called INVALID. As we shall see, this term is a little misleading, because less than perfect arguments can be very useful. Traditionally, however, formal logic has only been concerned with sorting valid from invalid arguments, and that will be the topic of all the chapters in this book after this one. First, though, we need to look in more detail at the concept of validity.

The crucial thing about a valid argument is that it is impossible for the premises to be true *at the same time* that the conclusion is false. Consider this example:

1. Oranges are either fruits or musical instruments.
2. Oranges are not fruits.

∴ Oranges are musical instruments.

The conclusion of this argument is ridiculous. Nevertheless, it follows validly from the premises. This is a valid argument. *If* both premises were true, *then* the conclusion would necessarily be true.

This shows that a valid argument does not need to have true premises or a true conclusion. Conversely, having true premises and a true conclusion is not enough to make an argument valid. Consider this example:

1. London is in England.
2. Beijing is in China.

∴ Paris is in France.

The premises and conclusion of this argument are, as a matter of fact, all true. This is a terrible argument, however, because the premises have nothing to do with the conclusion. Imagine what would happen if Paris declared independence from the rest of France. Then the conclusion would be false, even though the premises would both still be true. Thus, it is *logically possible* for the premises of this argument to be true and the conclusion false. The argument is invalid.

The important thing to remember is that validity is not about the actual truth or falsity of the sentences in the argument. Instead, it is about the way the premises and conclusion are put together. It is really about the *form* of the argument. A valid argument has perfect logical form. The premises and conclusion have been put together so that: The truth of the premises is incompatible with the falsity of the conclusion.

Here's one more example. This argument is valid, even though one of its premises is false.

1. Socrates is a person.
2. All people are carrots.

∴ Therefore, Socrates is a carrot.

All people are not carrots. But if people were carrots, and Socrates were a person, Socrates would have to be a carrot.

If an argument is not only valid, but also has true premises, we call it SOUND. "Sound" is the highest compliment you can pay an argument. The premises are true and they make the conclusion true. If someone gives a sound argument in a conversation, you have to believe the conclusion, or else you are irrational. If we take the valid argument above, keep its form, but replace the false statement with a true one, we get one of the most famous sound arguments in the history of logic in the West.

1. Socrates is a person.
2. All people are mortal.

∴ Therefore, Socrates is mortal.

Arguments that are valid, or at least try to be, are called DEDUCTIVE, and people who attempt to argue using valid arguments are said to be arguing DEDUCTIVELY. The notion of validity we are using here is, in fact, sometimes called *deductive validity.* Deductive argument is difficult, because in the real world sound arguments are hard to come by, and people don't always recognize them as sound when they find them.

Fortunately, arguments can still be worthwhile, even if they are not deductively valid. Consider this one:

1. In January 1997, it rained in San Diego.
2. In January 1998, it rained in San Diego.
3. In January 1999, it rained in San Diego.

∴ It rains every January in San Diego.

This argument is not valid, because the conclusion could be false even though the premises are true. It is possible, although unlikely, that it will fail to rain next January in San Diego. Moreover, we know that the weather can be fickle.

No amount of evidence should convince us that it rains there *every* January. Who is to say that some year will not be a freakish year in which there is no rain in January in San Diego? Even a single counterexample is enough to make the conclusion of the argument false.

Still, this argument is pretty good. Certainly, the argument could be made stronger by adding additional premises: In January 2000, it rained in San Diego. In January 2001... and so on. Regardless of how many premises we add, however, the argument will still not be deductively valid. Instead of being valid, this argument is strong. An argument is STRONG if the premises would make the conclusion more likely, were they true. In a strong argument, the premises don't guarantee the truth of the conclusion, but they do make it a good bet. Arguments that purport to merely be strong rather than valid are called INDUCTIVE. The most common kind of inductive argument includes arguments like the one above, which generalize from many cases to a conclusion about all cases. One interesting feature of inductive arguments is that they can get stronger or weaker depending on the amount of evidence. Deductive validity is a black-or-white matter. You either have it, and you're perfect, or you don't, and you're nothing. Inductive arguments, on the other hand, come in degrees. If an argument is strong, and it has true premises, we say that it is COGENT. Cogency is the equivalent of soundness in inductive arguments.

Historically, most of formal logic has been devoted to the study of deductive arguments, although many great systems have been developed for the formal treatment of inductive logic. This textbook (in its current form) only deals with formal treatments of deductive logic.

Practice Exercises

Part A Which of the following arguments are valid? Mark all valid or invalid.

(1)

1. All dogs are mammals.
2. Fido is a dog.

∴ Fido is a mammal.

(2)

1. If grass is green, then I am the pope.
2. Grass is green.

∴ I am the pope.

(3)

1. All people are mortal.
2. Socrates is mortal.

$\therefore$ All people are Socrates.

(4)

1. If the triceratops were a dinosaur, it would be extinct.
2. The triceratops was a dinosaur.

$\therefore$ The triceratops is extinct.

(5)

1. Love is blind.
2. God is love.
3. Ray Charles is blind.

$\therefore$ Ray Charles is God.

Part B Which of the following arguments are valid? Mark all valid or invalid.

(1)

1. All dinosaurs are people.
2. All people are fruit.

$\therefore$, All dinosaurs are fruit.

(2)

1. If George Washington was assassinated, he is dead.
2. George Washington is dead.

$\therefore$ George Washington was assassinated.

(3)

1. All Whos live in Whoville.
2. Cindy Lou Who is a Who.

$\therefore$ Cindy Lou Who lives in Whoville.

(4)

1. If Frog and Toad like each other, they are friends.
2. Frog and Toad like each other.

∴ Frog and Toad are friends.

(5)

1. If Cindy Lou Who is no more than two, then she is not five.
2. Cindy Lou Who is not five.

∴ Cindy Lou Who is no more than two.

1.5 Other Logical Notions

In addition to deductive validity, we will be interested in some other logical concepts.

Truth values

A truth value is the status of a sentence as true or false. More precisely, a TRUTH VALUE is the status of a sentence with relationship to truth. We have to say this, because there are systems of logic that allow for truth values besides "true" and "false," like "maybe true," or approximately true." For instance, some philosophers have claimed that the future is not yet determined. If they are right, then sentences about *what will be the case* are not yet true or false. Some systems of logic accommodate this by having an additional truth value. Other formal languages, so-called paraconsistent logics, allow for sentences that are both true *and* false. We won't be dealing with those in this textbook, however. For our purposes, there are two truth values, "true" and "false," and every sentence has exactly one of these. Logical systems like ours are called BIVALENT.

Tautology, contingent sentence, contradiction

In considering arguments formally, we care about what would be true *if* the premises were true. Generally, we are not concerned with the actual truth value of any particular sentences—whether they are *actually* true or false. Yet there are some sentences that must be true, just as a matter of logic.

Consider these sentences:

1. It is raining.
2. Either it is raining, or it is not.
3. It is both raining and not raining.

In order to know if sentence 1 is true, you would need to look outside or check the weather channel. Logically speaking, it might be either true or false. Sentences like this are called *contingent* sentences.

Sentence 2 is different. You do not need to look outside to know that it is true. Regardless of what the weather is like, it is either raining or not. If it is drizzling, you might describe it as partly raining or in a way raining and a way not raining. However, our assumption of bivalence means that we have to draw a line, and say at some point that it is raining. And if we have not crossed this line, it is not raining. Thus the sentence "either it is raining or it is not" is always going to be true, no matter what is going on outside. Sentences that have to be true, as a matter of logic are called TAUTOLOGIES or logical truths.

You do not need to check the weather to know about sentence 3, either. It must be false, simply as a matter of logic. It might be raining here and not raining across town, it might be raining now but stop raining even as you read this, but it is impossible for it to be both raining and not raining here at this moment. The third sentence is *logically false*; it is false regardless of what the world is like. A logically false sentence is called a CONTRADICTION.

To be precise, we can define a CONTINGENT SENTENCE as a sentence that is neither a tautology nor a contradiction.

A sentence might *always* be true and still be contingent. For instance, it may be the case that in no time in the history of the universe was there ever an elephant with tiger stripes. Elephants only ever evolved on Earth, and there was never any reason for them to evolve tiger stripes. The sentence "Some elephants have tiger stripes," is therefore always false. It is, however, still a contingent sentence. The fact that it is always false is not a matter of logic. There is no contradiction in considering a possible world in which elephants evolved tiger stripes, perhaps to hide in really tall grass. The important question is whether the sentence *must* be true, just on account of logic.

Logical equivalence

We can also ask about the logical relations *between* two sentences. For example:

John went to the store after he washed the dishes.
John washed the dishes before he went to the store.

These two sentences are both contingent, since John might not have gone to the store or washed dishes at all. Yet they must have the same truth value. If either of the sentences is true, then they both are; if either of the sentences is false, then they both are. When two sentences necessarily have the same truth value, we say that they are LOGICALLY EQUIVALENT.

Consistency

Consider these two sentences:

B1 My only brother is taller than I am.

B2 My only brother is shorter than I am.

Logic alone cannot tell us which, if either, of these sentences is true. Yet we can say that *if* the first sentence (B1) is true, *then* the second sentence (B2) must be false. And if B2 is true, then B1 must be false. It cannot be the case that both of these sentences are true.

If a set of sentences could not all be true at the same time, like B1 and B2, they are said to be INCONSISTENT. Otherwise, they are CONSISTENT.

We can ask about the consistency of any number of sentences. For example, consider the following list of sentences:

G1 There are at least four giraffes at the wild animal park.

G2 There are exactly seven gorillas at the wild animal park.

G3 There are not more than two Martians at the wild animal park.

G4 Every giraffe at the wild animal park is a Martian.

G1 and G4 together imply that there are at least four Martian giraffes at the park. This conflicts with G3, which implies that there are no more than two Martian giraffes there. So the set of sentences G1–G4 is inconsistent. Notice that the inconsistency has nothing at all to do with G2. G2 just happens to be part of an inconsistent set.

Sometimes, people will say that an inconsistent set of sentences “contains a contradiction.” By this, they mean that it would be logically impossible for all of the sentences to be true at once. A set can be inconsistent even when all of the sentences in it are either contingent or tautologous. When a single sentence is a contradiction, then that sentence alone cannot be true.

1.6 Formal Languages

Consider these two arguments again:

1. Socrates is a person.
2. All men are mortal.

∴ Socrates is mortal.

1. Socrates is a person.
2. All people are carrots.

∴ Socrates is a carrot.

These arguments are both valid. In each case, if the premises were true, the conclusion would have to be true. (In the case of the first argument, the premises are actually true, so the argument is sound, but that is not what we are concerned with right now.) These arguments are valid, because they are put together the right way—the logical form is impeccable. In this case, the two arguments have the same logical form. Both arguments can be written like this:

1. S is P.
2. All Ps are Cs.

∴ S is C.

In both arguments S stands for Socrates and M stands for man. In the first argument, C stands for mortal; in the second, C stands for carrot. Both arguments have this form, and every argument of this form is valid. So both arguments are valid.

What we did here was replace words like *person* or *carrot* with symbols like P or C so as to make the logical form explicit. This is the central idea behind formal logic. We want to remove irrelevant or distracting features of the argument to make the logical form more perspicuous. The letters 'S', 'P', and 'C' are variables, just like the x you had to solve for in algebra. Just as x could stand for any number, 'S' can stand for any name. Aristotle, a philosopher who lived in Greece during the 4th century BC, was the first to use variables this way, and he did it while developing the first system of formal logic.

Aristotle's system worked for arguments like the ones above. This system, with some revisions, was the dominant logic in the Western world for more than two millennia. In Aristotelean logic, categories are replaced with capital letters. Every sentence of an argument is then represented as having one of four forms, which medieval logicians labeled in this way: (A) All As are Bs. (E) No As are Bs. (I) Some A is B. (O) Some A is not B.

It is then possible to describe valid *syllogisms*, three-line arguments like the two we considered above. Medieval logicians gave mnemonic names to all of the valid argument forms. The form of our two arguments, for instance, was called *Barbara*. The vowels in the name, all As, represent the fact that the two premises and the conclusion are all (A) form sentences. There are many limitations to Aristotelean logic. One is that it makes no distinction between kinds and individuals. So the first premise might just as well be written "All Ss are Ms": All Socrateses are men. Despite its historical importance, Aristotelean logic has been superseded.

Starting in the 19th century people learned to do more than simply replace categories with variables. They learned to replicate the whole structure of sentences with a formal system that brought out all sorts of features of the logical form of arguments. The result was the creation of entire formal languages. A FORMAL LANGUAGE is an artificial language designed to bring out the logical structure of the language and remove all the ambguity and vagueness that plague natural languages like English.

The remainder of this book will develop two formal languages. In each case, parts of the English sentences are replaced with letters and symbols. The goal is to reveal the formal structure of the argument, as we did with these two.

The first is *sentential logic* (SL). In SL, the smallest units are sentences themselves. Simple sentences are represented as letters and connected with logical connectives like *and* and *not* to make more complex sentences.

The second is *quantified logic* (QL). In QL, the basic units are objects, properties of objects, and relations between objects.

When we translate an argument into a formal language, we hope to make its logical structure clearer. We want to include enough of the structure of the English language argument so that we can judge whether the argument is valid or invalid. If we included every feature of the English language, all of the subtlety and nuance, then there would be no advantage in translating to a formal language. We might as well think about the argument in English.

At the same time, we would like a formal language that allows us to represent many kinds of English language arguments. This is one reason to prefer QL to Aristotelean logic; QL can represent every valid argument of Aristotelean logic and more.

So when deciding on a formal language, there is inevitably a tension between wanting to capture as much structure as possible and wanting a simple formal language—simpler formal languages leave out more. This means that there is no perfect formal language. Some will do a better job than others in translating particular English-language arguments.

The languages presented in this book are not the only possible formal languages. However, most nonstandard logics extend on the basic formal structure of the bivalent logics discussed in this book. So this is a good place to start.

Key Terms

- ▷ LOGIC is the study of virtue in argument (p. 2).
- ▷ FORMAL LOGIC is the method for studying argument that uses abstract notation to identify the formal structure of argument (p. 3).
- ▷ A SENTENCE is a unit of language that can be true or false (p. 3).
- ▷ An ARGUMENT is a connected series of sentence, called PREMISES, designed to convince an audience of another sentence, called the CONCLUSION (p. 5).
- ▷ An argument is written in CANONICAL FORM when the premises are each given their own numbered line, the conclusion is separated off by a horizontal line and includes the ∴ symbol, and all indicator words and unnecessary material has been removed (p. 6).
- ▷ An INFERENCE is the connection between the premises and the conclusion of an argument (p. 10).
- ▷ An argument is VALID if it is impossible for the premises to be true and the conclusion false; it is INVALID otherwise (p. 11).
- ▷ An argument is SOUND if it s valid and has true premises (p. 12).
- ▷ An argument is DEDUCTIVE if it purports to be valid, and a person is arguing DEDUCITVELY if they are trying to use valid arguments (p. 12).
- ▷ An argument is STRONG if the truth of the premises would make the conclusion likely to be true (p. 13).
- ▷ An argument is INDUCTIVE if it purports to be cogent (p. 13).
- ▷ An argument is COGENT if it is strong and has true premises (p. 13).
- ▷ A TRUTH VALUE is the status of a sentence regarding its truth (p. 15).
- ▷ A formal language is BIVALENT if there are two truth values and every sentence must have a truth value (p. 15).
- ▷ A TAUTOLOGY is a sentence that must be true, as a matter of logic (p. 16).
- ▷ A CONTRADICTION is a sentence that must be false, as a matter of logic (p. 16).
- ▷ A CONTINGENT SENTENCE is neither a tautology nor a contradiction (p. 16).
- ▷ Two sentences are LOGICALLY EQUIVALENT if they necessarily have the same truth value (p. 17).

▷ A set of sentences is consistent if it is logically possible for all the members of the set to be true at the same time; it is inconsistent otherwise (p. 17).

▷ A formal language is an artificial language designed to make the logical structure of the language explicit and remove ambiguity and vagueness (p. 19).

Practice Exercises

Part A Label the following tautology, contradiction, or contingent sentence.

1. Caesar crossed the Rubicon.
2. Someone once crossed the Rubicon.
3. No one has ever crossed the Rubicon.
4. If Caesar crossed the Rubicon, then someone has.
5. Even though Caesar crossed the Rubicon, no one has ever crossed the Rubicon.
6. If anyone has ever crossed the Rubicon, it was Caesar.

Part B Label the following tautology, contradiction, or contingent sentence.

1. Elephants dissolve in water.
2. Wood is a light, durable substance useful for building things.
3. If wood were a good building material, it would be useful for building things.
4. I live in a three story building that is two stories tall.
5. If gerbils were mammals they would nurse their young.

Part C Which of the following pairs of sentences are logically equivalent?

1.	Elephants dissolve in water.	If you put an elephant in water, it will disintegrate.
2.	All mammals dissolve in water.	If you put an elephant in water, it will disintegrate.
3.	George Bush was the 43rd president.	Barack Obama is the 44th president.
4.	Barack Obama is the 44th president.	Barack Obama was president immediately after the 43rd president.
5.	Elephants dissolve in water.	All mammals dissolve in water.

Part D Which of the following pairs of sentences are logically equivalent?

1. Thelonious Monk played piano.	John Coltrane played tenor sax.
2. Thelonious Monk played gigs with John Coltrane.	John Coltrane played gigs with Thelonious Monk.
3. All professional piano players have big hands.	Piano player Bud Powell had big hands.
4. Bud Powell suffered from severe mental illness.	All piano players suffer from severe mental illness.
5. John Coltrane was deeply religious.	John Coltrane viewed music as an expression of spirituality.

Part E

Consider again sentences G1–G4 on p.17.

G1 There are at least four giraffes at the wild animal park.
G2 There are exactly seven gorillas at the wild animal park.
G3 There are not more than two Martians at the wild animal park.
G4 Every giraffe at the wild animal park is a Martian.

Now consider each of the following sets of sentences. Which are consistent? Which are inconsistent?

1. G2, G3, and G4
2. G1, G3, and G4
3. G1, G2, and G4
4. G1, G2, and G3

Part F Consider the following set of sentences.

S1 All people are mortal.
S2 Socrates is a person.
S3 Socrates will never die.
S4 Socrates is mortal.

Which combinations of sentences form consistent sets? Mark each consistent or inconsistent.

1. S1, S2, and S3
2. S2, S3, and S4
3. S2 and S3.
4. S1 and S4
5. S1, S2, S3, and S4

Part G Which of the following is possible? If it is possible, give an example. If it is not possible, explain why.

1. A valid argument that has one false premise and one true premise
2. A valid argument that has a false conclusion
3. A valid argument, the conclusion of which is a contradiction
4. An invalid argument, the conclusion of which is a tautology
5. A tautology that is contingent
6. Two logically equivalent sentences, both of which are tautologies
7. Two logically equivalent sentences, one of which is a tautology and one of which is contingent
8. Two logically equivalent sentences that together are an inconsistent set
9. A consistent set of sentences that contains a contradiction
10. An inconsistent set of sentences that contains a tautology

Part H Which of the following is possible? If it is possible, give an example. If it is not possible, explain why.

1. A valid argument, whose premises are all tautologies, and whose conclusion is contingent
2. A valid argument with true premises and a false conclusion
3. A consistent set of sentences that contains two sentences that are not logically equivalent
4. A consistent set of sentences, all of which are contingent
5. A false tautology
6. A valid argument with false premises
7. A logically equivalent pair of sentences that are not consistent
8. A tautological contradiction
9. A consistent set of sentences that are all contradictions

Chapter 2

Sentential Logic

This chapter introduces a logical language called SL. It is a version of *sentential logic*, because the basic units of the language will represent entire sentences.

2.1 Sentence Letters

In SL, capital letters, called SENTENCE LETTERS are used to represent basic sentences. Considered only as a symbol of SL, the letter A could mean any sentence. So when translating from English into SL, it is important to provide a symbolization key, or dictionary. The SYMBOLIZATION KEY provides an English language sentence for each sentence letter used in the symbolization.

For example, consider this argument:

There is an apple on the desk.
If there is an apple on the desk, then Jenny made it to class.
$\therefore$ Jenny made it to class.

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

A: There is an apple on the desk.
B: If there is an apple on the desk, then Jenny made it to class.
C: Jenny made it to class.

We would then symbolize the argument in this way:

$$\begin{array}{rl} & A \\ & B \\ \therefore & C \end{array}$$

There is no necessary connection between some sentence A, which could be any sentence, and some other sentences B and C, which could be any sentences. The structure of the argument has been completely lost in this translation.

The important thing about the argument is that the second premise is not merely *any* sentence, logically divorced from the other sentences in the argument. The second premise contains the first premise and the conclusion *as parts*. Our symbolization key for the argument only needs to include meanings for A and C, and we can build the second premise from those pieces. So we symbolize the argument this way:

$$\begin{array}{rl} & A \\ & \text{If } A\text{, then } C. \\ \therefore & C \end{array}$$

This preserves the structure of the argument that makes it valid, but it still makes use of the English expression "If… then…." Although we ultimately want to replace all of the English expressions with logical notation, this is a good start.

The individual sentence letters in SL are called atomic sentences, because they are the basic building blocks out of which more complex sentences can be built. We can identify atomic sentences in English as well. An ATOMIC SENTENCE is one that cannot be broken into parts that are themselves sentences. "There is an apple on the desk" is an atomic sentence in English, because you can't find any proper part of it that forms a complete sentence. For instance "an apple on the desk" is a noun phrase, not a complete sentence. Similarly "on the desk" is a prepositional phrase, and not a sentence, and "is an" is not any kind of phrase at all. This is what you will find no matter how you divide "There is an apple on the desk." On the other hand you can find two proper parts of "If there is an apple on the desk, then Jenny made it to class" that are complete sentences: "There is an apple on the desk" and "Jenny made it to class." As a general rule, we will want to use atomic sentences in SL (that is, the sentence letters) to represent atomic sentences in English. Otherwise, we will lose some of the logical structure of the English sentence, as we have just seen.

There are only 26 letters of the alphabet, but there is no logical limit to the number of atomic sentences. We can use the same letter to symbolize different atomic sentences by adding a subscript, a small number written after the letter. We could have a symbolization key that looks like this:

A_1: The apple is under the armoire.
A_2: Arguments in SL always contain atomic sentences.
A_3: Adam Ant is taking an airplane from Anchorage to Albany.
⋮
A_{294}: Alliteration angers otherwise affable astronauts.

Keep in mind that each of these is a different sentence letter. When there are subscripts in the symbolization key, it is important to keep track of them.

2.2 Sentential Connectives

Logical connectives are used to build complex sentences from atomic components. In SL, our logical connectives are called SENTENTIAL CONNECTIVES because they connect sentence letters. There are five sentential connectives in SL. This table summarizes them, and they are explained below.

Symbol	What it is called	What it means
$\sim$	negation	"It is not the case that. . ."
&	conjunction	"Both . . . and . . ."
$\vee$	disjunction	"Either . . . or . . ."
$\rightarrow$	conditional	"If . . . then . . ."
$\leftrightarrow$	biconditional	". . . if and only if . . ."

Negation

Consider how we might symbolize these sentences:

1. Mary is in Barcelona.
2. Mary is not in Barcelona.
3. Mary is somewhere other than Barcelona.

In order to symbolize sentence 1, we will need one sentence letter. We can provide a symbolization key:

B: Mary is in Barcelona.

Note that here we are giving B a different interpretation than we did in the previous section. The symbolization key only specifies what B means *in a specific context.* It is vital that we continue to use this meaning of B so long as we are talking about Mary and Barcelona. Later, when we are symbolizing different sentences, we can write a new symbolization key and use B to mean something else.

Now, sentence 1 is simply B. Sentence 2 is obviously related to sentence 1: it is basically 1 with a "not" added. We could put the sentence partly our symbolic language by writing "Not B." This means we do not want to introduce a different sentence letter for 2. We just need a new symbol for the "not" part. Let's use '$\sim$.' Now we can translate 'Not B' to $\sim B$.

Sentence 3 is about whether or not Mary is in Barcelona, but it does not contain the word "not." Nevertheless, it is obviously logically equivalent to sentence 2. They both mean: It is not the case that Mary is in Barcelona. As such, we can translate both sentence 2 and sentence 3 as $\sim B$.

A sentence can be symbolized as $\sim\mathscr{A}$ if it can be paraphrased in English as "It is not the case that $\mathscr{A}$."

Consider these further examples:

4. The widget can be replaced if it breaks.
5. The widget is irreplaceable.
6. The widget is not irreplaceable.

If we let R mean "The widget is replaceable", then sentence 4 can be translated as R.

What about sentence 5? Saying the widget is irreplaceable means that it is not the case that the widget is replaceable. So even though sentence 5 is not negative in English, we symbolize it using negation as $\sim R$.

Sentence 6 can be paraphrased as "It is not the case that the widget is irreplaceable." Using negation twice, we translate this as $\sim\sim R$. 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." If you think about the sentence in English, it is logically equivalent to sentence 4. So when we define logical equivalence in SL, we will make sure that R and $\sim\sim R$ are logically equivalent.

More examples:

7. Elliott is happy.

8. Elliott is unhappy.

If we let H mean "Elliot is happy", then we can symbolize sentence 7 as H.

However, it would be a mistake to symbolize sentence 8 as $\sim H$. If Elliott is unhappy, then he is not happy—but sentence 8 does not mean the same thing as "It is not the case that Elliott is happy." It could be that he is not happy but that he is not unhappy either. Perhaps he is somewhere between the two. In order to symbolize sentence 8, we would need a new sentence letter.

For any sentence $\mathcal{A}$: If $\mathcal{A}$ is true, then $\sim\mathcal{A}$ is false. If $\sim\mathcal{A}$ is true, then $\mathcal{A}$ is false. Using T for true and F for false, we can summarize this in a *characteristic truth table* for negation:

$\mathcal{A}$	$\sim\mathcal{A}$
T	F
F	T

We will discuss truth tables at greater length in the next chapter.

Conjunction

Consider these sentences:

9. Adam is athletic.
10. Barbara is athletic.
11. Adam is athletic, and Barbara is also athletic.

We will need separate sentence letters for 9 and 10, so we define this symbolization key:

A: Adam is athletic.
B: Barbara is athletic.

Sentence 9 can be symbolized as A.

Sentence 10 can be symbolized as B.

Sentence 11 can be paraphrased as "A and B." In order to fully symbolize this sentence, we need another symbol. We will use & . We translate "A and B" as $A \& B$. The logical connective & is called CONJUNCTION, and A and B are each called CONJUNCTS.

Notice that we make no attempt to symbolize "also" in sentence 11. Words like "both" and "also" function to draw our attention to the fact that two things are being conjoined. They are not doing any further logical work, so we do not need to represent them in SL.

Some more examples:

12. Barbara is athletic and energetic.
13. Barbara and Adam are both athletic.
14. Although Barbara is energetic, she is not athletic.
15. Barbara is athletic, but Adam is more athletic than she is.

Sentence 12 is obviously a conjunction. The sentence says two things about Barbara, so in English it is permissible to refer to Barbara only once. It might be tempting to try this when translating the argument: Since B means "Barbara is athletic", one might paraphrase the sentences as "B and energetic." This would be a mistake. Once we translate part of a sentence as B, any further structure is lost. B is an atomic sentence; it is nothing more than true or false. Conversely, "energetic" is not a sentence; on its own it is neither true nor false. We should instead paraphrase the sentence as "B and Barbara is energetic." Now we need to add a sentence letter to the symbolization key. Let E mean "Barbara is energetic." Now the sentence can be translated as $B \,\&\, E$.

A sentence can be symbolized as $\mathcal{A} \,\&\, \mathcal{B}$ if it can be paraphrased in English as 'Both $\mathcal{A}$, and $\mathcal{B}$.' Each of the conjuncts must be a sentence.

Sentence 13 says one thing about two different subjects. It says of both Barbara and Adam that they are athletic, and in English we use the word "athletic" only once. In translating to SL, it is important to realize that the sentence can be paraphrased as, "Barbara is athletic, and Adam is athletic." This translates as $B \,\&\, A$.

Sentence 14 is a bit more complicated. The word "although" sets up a contrast between the first part of the sentence and the second part. Nevertheless, the sentence says both that Barbara is energetic and that she is not athletic. In order to make each of the conjuncts an atomic sentence, we need to replace "she" with "Barbara."

So we can paraphrase sentence 14 as, "*Both* Barbara is energetic, *and* Barbara is not athletic." The second conjunct contains a negation, so we paraphrase further: "*Both* Barbara is energetic *and it is not the case that* Barbara is athletic." This translates as $E \,\&\, {\sim}B$.

Sentence 15 contains a similar contrastive structure. It is irrelevant for the purpose of translating to SL, so we can paraphrase the sentence as "*Both* Barbara is athletic, *and* Adam is more athletic than Barbara." (Notice that we once again replace the pronoun "she" with her name.) How should we translate the second conjunct? We already have the sentence letter A which is about Adam's being athletic and B which is about Barbara's being athletic, but neither is about one of them being more athletic than the other. We need a new sentence letter. Let R mean "Adam is more athletic than Barbara." Now the sentence translates as $B \,\&\, R$.

Sentences that can be paraphrased "$\mathcal{A}$, but $\mathcal{B}$" or "Although $\mathcal{A}$, $\mathcal{B}$" are best symbolized using conjunction $\mathcal{A}\ \&\, \mathcal{B}$.

It is important to keep in mind that the sentence letters A, B, and R are atomic sentences. Considered as symbols of SL, they have no meaning beyond being true or false. We have used them to symbolize different English language sentences that are all about people being athletic, but this similarity is completely lost when we translate to SL. No formal language can capture all the structure of the English language, but as long as this structure is not important to the argument there is nothing lost by leaving it out.

For any sentences $\mathcal{A}$ and $\mathcal{B}$, $\mathcal{A}\ \&\, \mathcal{B}$ is true if and only if both $\mathcal{A}$ and $\mathcal{B}$ are true. We can summarize this in the characteristic truth table for conjunction:

$\mathcal{A}$	$\mathcal{B}$	$\mathcal{A}\ \&\, \mathcal{B}$
T	T	T
T	F	F
F	T	F
F	F	F

Conjunction is symmetrical because we can swap the conjuncts without changing the truth value of the sentence. Regardless of what $\mathcal{A}$ and $\mathcal{B}$ are, $\mathcal{A} \,\&\, \mathcal{B}$ is logically equivalent to $\mathcal{B}\ \&\, \mathcal{A}$.

Disjunction

Consider these sentences:

16. Either Denison will play golf with me, or he will watch movies.
17. Either Denison or Ellery will play golf with me.

For these sentences we can use this symbolization key:

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

Sentence 16 is "Either D or M." To fully symbolize this, we introduce a new symbol. The sentence becomes $D \vee M$. The $\vee$connective is called DISJUNCTION, and D and M are called DISJUNCTS.

Sentence 17 is only slightly more complicated. There are two subjects, but the English sentence only gives the verb once. In translating, we can paraphrase it as "Either Denison will play golf with me, or Ellery will play golf with me." Now it obviously translates as $D \vee E$.

> A sentence can be symbolized as $\mathcal{A} \vee \mathcal{B}$ if it can be paraphrased in English as "Either $\mathcal{A}$ or $\mathcal{B}$." Each of the disjuncts must be a sentence.

Sometimes in English, the word "or" excludes the possibility that both disjuncts are true. This is called an EXCLUSIVE OR. An *exclusive or* is clearly intended when a restaurant menu says, "Entrees come with either soup or salad." You may have soup; you may have salad; but, if you want *both* soup *and* salad, then you will have to pay extra.

At other times, the word "or" allows for the possibility that both disjuncts might be true. This is probably the case with sentence 17, above. I might play with Denison, with Ellery, or with both Denison and Ellery. Sentence 17 merely says that I will play with *at least* one of them. This is called an INCLUSIVE OR.

The symbol $\vee$represents an *inclusive or.* 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 characteristic truth table for disjunction:

$\mathcal{A}$	$\mathcal{B}$	$\mathcal{A} \vee\mathcal{B}$
T	T	T
T	F	T
F	T	T
F	F	F

Like conjunction, disjunction is symmetrical. $\mathcal{A} \vee\mathcal{B}$ is logically equivalent to $\mathcal{B}$ $\vee\mathcal{A}$.

These sentences are somewhat more complicated:

18. Either you will not have soup, or you will not have salad.

19. You will have neither soup nor salad.
20. You get either soup or salad, but not both.

We let S_1 mean that you get soup and S_2 mean that you get salad.

Sentence 18 can be paraphrased in this way: "Either *it is not the case that* you get soup, or *it is not the case that* you get salad." Translating this requires both disjunction and negation. It becomes $\sim S_1 \vee \sim S_2$.

Sentence 19 also requires negation. It can be paraphrased as, "*It is not the case that* either you get soup or you get salad." We need some way of indicating that the negation does not just negate the right or left disjunct, but rather negates the entire disjunction. In order to do this, we put parentheses around the disjunction: "It is not the case that $(S_1 \vee S_2)$." This becomes simply $\sim(S_1 \vee S_2)$.

Notice that the parentheses are doing important work here. The sentence $\sim S_1 \vee S_2$ would mean "Either you will not have soup, or you will have salad."

Sentence 20 is an *exclusive or.* We can break the sentence into two parts. The first part says that you get one or the other. We translate this as $(S_1 \vee S_2)$. The second part says that you do not get both. We can paraphrase this as "It is not the case both that you get soup and that you get salad." Using both negation and conjunction, we translate this as $\sim(S_1 \,\&\, S_2)$. Now we just need to put the two parts together. As we saw above, "but" can usually be translated as a conjunction. Sentence 20 can thus be translated as $(S_1 \vee S_2) \,\&\, \sim(S_1 \,\&\, S_2)$.

Although $\vee$is an *inclusive or*, we can symbolize an *exclusive or* in SL. We just need more than one connective to do it.

Conditional

For the following sentences, let R mean "You will cut the red wire" and B mean "The bomb will explode."

21. If you cut the red wire, then the bomb will explode.
22. The bomb will explode only if you cut the red wire.

Sentence 21 can be translated partially as "If R, then B." We will use the symbol $\rightarrow$to represent logical entailment. The sentence becomes $R \rightarrow B$. The connective is called a CONDITIONAL. The sentence on the left-hand side of the conditional (R in this example) is called the ANTECEDENT. The sentence on the right-hand side (B) is called the CONSEQUENT.

Sentence 22 is also a conditional. Since the word "if" appears in the second half of the sentence, it might be tempting to symbolize this in the same way as sentence 21. That would be a mistake.

The conditional $R \rightarrow B$ says that *if* R were true, *then* B would also be true. It does not say that you cutting the red wire is the *only* way that the bomb could explode. Someone else might cut the wire, or the bomb might be on a timer. The sentence $R \rightarrow B$ does not say anything about what to expect if R is false. Sentence 22 is different. It says that the only conditions under which the bomb will explode involve you having cut the red wire; i.e., if the bomb explodes, then you must have cut the wire. As such, sentence 22 should be symbolized as $B \rightarrow R$.

It is important to remember that the connective $\rightarrow$says only that, if the antecedent is true, then the consequent is true. It says nothing about the *causal* connection between the two events. Translating sentence 22 as $B \rightarrow R$ does not mean that the bomb exploding would somehow have caused you cutting the wire. Both sentence 21 and 22 suggest that, if you cut the red wire, you cutting the red wire would be the cause of the bomb exploding. They differ on the *logical* connection. If sentence 22 were true, then an explosion would tell us—those of us safely away from the bomb—that you had cut the red wire. Without an explosion, sentence 22 tells us nothing.

The paraphrased sentence "$\mathcal{A}$ only if $\mathcal{B}$" is logically equivalent to "If $\mathcal{A}$, then $\mathcal{B}$."

"If $\mathcal{A}$, then $\mathcal{B}$" means that if $\mathcal{A}$ is true, then so is $\mathcal{B}$. So we know that if the antecedent $\mathcal{A}$ is true but the consequent $\mathcal{B}$ is false, then the conditional "If $\mathcal{A}$ then $\mathcal{B}$" is false. What is the truth value of "If $\mathcal{A}$, then $\mathcal{B}$" under other circumstances? Suppose, for instance, that the antecedent $\mathcal{A}$ happened to be false. "If $\mathcal{A}$, then $\mathcal{B}$" would then not tell us anything about the actual truth value of the consequent $\mathcal{B}$, and it is unclear what the truth value of "If $\mathcal{A}$, then $\mathcal{B}$" would be.

In English, the truth of conditionals often depends on what *would* be the case if the antecedent *were true*—even if, as a matter of fact, the antecedent is false. This poses a problem for translating conditionals into SL. Considered as sentences of SL, R and B in the above examples have nothing intrinsic to do with each other. In order to consider what the world would be like if R were true, we would need to analyze what R says about the world. Since R is an atomic symbol of SL, however, there is no further structure to be analyzed. When we replace a sentence with a sentence letter, we consider it merely as some atomic sentence that might be true or false.

In order to translate conditionals into SL, we will not try to capture all the subtleties of the English language "If…, then…." Instead, the symbol →will be what logicians call a material conditional. This means that when $\mathcal{A}$ is false, the conditional $\mathcal{A} \rightarrow \mathcal{B}$ is automatically true, regardless of the truth value of $\mathcal{B}$. If both $\mathcal{A}$ and $\mathcal{B}$ are true, then the conditional $\mathcal{A} \rightarrow \mathcal{B}$ is true.

In short, $\mathcal{A} \rightarrow \mathcal{B}$ is false if and only if $\mathcal{A}$ is true and $\mathcal{B}$ is false. We can summarize this with a characteristic truth table for the conditional.

$\mathcal{A}$	$\mathcal{B}$	$\mathcal{A}\rightarrow\mathcal{B}$
T	T	T
T	F	F
F	T	T
F	F	T

The conditional is asymmetrical. You cannot swap the antecedent and consequent without changing the meaning of the sentence, because $\mathcal{A} \rightarrow \mathcal{B}$ and $\mathcal{B} \rightarrow \mathcal{A}$ are not logically equivalent.

Not all sentences of the form "If…, then…" are conditionals. Consider this sentence:

23. If anyone wants to see me, then I will be on the porch.

When I say this, it means that I will be on the porch, regardless of whether anyone wants to see me or not—but if someone did want to see me, then they should look for me there. If we let P mean "I will be on the porch," then sentence 23 can be translated simply as P.

Biconditional

Consider these sentences:

24. The figure on the board is a triangle only if it has exactly three sides.
25. The figure on the board is a triangle if it has exactly three sides.
26. The figure on the board is a triangle if and only if it has exactly three sides.

Let T mean "The figure is a triangle" and S mean "The figure has three sides."

Sentence 24, for reasons discussed above, can be translated as $T \rightarrow S$.

Sentence 25 is importantly different. It can be paraphrased as "If the figure has three sides, then it is a triangle." So it can be translated as $S \to T$.

Sentence 26 says that T is true *if and only if* S is true; we can infer S from T, and we can infer T from S. This is called a BICONDITIONAL, because it entails the two conditionals $S \to T$ and $T \to S$. We will use $\leftrightarrow$to represent the biconditional; sentence 26 can be translated as $S \leftrightarrow T$.

We could abide without a new symbol for the biconditional. Since sentence 26 means "$T \to S$ and $S \to T$," we could translate it as $(T \to S) \& (S \to T)$. We would need parentheses to indicate that $(T \to S)$ and $(S \to T)$ are separate conjuncts; the expression $T \to S \& S \to T$ would be ambiguous.

Because we could always write $(\mathcal{A} \to \mathcal{B}) \& (\mathcal{B} \to \mathcal{A})$ instead of $\mathcal{A} \leftrightarrow \mathcal{B}$, we do not strictly speaking *need* to introduce a new symbol for the biconditional. Nevertheless, logical languages usually have such a symbol. SL will have one, which makes it easier to translate phrases like "if and only if."

$\mathcal{A} \leftrightarrow \mathcal{B}$ is true if and only if $\mathcal{A}$ and $\mathcal{B}$ have the same truth value. This is the characteristic truth table for the biconditional:

$\mathcal{A}$	$\mathcal{B}$	$\mathcal{A} \leftrightarrow \mathcal{B}$
T	T	T
T	F	F
F	T	F
F	F	T

2.3 Other Symbolization

We have now introduced all of the connectives of SL. We can use them together to translate many kinds of sentences. Consider these examples of sentences that use the English-language connective "unless":

27. Unless you wear a jacket, you will catch cold.
28. You will catch cold unless you wear a jacket.

Let J mean "You will wear a jacket" and let D mean "You will catch a cold."

We can paraphrase sentence 27 as "Unless J, D." This means that if you do not wear a jacket, then you will catch cold; with this in mind, we might translate it as $\sim J \to D$. It also means that if you do not catch a cold, then you must have worn a jacket; with this in mind, we might translate it as $\sim D \to J$.

Which of these is the correct translation of sentence 27? Both translations are correct, because the two translations are logically equivalent in SL.

Sentence 28, in English, is logically equivalent to sentence 27. It can be translated as either $\sim J \rightarrow D$ or $\sim D \rightarrow J$.

When symbolizing sentences like sentence 27 and sentence 28, it is easy to get turned around. Since the conditional is not symmetric, it would be wrong to translate either sentence as $J \rightarrow \sim D$. Fortunately, there are other logically equivalent expressions. Both sentences mean that you will wear a jacket or—if you do not wear a jacket—then you will catch a cold. So we can translate them as $J \vee D$. (You might worry that the "or" here should be an *exclusive or*. However, the sentences do not exclude the possibility that you might *both* wear a jacket *and* catch a cold; jackets do not protect you from all the possible ways that you might catch a cold.)

> If a sentence can be paraphrased as "Unless $\mathcal{A}$, $\mathcal{B}$," then it can be symbolized as $\mathcal{A} \vee \mathcal{B}$.

Symbolization of standard sentence types is summarized on p. 167.

2.4 Sentences of Sentential Logic

The sentence "Apples are red, or berries are blue" is a sentence of English, and the sentence "$(A \vee B)$" is a sentence of SL. When we defined sentences in English, we did so using the concept of truth. Sentences were units of language that can be true or false. In SL, it is possible to define what counts as a sentence without talking about truth. Instead, we can just talk about the structure of the sentence. This is one respect in which a formal language like SL is more precise than a natural language like English.

The structure of a sentence in SL considered without reference to truth or falsity is called its syntax. More generally SYNTAX refers to the study of the properties of language that are there even when you don't consider meaning. Whether a sentence is true or false is considered part of its meaning. In this chapter, we will be giving a purely syntactical definition of a sentence in SL. The contrasting term is SEMANTICS, the study of aspects of language that relate to meaning, including truth and falsity. (The word "semantics" comes from the Greek word for "mark")

It is important to distinguish between the logical language SL, which we are developing, and the language that we use to talk about SL. When we talk about a language, the language that we are talking about is called the OBJECT

LANGUAGE. The language that we use to talk about the object language is called the METALANGUAGE.

The object language in this chapter is SL. The metalanguage is English—not conversational English, but English supplemented with some logical and mathematical vocabulary. The sentence "$(A \vee B)$" is a sentence in the object language, because it uses only symbols of SL. The word "sentence" is not itself part of SL, however, so the sentence "This expression is a sentence of SL" is not a sentence of SL. It is a sentence in the metalanguage, a sentence that we use to talk *about* SL.

An important part of our metalanguage are the METAVARIABLES. These are the fancy script letters we have been using in the characteristic truth tables for the connectives: $\mathcal{A}$, $\mathcal{B}$, $\mathcal{C}$, etc. These are letters that can refer to any sentence in SL. They can represent sentences like P or Q, or they can represent longer sentences, like $(((A \vee B) \,\&\, G) \rightarrow (P \leftrightarrow Q))$. Just the sentence letters A, B, etc. are variables that range over any English sentence, the metavariables $\mathcal{A}$, $\mathcal{B}$, etc. are variables that range over any sentence in SL, including the sentence letters A, B, etc.

As we said, in this chapter we will give a syntactic definition for "sentence of SL." The definition itself will be given in mathematical English, the metalanguage.

Expressions

There are three kinds of symbols in SL:

sentence letters with subscripts, as needed	$A, B, C, \ldots, Z$ $A_1, B_1, Z_1, A_2, A_{25}, J_{375}, \ldots$
connectives	$\sim, \&, \vee, \rightarrow, \leftrightarrow$
parentheses	(,)

We define an EXPRESSION OF SL as any string of symbols of SL. Take any of the symbols of SL and write them down, in any order, and you have an expression.

Sentences

Since any sequence of symbols is an expression, many expressions of SL will be gobbledegook. We are interested in identifying the expressions that are capable of having meaning. Since the smallest meaningful unit of language in SL is the sentence, we need to define a sentence.

Obviously, individual sentence letters like A and G_{13} will be sentences. We can form further sentences out of these by using the various connectives. Using negation, we can get $\sim A$ and $\sim G_{13}$. Using conjunction, we can get $A \,\&\, G_{13}$, $G_{13} \,\&\, A$, $A \,\&\, A$, and $G_{13} \,\&\, G_{13}$. We could also apply negation repeatedly to get sentences like $\sim\sim A$ or apply negation along with conjunction to get sentences like $\sim(A \,\&\, G_{13})$ and $\sim(G_{13} \,\&\, \sim G_{13})$. The possible combinations are endless, even starting with just these two sentence letters, and there are infinitely many sentence letters. So there is no point in trying to list all the sentences.

Instead, we will describe the process by which sentences can be constructed. Consider negation: Given any sentence $\mathcal{A}$ of SL, $\sim\mathcal{A}$ is a sentence of SL. It is important here that $\mathcal{A}$ is not the sentence letter A. Rather, it is a metavariable: part of the metalanguage, not the object language. Since $\mathcal{A}$ is not a symbol of SL, $\sim\mathcal{A}$ is not an expression of SL. Instead, it is an expression of the metalanguage that allows us to talk about infinitely many expressions of SL: all of the expressions that start with the negation symbol.

We can say similar things for each of the other connectives. For instance, if $\mathcal{A}$ and $\mathcal{B}$ are sentences of SL, then $(\mathcal{A} \,\&\, \mathcal{B})$ is a sentence of SL. Providing clauses like this for all of the connectives, we arrive at the following formal definition for a well-formed formula of SL:

1. Every atomic sentence is a sentence.
2. If $\mathcal{A}$ is a sentence, then $\sim\mathcal{A}$ is a sentence of SL.
3. If $\mathcal{A}$ and $\mathcal{B}$ are sentences, then $(\mathcal{A} \,\&\, \mathcal{B})$ is a sentence.
4. If $\mathcal{A}$ and $\mathcal{B}$ are sentences, then $(\mathcal{A} \vee \mathcal{B})$ is a sentence.
5. If $\mathcal{A}$ and $\mathcal{B}$ are sentences, then $(\mathcal{A} \rightarrow \mathcal{B})$ is a sentence.
6. If $\mathcal{A}$ and $\mathcal{B}$ are sentences, then $(\mathcal{A} \leftrightarrow \mathcal{B})$ is a sentence.
7. All and only sentences of SL can be generated by applications of these rules.

Notice that we cannot immediately apply this definition to see whether an arbitrary expression is a sentence. Suppose we want to know whether or not $\sim\sim\sim D$ is a sentence of SL. Looking at the second clause of the definition, we know that $\sim\sim\sim D$ is a sentence *if* $\sim\sim D$ is a sentence. So now we need to ask whether or not $\sim\sim D$ is a sentence. Again looking at the second clause of the definition, $\sim\sim D$ is a sentence *if* $\sim D$ is. Again, $\sim D$ is a sentence *if* D is a sentence. Now D is a sentence letter, an atomic sentence of SL, so we know that D is a sentence by the first clause of the definition. So for a compound formula like $\sim\sim\sim D$, we must apply the definition repeatedly. Eventually we arrive at the atomic sentences from which the sentence is built up.

Definitions like this are called recursive. RECURSIVE DEFINITIONS begin with some specifiable base elements and define ways to indefinitely compound the base elements. Just as the recursive definition allows complex sentences to be built up from simple parts, you can use it to decompose sentences into their simpler parts. To determine whether or not something meets the definition, you may have to refer back to the definition many times.

The last connective that you add when you assemble a sentence using the recursive definition is the MAIN CONNECTIVE of that sentence. For example: The main logical operator of $\sim(E \vee (F \rightarrow G))$ is negation, $\sim$. The main logical operator of $(\sim E \vee (F \rightarrow G))$ is disjunction, $\vee$.

The recursive structure of sentences in SL will be important when we consider the circumstances under which a particular sentence would be true or false. The sentence $\sim\sim\sim D$ is true if and only if the sentence $\sim\sim D$ is false, and so on through the structure of the sentence until we arrive at the atomic components: $\sim\sim\sim D$ is true if and only if the atomic sentence D is false. We will return to this point in the next chapter.

Notational conventions

A sentence like $(Q \,\&\, R)$ must be surrounded by parentheses, because we might apply the definition again to use this as part of a more complicated sentence. If we negate $(Q \,\&\, R)$, we get $\sim(Q \,\&\, R)$. If we just had $Q \,\&\, R$ without the parentheses and put a negation in front of it, we would have $\sim Q \,\&\, R$. It is most natural to read this as meaning the same thing as $(\sim Q \,\&\, R)$, something very different than $\sim(Q \,\&\, R)$. The sentence $\sim(Q \,\&\, R)$ 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 $(\sim Q \,\&\, R)$ means specifically that Q is false and that R is true. As such, parentheses are crucial to the meaning of the sentence.

So, strictly speaking, $Q \,\&\, R$ without parentheses is *not* a sentence of SL. When using SL, however, we will often be able to relax the precise definition so as to make things easier for ourselves. We will do this in several ways.

First, we understand that $Q \,\&\, R$ means the same thing as $(Q \,\&\, R)$. As a matter of convention, we can leave off parentheses that occur *around the entire sentence*.

Second, it can sometimes be confusing to look at long sentences with many nested pairs of parentheses. We adopt the convention of using square brackets [and] in place of parentheses. There is no logical difference between $(P \vee Q)$ and $[P \vee Q]$, for example. The unwieldy sentence

$$(((H \rightarrow I) \vee (I \rightarrow H)) \,\&\, (J \vee K))$$

could be written in this way:

$$\big[(H \to I) \vee (I \to H)\big] \,\&\, (J \vee K)$$

Third, we will sometimes want to translate the conjunction of three or more sentences. For the sentence “Alice, Bob, and Candice all went to the party,” suppose we let A mean “Alice went,” B mean “Bob went,” and C mean “Candice went.” The definition only allows us to form a conjunction out of two sentences, so we can translate it as $(A \,\&\, B) \,\&\, C$ or as $A \,\&\, (B \,\&\, C)$. There is no reason to distinguish between these, since the two translations are logically equivalent. There is no logical difference between the first, in which $(A \,\&\, B)$ is conjoined with C, and the second, in which A is conjoined with $(B \,\&\, C)$. So we might as well just write $A \,\&\, B \,\&\, C$. As a matter of convention, we can leave out parentheses when we conjoin three or more sentences.

Fourth, a similar situation arises with multiple disjunctions. “Either Alice, Bob, or Candice went to the party” can be translated as $(A \vee B) \vee C$ or as $A \vee (B \vee C)$. Since these two translations are logically equivalent, we may write $A \vee B \vee C$.

These latter two conventions only apply to multiple conjunctions or multiple disjunctions. If a series of connectives includes both disjunctions and conjunctions, then the parentheses are essential; as with $(A \,\&\, B) \vee C$ and $A \,\&\, (B \vee C)$. The parentheses are also required if there is a series of conditionals or biconditionals; as with $(A \to B) \to C$ and $A \leftrightarrow (B \leftrightarrow C)$.

We have adopted these four rules as notational conventions, not as changes to the definition of a sentence. Strictly speaking, $A \vee B \vee C$ is still not a sentence. Instead, it is a kind of shorthand. We write it for the sake of convenience, but we really mean the sentence $(A \vee (B \vee C))$.

If we had given a different definition for a sentence, then these could count as sentences. We might have written rule 3 in this way: “If $\mathcal{A}$, $\mathcal{B}$, $\ldots$ $\mathcal{Z}$ are sentences, then $(\mathcal{A} \,\&\, \mathcal{B} \,\&\, \ldots \,\&\, \mathcal{Z})$, is a sentence .” This would make it easier to translate some English sentences, but would have the cost of making our formal language more complicated. We would have to keep the complex definition in mind when we develop truth tables and a proof system. We want a logical language that is expressively simple and allows us to translate easily from English, but we also want a formally simple language. Adopting notational conventions is a compromise between these two desires.

Key Terms

▷ A SENTENCE LETTER is a single capital letter, used in SL to represent a basic sentence (p. 24).

▷ A symbolization key is list that shows which English sentences are represented by which sentence letters in SL (p. 24).

▷ An atomic sentence is a sentence that does not have any sentences as proper parts (p. 25).

▷ A sentential connective is a logical operator in SL used to combine sentence letters into larger sentences (p. 26).

▷ The negation, $\sim$, is used to represent words and phrases that function like the English word "not" (p. 27).

▷ The conjunction, & , is used to represent words and phrases that function like the English word "and." The sentences joined by the conjunction are called conjuncts (p. 28).

▷ The disjunction, $\vee$, is used to represent words and phrases that function like the English word "or" in its inclusive sense (p. 31).

▷ The conditional, $\rightarrow$, is used to represent words and phrases that function like the English phrase "if . . . then" (p. 32). The sentence on its left is called the antecedent and the sentence on its right is called the consequent

▷ The biconditional, $\leftrightarrow$, is used to represent words and phrases that function like the English phrase "if and only if" (p. 35).

▷ The syntax of a bit of language is its structure, considered without reference to truth, falsity, or meaning (p. 36).

▷ The semantics of a bit of language is its meaning, including truth and falsity (p. 36).

▷ The object language is a language that is constructed and studied by logicians. In this textbook, the object languages are SL and QL (p. 37).

▷ The metalanguage is the language logicians use to talk about the object language. In this textbook, the metalanguage is English, supplemented by certain symbols like metavariables and technical terms like "valid" (p. 37).

▷ metavariables are variables in the metalanguage that can represent any sentence in the object language (p. 37).

▷ An expression of sl is any string of symbols in any order (p. 37).

▷ A sentence of sl is an expression that can be formed using the recursive definition of a sentence (p. 38).

▷ a recursive definition is one that defines a term by identifying base class and rules for extending that class (p. 39).

▷ The MAIN CONNECTIVE of a sentence is the last connective that you add when you assemble a sentence using the recursive definition (p. 39).

Practice Exercises

Part A Using the symbolization key given, translate each English-language sentence into SL.

M: Those creatures are men in suits.
C: Those creatures are chimpanzees.
G: Those creatures are gorillas.

1. Those creatures are not men in suits.
2. Those creatures are men in suits, or they are not.
3. Those creatures are either gorillas or chimpanzees.
4. Those creatures are neither gorillas nor chimpanzees.
5. If those creatures are chimpanzees, then they are neither gorillas nor men in suits.
6. Unless those creatures are men in suits, they are either chimpanzees or they are gorillas.

Part B Using the symbolization key given, translate each English-language sentence into SL.

A: Mister Ace was murdered.
B: The butler did it.
C: The cook did it.
D: The Duchess is lying.
E: Mister Edge was murdered.
F: The murder weapon was a frying pan.

1. Either Mister Ace or Mister Edge was murdered.
2. If Mister Ace was murdered, then the cook did it.
3. If Mister Edge was murdered, then the cook did not do it.
4. Either the butler did it, or the Duchess is lying.
5. The cook did it only if the Duchess is lying.
6. If the murder weapon was a frying pan, then the culprit must have been the cook.
7. If the murder weapon was not a frying pan, then the culprit was either the cook or the butler.
8. Mister Ace was murdered if and only if Mister Edge was not murdered.

9. The Duchess is lying, unless it was Mister Edge who was murdered.
10. If Mister Ace was murdered, he was done in with a frying pan.
11. The cook did it, so the butler did not.
12. Of course the Duchess is lying!

Part C Using the symbolization key given, translate each English-language sentence into SL.

$\mathbf{E}_1$**:** Ava is an electrician.
$\mathbf{E}_2$**:** Harrison is an electrician.
$\mathbf{F}_1$**:** Ava is a firefighter.
$\mathbf{F}_2$**:** Harrison is a firefighter.
$\mathbf{S}_1$**:** Ava is satisfied with her career.
$\mathbf{S}_2$**:** Harrison is satisfied with his career.

1. Ava and Harrison are both electricians.
2. If Ava is a firefighter, then she is satisfied with her career.
3. Ava is a firefighter, unless she is an electrician.
4. Harrison is an unsatisfied electrician.
5. Neither Ava nor Harrison is an electrician.
6. Both Ava and Harrison are electricians, but neither of them find it satisfying.
7. Harrison is satisfied only if he is a firefighter.
8. If Ava is not an electrician, then neither is Harrison, but if she is, then he is too.
9. Ava is satisfied with her career if and only if Harrison is not satisfied with his.
10. If Harrison is both an electrician and a firefighter, then he must be satisfied with his work.
11. It cannot be that Harrison is both an electrician and a firefighter.
12. Harrison and Ava are both firefighters if and only if neither of them is an electrician.

Part D Give a symbolization key and symbolize the following sentences in SL.

1. Alice and Bob are both spies.
2. If either Alice or Bob is a spy, then the code has been broken.
3. If neither Alice nor Bob is a spy, then the code remains unbroken.
4. The German embassy will be in an uproar, unless someone has broken the code.
5. Either the code has been broken or it has not, but the German embassy will be in an uproar regardless.

6. Either Alice or Bob is a spy, but not both.

Part E Give a symbolization key and symbolize the following sentences in SL.

1. If Gregor plays first base, then the team will lose.
2. The team will lose unless there is a miracle.
3. The team will either lose or it won't, but Gregor will play first base regardless.
4. Gregor's mom will bake cookies if and only if Gregor plays first base.
5. If there is a miracle, then Gregor's mom will not bake cookies.

Part F For each argument, write a symbolization key and translate the argument as well as possible into SL.

1. If Dorothy plays the piano in the morning, then Roger wakes up cranky. Dorothy plays piano in the morning unless she is distracted. So if Roger does not wake up cranky, then Dorothy must be distracted.
2. It will either rain or snow on Tuesday. If it rains, Neville will be sad. If it snows, Neville will be cold. Therefore, Neville will either be sad or cold on Tuesday.
3. If Zoog remembered to do his chores, then things are clean but not neat. If he forgot, then things are neat but not clean. Therefore, things are either neat or clean—but not both.

Part G

1. Are there any sentences of SL that contain no sentence letters? Why or why not?
2. In the chapter, we symbolized an *exclusive or* using $\vee$, $\&$, and $\sim$. How could you translate an *exclusive or* using only two connectives? Is there any way to translate an *exclusive or* using only one connective?

Chapter 3

Truth Tables

This chapter introduces a way of evaluating sentences and arguments of SL called the truth table method. As we shall see, the truth table method is *semantic* because it involves one aspect of the meaning of sentences, whether those sentences are true or false. As we saw on page 36, semantics is the study of aspects of language related to meaning, including truth and falsity. Although it can be laborious, the truth table method is a purely mechanical procedure that requires no intuition or special insight. When we get to Chapter 6, we will provide a parallel semantic method for QL; however, this method will not be purely mechanical.

3.1 Basic Concepts

A formal, logical language is built from two kinds of elements: logical symbols and nonlogical symbols. LOGICAL SYMBOLS have their meaning fixed by the formal language. In SL, the logical symbols are the sentential connectives and the parentheses. When writing a symbolization key, you are not allowed to change the meaning of the logical symbols. You cannot say, for instance, that the ∼symbol will mean "not" in one argument and "perhaps" in another. The ∼symbol always means logical negation. It is used to translate the English language word "not", but it is a symbol of a formal language and is defined by its truth conditions.

The NONLOGICAL SYMBOLS are defined simply as all the symbols that aren't logical. The nonlogical symbols in SL are the sentence letters. When we translate an argument from English to SL, for example, the sentence letter M does not have its meaning fixed in advance; instead, we provide a symbolization key that says how M should be interpreted in that argument. In translating from English to a formal language, we provided symbolization keys which were interpretations of all the nonlogical symbols we used in the translation.

In logic, when we study artificial languages, we investigate their semantics by providing an interpretation of the nonlogical symbols. An INTERPRETATION is a way of setting up a correspondence between elements of the object language and elements of some other language or logical structure. The symbolization keys we used in Chapter 2 (p. 24) are a sort of interpretation.

The truth table method will also involve giving an interpretation of sentences, but they will be much simpler than the translation schemes we used in Chapter 2. We will not be concerned with what the individual sentence letters mean. We will only care whether they are true or false. We can do this, because of the way that the meaning of larger sentences is generated by the meaning of their parts.

Any nonatomic sentence of SL is composed of atomic sentences with sentential connectives. The truth value of the compound sentence depends only on the truth value of the atomic sentences that comprise it. In order to know the truth value of $(D \leftrightarrow E)$, for instance, you only need to know the truth value of D and the truth value of E. Connectives that work in this way are called truth functional. More technically, we define a TRUTH-FUNCTIONAL connective as an operator that builds larger sentences out of smaller ones, and fixes the truth value of the resulting sentence based only on the truth value of the component sentences.

Because all of the logical symbols in SL are truth functional, the only aspect of meaning we need to worry about in studying the semantics of SL is truth and falsity. If we want to know about the truth of the sentence $A \& B$, the only thing we need to know is whether A and B are true. It doesn't actually matter what else they mean. So if A is false, then $A \& B$ is false no matter what false sentence A is used to represent. It could be "I am the Pope" or "Pi is equal to 3.19." The larger sentence $A \& B$ is still false. So to give an interpretation of sentences in SL, all we need to do is create a truth assignment. A TRUTH ASSIGNMENT is a function that maps the sentence letters in SL onto our two truth values. In other words, we just need to assign Ts and Fs to all our sentence letters.

It is worth knowing that most languages are not built only out of truth functional connectives. In English, it is possible to form a new sentence from any simpler sentence $\mathcal{X}$ by saying "It is possible that $\mathcal{X}$." The truth value of this new sentence does not depend directly on the truth value of $\mathcal{X}$. Even if $\mathcal{X}$ is false, perhaps in some sense $\mathcal{X}$ *could* have been true—then the new sentence would be true. Some formal languages, called *modal logics*, have an operator for possibility. In a modal logic, we could translate "It is possible that $\mathcal{X}$" as $\diamond\mathcal{X}$.

$\mathcal{A}$	$\sim\mathcal{A}$
T	F
F	T

$\mathcal{A}$	$\mathcal{B}$	$\mathcal{A}\,\&\,\mathcal{B}$	$\mathcal{A}\vee\mathcal{B}$	$\mathcal{A}\rightarrow\mathcal{B}$	$\mathcal{A}\leftrightarrow\mathcal{B}$
T	T	T	T	T	T
T	F	F	T	F	F
F	T	F	T	T	F
F	F	F	F	T	T

Table 3.1: The characteristic truth tables for the connectives of SL.

However, the ability to translate sentences like these comes at a cost: The $\Diamond$ operator is not truth-functional, and so modal logics are not amenable to truth tables.

3.2 Complete Truth Tables

In the last chapter we introduced the characteristic truth tables for the different connectives. To put them all in one place, the truth tables for the connectives of SL are repeated in Table 3.1. Notice that when we did this, we listed all the possible combinations of truth and falsity for the sentence letters in these basic sentences. Each line of the truth table is thus a *truth assignment* for the sentence letters used in the sentence we are giving a truth table for. Thus one line of the truth table is all we need to give an *interpretation* of the sentence, and the full table gives all the possible interpretations of the sentence.

The truth value of sentences that contain only one connective is given by the characteristic truth table for that connective. The characteristic truth table for conjunction, for example, gives the truth conditions for any sentence of the form $(\mathcal{A}\,\&\,\mathcal{B})$. Even if the conjuncts $\mathcal{A}$ and $\mathcal{B}$ are long, complicated sentences, the conjunction is true if and only if both $\mathcal{A}$ and $\mathcal{B}$ are true. Consider the sentence $(H\,\&\,I) \rightarrow H$. We consider all the possible combinations of true and false 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	$(H$	$\&$	$I)$	$\rightarrow$	H
T	T	**T**		**T**		**T**
T	F	**T**		**F**		**T**
F	T	**F**		**T**		**F**
F	F	**F**		**F**		**F**

Now consider the subsentence $H\,\&\,I$. This is a conjunction $\mathcal{A}\,\&\,\mathcal{B}$ with H as $\mathcal{A}$ and with I as $\mathcal{B}$. H and I are both true on the first row. Since a conjunction is true when both conjuncts are true, we write a T underneath the conjunction symbol. We continue for the other three rows and get this:

H	I	$(H$	$\&$	$I)$	$\to$	H
		$\mathcal{A}$	$\&$	$\mathcal{B}$		
T	T	T	**T**	T		T
T	F	T	**F**	F		T
F	T	F	**F**	T		F
F	F	F	**F**	F		F

The entire sentence is a conditional $\mathcal{A}\to\mathcal{B}$ with $(H \& I)$ as $\mathcal{A}$ and with H as $\mathcal{B}$. On the second row, for example, $(H \& I)$ is false and H is true. Since a conditional is true when the antecedent is false, we write a T in the second row underneath the conditional symbol. We continue for the other three rows and get this:

H	I	$(H$	$\&$	$I)$	$\to$	H
			$\mathcal{A}$		$\to$	$\mathcal{B}$
T	T		T		**T**	T
T	F		F		**T**	T
F	T		F		**T**	F
F	F		F		**T**	F

The column of Ts underneath the conditional tells us that the sentence $(H \& I) \to H$ 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.

In this example, we have not repeated all of the entries in every successive table. 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	$(H$	$\&$	$I)$	$\to$	H
T	T	T	T	T	T	T
T	F	T	F	F	T	T
F	T	F	F	T	T	F
F	F	F	F	F	T	F

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 value of the sentence on each row is just the column underneath the *main connective* (see p. 39) of the sentence, in this case, the column underneath the conditional.

A COMPLETE TRUTH TABLE is a table that gives all the possible interpretations for a sentence or set of sentences in SL. It has a row for all the possible combinations of T and F for all of the sentence letters. 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 $[(C \leftrightarrow C) \rightarrow C] \,\&\, {\sim}(C \rightarrow C)$. The complete truth table requires only two lines because there are only two possibilities: C can be true, or it can be false. A single sentence letter can never be marked both T and F on the same row. The truth table for this sentence looks like this:

C	$[(C$	$\leftrightarrow$	$C)$	$\rightarrow$	$C]$	$\&$	$\sim$	$(C$	$\rightarrow$	$C)$
T	T	T	T	T	T	**F**	F	T	T	T
F	F	T	F	F	F	**F**	F	F	T	F

Looking at the column underneath the main connective, we see that the sentence is false on both rows of the table; i.e., it is false regardless of whether C is true or false.

A sentence that contains two sentence letters requires four lines for a complete truth table, as in the characteristic truth tables and the table for $(H \,\&\, I) \rightarrow I$.

A sentence that contains three sentence letters requires eight lines. For example:

M	N	P	M	$\&$	$(N$	$\vee$	$P)$
T	T	T	T	**T**	T	T	T
T	T	F	T	**T**	T	T	F
T	F	T	T	**T**	F	T	T
T	F	F	T	**F**	F	F	F
F	T	T	F	**F**	T	T	T
F	T	F	F	**F**	T	T	F
F	F	T	F	**F**	F	T	T
F	F	F	F	**F**	F	F	F

From this table, we know that the sentence $M \,\&\, (N \vee P)$ might be true or false, depending on the truth values of M, N, and P.

A complete truth table for a sentence that contains four different sentence letters requires 16 lines. For five letters, 32 lines are required. For six letters, 64 lines, and so on. To be perfectly general: If a complete truth table has n different sentence letters, then it must have 2^n rows.

In order to fill in the columns of a complete truth table, begin with the rightmost sentence letter and alternate Ts and Fs. In the next column to the left, write two Ts, write two Fs, and repeat. For the third sentence letter, write four Ts followed by four Fs. This yields an eight line truth table like the one above. For a 16 line truth table, the next column of sentence letters should have eight Ts followed by eight Fs. For a 32 line table, the next column would have 16 Ts followed by 16 Fs. And so on.

Practice Exercises

Part A Identify the main connective in the each sentence.

1. $\sim(A \vee \sim B)$
2. $\sim(A \vee \sim B) \vee \sim(A \,\&\, D)$
3. $[\sim(A \vee \sim B) \vee \sim(A \,\&\, D)] \rightarrow E$
4. $[(A \rightarrow B) \,\&\, C] \leftrightarrow [A \vee (B \,\&\, C)]$
5. $\sim\sim\sim[A \vee (B \,\&\, (C \vee D))]$

Part B Identify the main connective in the each sentence.

1. $[(A \leftrightarrow B) \,\&\, C] \rightarrow D$
2. $[(D \,\&\, (E \,\&\, F)) \vee G] \leftrightarrow \sim[A \rightarrow (C \vee G)]$
3. $\sim(\sim Z \vee \sim H)$
4. $(\sim(P \,\&\, S) \leftrightarrow G) \,\&\, Y$
5. $(A \,\&\, (B \rightarrow C)) \vee \sim D$

Part C Assume A, B, and C are true and X, Y, and Z are false and evaluate the truth of the each sentence.

1. ∼((A &B) →X)
2. (Y ∨Z) ↔(∼X ↔B)
3. $[(X \rightarrow A) \vee (A \rightarrow X)] \,\&\, Y$
4. (X →A) ∨(A →X)
5. $[A \,\&\, (Y \,\&\, Z)] \vee A$

Part D Assume A, B, and C are true and X, Y, and Z are false and evaluate the truth of the each sentence.

1. $\sim\sim(\sim\sim\sim A \vee X)$
2. $(A \rightarrow B) \rightarrow X$
3. $((A \vee B) \,\&\, (C \leftrightarrow X)) \vee Y$
4. $(A \rightarrow B) \vee (X \,\&\, (Y \,\&\, Z))$

5. $((A \vee X) \to Y) \,\&\, B$

Part E Write complete truth tables for the following sentences and mark the column that represents the possible truth values for the whole sentence.

1. $\sim(S \leftrightarrow (P \to S))$
2. $\sim[(X \,\&\, Y) \vee (X \vee Y)]$
3. $(A \to B) \leftrightarrow (\sim B \leftrightarrow \sim A)$
4. $[C \leftrightarrow (D \vee E)] \,\&\, \sim C$
5. $\sim(G \,\&\, (B \,\&\, H)) \leftrightarrow (G \vee (B \vee H))$

Part F Write complete truth tables for the following sentences and mark the column that represents the possible truth values for the whole sentence.

1. $(D \,\&\, \sim D) \to G$
2. $(\sim P \vee \sim M) \leftrightarrow M$
3. $\sim\sim(\sim A \,\&\, \sim B)$
4. $[(D \,\&\, R) \to I] \to \sim(D \vee R)$
5. $\sim[(D \leftrightarrow O) \leftrightarrow A] \to (\sim D \,\&\, O)$

3.3 Using Truth Tables

A complete truth table shows us every possible combination of truth assignments on the sentence letters. It tells us every possible way sentences can relate to truth. We can use this to discover all sorts of logical properties of sentences and sets of sentences.

Tautologies, contradictions, and contingent sentences

We defined a tautology as a statement that must be true as a matter of logic, no matter how the world is (p. 16). A statement like “Either it is raining or it is not raining” is always true, no matter what the weather is like outside. Something similar goes on in truth tables. With a complete truth table, we consider all of the ways that the world might be. Each line of the truth table corresponds to a way the world might be. This means that if the sentence is true on every line of a complete truth table, then it is true as a matter of logic, regardless of what the world is like.

We can use this fact to create a test for whether a sentence is a tautology: if the column under the main connective of a sentence is a T on every row, the sentence is a tautology. Not every tautology in English will correspond to a tautology in SL. The sentence “All bachelors are unmarried” is a tautology in English, but we cannot represent it as a tautology in SL, because it just translates as a single sentence letter, like B. On the other hand, if something is a tautology in SL, it will also be a tautology in English. No matter how you translate $A \vee {\sim}A$, if you translate the As consistently, the statement will be a tautology.

Rather than thinking of complete truth tables as an imperfect test for the English notion of a tautology, we can define a separate notion of a tautology in SL. A statement is a SEMANTIC TAUTOLOGY IN SL if and only if the column under the main connective in the complete truth table for the sentence contains only Ts. We are specifying that this is the semantic notion of tautology because in the next chapter we will define a separate, syntactic, notion of tautology.

Conversely, we defined a contradiction as a sentence that is false no matter how the world is (p. 16). This means we can define a SEMANTIC CONTRADICTION IN SL as a sentence that has only Ts in the column under them main connective of its complete truth table.

Finally, a sentence is contingent if it is sometimes true and sometimes false (p. 16). Similarly, a sentence is SEMANTICALLY CONTINGENT IN SL if and only if its complete truth table for has both Ts and Fs under the main connective.

From the truth tables in the previous section, we know that $(H \,\&\, I) \rightarrow H$ is a tautology (p. 48), that $[(C \leftrightarrow C) \rightarrow C] \,\&\, {\sim}(C \rightarrow C)$ is a contradiction (p. 49), and that $M \,\&\, (N \vee P)$ is contingent (p. 49).

Logical equivalence

Two sentences are logically equivalent in English if they have the same truth value as a of matter logic (p. 17). Once again, we can use truth tables to define a similar property in SL: Two sentences are SEMANTICALLY LOGICALLY EQUIVALENT IN SL if they have the same truth value on every row of a complete truth table.

Consider the sentences ${\sim}(A \vee B)$ and ${\sim}A \,\&\, {\sim}B$. Are they logically equivalent? To find out, we construct a truth table.

A	B	$\sim$	$(A$	$\vee$	$B)$	$\sim$	A	$\&$	$\sim$	B
T	T	**F**	T	T	T	F	T	**F**	F	T
T	F	**F**	T	T	F	F	T	**F**	T	F
F	T	**F**	F	T	T	T	F	**F**	F	T
F	F	**T**	F	F	F	T	F	**T**	T	F

Look at the columns for the main connectives; negation for the first sentence, conjunction for the second. On the first three rows, both are F. On the final row, both are T. Since they match on every row, the two sentences are logically equivalent.

Consistency

A set of sentences in English is consistent if it is logically possible for them all to be true at once (p. 17). This means that a sentence is SEMANTICALLY CONSISTENT IN SL if and only if there is at least one line of a complete truth table on which all of the sentences are true. It is inconsistent otherwise.

Validity

Logic is the study of argument, so the most important thing we can use truth tables of is to test the validity of arguments. An argument in English is valid if it is logically impossible for the premises to be true and for the conclusion to be false at the same time (p. 11). So we can define SEMANTIC VALIDITY IN SL as the situation where there is no row of a complete truth table on which the premises are all marked "T" and the conclusion is marked "F." An argument is invalid if there is such a row.

Consider this argument:

$\sim L \rightarrow (J \vee L)$
$\sim L$
$\therefore\ J$

Is it valid? To find out, we construct a truth table.

J	L	$\sim L \rightarrow (J \vee L)$	∼ L	J
T	T	F T **T** T T T	**F** T	**T**
T	F	T F **T** T T F	**T** F	**T**
F	T	F T **T** F T T	**F** T	**F**
F	F	T F **F** F F F	**T** F	**F**

Yes, the argument is valid. The only row on which both the premises are T is the second row, and on that row the conclusion is also T.

In Chapters 1 and 2 we used the three dots ∴ to represent an inference in English. We used this symbol to represent any kind of inference. The truth table method gives us a more specific notion of a valid inference. We will call this semantic entailment and represent it using a new symbol, $\models$, called the "double turnstile." The $\models$ is like the $\therefore$, except for arguments verified by truth tables. When you use the double turnstile, you write the premises as a set, using curly brackets, { and }, which mathematicians use in set theory. The argument above would be written $\{\sim L \rightarrow (J \vee L), \sim L\} \models J$.

More formally, we can define the double turnstile this way: $\{\mathcal{A}_1 \ldots \mathcal{A}_n\} \models \mathcal{B}$ if and only if there is no truth value assignment for which $\mathcal{A}$ is true and $\mathcal{B}$ is false. Put differently, it means that $\mathcal{B}$ is true for any and all truth value assignments for which $\mathcal{A}$ is true.

We can also use the double turnstile to represent other logical notions. Since a tautology is always true, it is like the conclusion of a valid argument with no premises. The string $\models \mathcal{C}$ means that $\mathcal{C}$ is true for all truth value assignments. This is equivalent to saying that the sentence is entailed by anything. We can represent logical equivalence by writing the double turnstile in both directions: $\mathcal{A} \dashv\,\models \mathcal{B}$

Practice Exercises

If you want additional practice, you can construct truth tables for any of the sentences and arguments in the exercises for the previous chapter.

Part A

Determine whether each sentence is a tautology, a contradiction, or a contingent sentence, using a complete truth table.

1. $A \rightarrow A$
2. $C \rightarrow \sim C$
3. $(A \leftrightarrow B) \leftrightarrow \sim(A \leftrightarrow \sim B)$
4. $(A \rightarrow B) \vee (B \rightarrow A)$
5. $(A \,\&\, B) \rightarrow (B \vee A)$
6. $[(\sim A \vee A) \vee B] \rightarrow B$
7. $[(A \vee B) \,\&\, \sim A] \,\&\, (B \rightarrow A)$

Part B

Determine whether each sentence is a tautology, a contradiction, or a contingent sentence, using a complete truth table.

1. $\sim B \,\&\, B$
2. $\sim D \vee D$

3. $(A \& B) \vee (B \& A)$
4. $\sim[A \rightarrow (B \rightarrow A)]$
5. $A \leftrightarrow [A \rightarrow (B \& \sim B)]$
6. $[(A \& B) \leftrightarrow B] \rightarrow (A \rightarrow B)$

Part C Determine whether each pair of sentences is logically equivalent using complete truth tables.

1. A, $\sim A$
2. $A \& \sim A$, $\sim B \leftrightarrow B$
3. $[(A \vee B) \vee C]$, $[A \vee (B \vee C)]$
4. $A \vee (B \& C)$, $(A \vee B) \& (A \vee C)$
5. $[A \& (A \vee B)] \rightarrow B$, $A \rightarrow B$

Part D Determine whether each pair of sentences is logically equivalent using complete truth tables.

1. $A \rightarrow A$, $A \leftrightarrow A$
2. $\sim(A \rightarrow B)$, $\sim A \rightarrow \sim B$
3. $A \vee B$, $\sim A \rightarrow B$
4. $(A \rightarrow B) \rightarrow C$, $A \rightarrow (B \rightarrow C)$
5. $A \leftrightarrow (B \leftrightarrow C)$, $A \& (B \& C)$

Part E Determine whether each set of sentences is consistent or inconsistent using a complete truth table.

1. $A \& \sim B$, $\sim(A \rightarrow B)$, $B \rightarrow A$
2. $A \vee B$, $A \rightarrow \sim A$, $B \rightarrow \sim B$
3. $\sim(\sim A \vee B)$, $A \rightarrow \sim C$, $A \rightarrow (B \rightarrow C)$
4. $A \rightarrow B$, $A \& \sim B$
5. $A \rightarrow (B \rightarrow C)$, $(A \rightarrow B) \rightarrow C$, $A \rightarrow C$

Part F Determine whether each set of sentences is consistent or inconsistent, using a complete truth table.

1. $\sim B$, $A \rightarrow B$, A
2. $\sim(A \vee B)$, $A \leftrightarrow B$, $B \rightarrow A$
3. $A \vee B$, $\sim B$, $\sim B \rightarrow \sim A$
4. $A \leftrightarrow B$, $\sim B \vee \sim A$, $A \rightarrow B$
5. $(A \vee B) \vee C$, $\sim A \vee \sim B$, $\sim C \vee \sim B$

Part G Determine whether each argument is valid or invalid, using a complete truth table.

1. $A \rightarrow A$, $\therefore A$
2. $A \rightarrow B$, B, $\therefore A$
3. $A \leftrightarrow B$, $B \leftrightarrow C$, $\therefore A \leftrightarrow C$
4. $A \rightarrow B$, $A \rightarrow C$ $\therefore B \rightarrow C$
5. $A \rightarrow B$, $B \rightarrow A$ $\therefore A \leftrightarrow B$

Part H Determine whether each argument is valid or invalid, using a complete truth table.

1. $A \vee \big[A \rightarrow (A \leftrightarrow A)\big]$, $\therefore$ A
2. $A \vee B$, $B \vee C$, $\sim B$, $\therefore A \,\&\, C$
3. $A \rightarrow B$, $\sim A$ $\therefore \sim B$
4. A, B $\therefore \sim(A \rightarrow \sim B)$
5. $\sim(A \,\&\, B)$, $A \vee B$, $A \leftrightarrow B$ $\therefore C$

3.4 Partial Truth Tables

In order to show that a sentence is a tautology, we need to show that it is T on every row. So we need a complete truth table. To show that a sentence is *not* a tautology, however, we only need one line: a line on which the sentence is F. Therefore, in order to show that something is not a tautology, it is enough to provide a one-line *partial truth table*—regardless of how many sentence letters the sentence might have in it.

Consider, for example, the sentence $(U \,\&\, T) \rightarrow (S \,\&\, W)$. We want to show that it is *not* a tautology by providing a partial truth table. We fill in F for the entire sentence. The main connective of the sentence is a conditional. In order for the conditional to be false, the antecedent must be true (T) and the consequent must be false (F). So we fill these in on the table:

S	T	U	W	$(U$	$\&$	$T)$	$\rightarrow$	$(S$	$\&$	$W)$
					T		**F**		F	

In order for the $(U \,\&\, T)$ to be true, both U and T must be true.

S	T	U	W	$(U$	$\&$	$T)$	$\rightarrow$	$(S$	$\&$	$W)$
	T	T		T	T	T	**F**		F	

Now we just need to make $(S \,\&\, W)$ false. To do this, we need to make at least one of S and W false. We can make both S and W false if we want. All that matters is that the whole sentence turns out false on this line. Making an arbitrary decision, we finish the table in this way:

S	T	U	W	$(U \,\&\, T) \rightarrow (S \,\&\, W)$
F	T	T	F	T T T **F** F F F

Showing that something is a contradiction requires a complete truth table. Showing that something is *not* a contradiction requires only a one-line partial truth table, where the sentence is true on that one line.

A sentence is contingent if it is neither a tautology nor a contradiction. So showing that a sentence is contingent requires a *two-line* partial truth table: The sentence must be true on one line and false on the other. For example, we can show that the sentence above is contingent with this truth table:

S	T	U	W	$(U \,\&\, T) \rightarrow (S \,\&\, W)$
F	T	T	F	T T T **F** F F F
F	T	F	F	F F T **T** F F F

Note that there are many combinations of truth values that would have made the sentence true, so there are many ways we could have written the second line.

Showing that a sentence is *not* contingent requires providing a complete truth table, because it requires showing that the sentence is a tautology or that it is a contradiction. If you do not know whether a particular sentence is contingent, then you do not know whether you will need a complete or partial truth table. You can always start working on a complete truth table. If you complete rows that show the sentence is contingent, then you can stop. If not, then complete the truth table. Even though two carefully selected rows will show that a contingent sentence is contingent, there is nothing wrong with filling in more rows.

Showing that two sentences are logically equivalent requires providing a complete truth table. Showing that two sentences are *not* logically equivalent requires only a one-line partial truth table: Make the table so that one sentence is true and the other false.

Showing that a set of sentences is consistent requires providing one row of a truth table on which all of the sentences are true. The rest of the table is irrelevant, so a one-line partial truth table will do. Showing that a set of sentences is inconsistent, on the other hand, requires a complete truth table: You must show that on every row of the table at least one of the sentences is false.

Showing that an argument is valid requires a complete truth table. Showing that an argument is *invalid* only requires providing a one-line truth table: If you can produce a line on which the premises are all true and the conclusion is false, then the argument is invalid.

	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
equivalent?	complete truth table	one-line partial truth table
consistent?	one-line partial truth table	complete truth table
valid?	complete truth table	one-line partial truth table

Table 3.2: Do you need a complete truth table or a partial truth table? It depends on what you are trying to show.

Table 3.2 summarizes when a complete truth table is required and when a partial truth table will do.

Key Terms

- LOGICAL SYMBOLS are symbols that have their meaning fixed by the formal language (p. 45).
- The NONLOGICAL SYMBOLS are all the symbols that aren't logical (p. 45).
- An INTERPRETATION is a correspondence between elements of the object language and elements of some other language or logical structure (p. 46).
- A TRUTH-FUNCTIONAL connective is an operator that builds larger sentences out of smaller ones and fixes the truth value of the resulting sentence based only on the truth value of the component sentences (p. 46)
- A TRUTH ASSIGNMENT is a function that maps the sentence letters in SL onto truth values (p. 46).
- A COMPLETE TRUTH TABLE is a table that gives all the possible interpretations for a sentence or set of sentences in SL (p. 49).
- A statement is a SEMANTIC TAUTOLOGY IN SL if and only if the column under the main connective in the complete truth table for the sentence contains only Ts (p. 52).
- A statement is a SEMANTIC CONTRADICTION IN SL if and only if it has only Ts in the column under the main connective of its complete truth table (p. 52).
- A sentence is SEMANTICALLY CONTINGENT IN SL if and only if its complete truth table for has both Ts and Fs under the main connective (p. 52).
- Two sentences are SEMANTICALLY LOGICALLY EQUIVALENT IN SL if they have the same truth value on every row of a complete truth table (p. 52).

- A set of sentences is SEMANTICALLY CONSISTENT IN SL if and only if there is at least one line of a complete truth table on which all of the sentences are true. It is inconsistent otherwise (p. 53).

- An argument is SEMANTICALLY VALID IN SL if there is no row of a complete truth table on which the premises are all true and the conclusion is False (p. 53).

Practice Exercises

If you want additional practice, you can construct truth tables for any of the sentences and arguments in the exercises for the previous chapter.

Part A Determine whether each sentence is a tautology, a contradiction, or a contingent sentence. Justify your answer with a complete or partial truth table where appropriate.

1. $\sim(A \vee B) \leftrightarrow (\sim A \,\&\, \sim B)$
2. $\sim(A \,\&\, B) \leftrightarrow A$
3. $\big[(A \,\&\, B) \,\&\, \sim(A \,\&\, B)\big] \,\&\, C$
4. $A \rightarrow (B \vee C)$
5. $[(A \,\&\, B) \,\&\, C] \rightarrow B$
6. $(A \,\&\, \sim A) \rightarrow (B \vee C)$
7. $\sim\big[(C \vee A) \vee B\big]$
8. $(B \,\&\, D) \leftrightarrow [A \leftrightarrow (A \vee C)]$
9. $(A \,\&\, B)] \rightarrow [(A \,\&\, C) \vee (B \,\&\, D)]$
10. $\sim[(A \rightarrow B) \vee (C \rightarrow D)]$

Part B Determine whether each pair of sentences is logically equivalent. Justify your answer with a complete or partial truth table where appropriate.

1. A, $\sim A$
2. A, $A \vee A$
3. $A \rightarrow A$, $A \leftrightarrow A$
4. $A \vee \sim B$, $A \rightarrow B$
5. $A \,\&\, \sim A$, $\sim B \leftrightarrow B$
6. $\sim(A \,\&\, B)$, $\sim A \vee \sim B$
7. $\sim(A \rightarrow B)$, $\sim A \rightarrow \sim B$
8. $(A \rightarrow B)$, $(\sim B \rightarrow \sim A)$
9. $[(A \vee B) \vee C]$, $[A \vee (B \vee C)]$
10. $[(A \vee B) \,\&\, C]$, $[A \vee (B \,\&\, C)]$

Part C Determine whether each set of sentences is consistent or inconsistent. Justify your answer with a complete or partial truth table where appropriate.

1. $A \rightarrow A$, $\sim A \rightarrow \sim A$, $A \,\&\, A$, $A \vee A$
2. $A \,\&\, B$, $C \rightarrow \sim B$, C
3. $A \vee B$, $A \rightarrow C$, $B \rightarrow C$
4. $A \rightarrow B$, $B \rightarrow C$, A, $\sim C$
5. $B \,\&\, (C \vee A)$, $A \rightarrow B$, $\sim(B \vee C)$
6. $A \vee B$, $B \vee C$, $C \rightarrow \sim A$
7. $A \leftrightarrow (B \vee C)$, $C \rightarrow \sim A$, $A \rightarrow \sim B$
8. A, B, C, $\sim D$, $\sim E$, F
9. $A \leftrightarrow B$, $A \rightarrow C$, $B \rightarrow D$, $\sim(C \vee D)$
10. $A \,\&\, (B \vee C)$, $\sim(A \,\&\, C)$, $\sim(B \,\&\, C)$

Part D Determine whether each argument is valid or invalid. Justify your answer with a complete or partial truth table where appropriate.

1. $A \rightarrow (A \,\&\, \sim A)$, $\therefore \sim A$
2. $A \leftrightarrow \sim(B \leftrightarrow A)$, $\therefore A$
3. $A \vee (B \rightarrow A)$, $\therefore \sim A \rightarrow \sim B$
4. $A \vee B$, $B \vee C$, $\sim A$, $\therefore B \,\&\, C$
5. $(B \,\&\, A) \rightarrow C$, $(C \,\&\, A) \rightarrow B$, $\therefore (C \,\&\, B) \rightarrow A$
6. $A \rightarrow C$, $E \rightarrow (D \vee B)$, $B \rightarrow \sim D$, $\therefore (A \vee C) \vee (B \rightarrow (E \,\&\, D))$
7. $A \,\&\, (B \rightarrow C)$, $\sim C \,\&\, (\sim B \rightarrow \sim A)$ $\therefore C \,\&\, \sim C$
8. $A \vee B$, $C \rightarrow A$, $C \rightarrow B$ $\therefore A \rightarrow (B \rightarrow C)$
9. $A \rightarrow B$ $\therefore (A \,\&\, B) \vee (\sim A \,\&\, \sim B)$
10. $A \rightarrow B$, $\sim B \vee A$ $\therefore A \leftrightarrow B$

Part E Answer each of the questions below and justify your answer.

1. Suppose that $\mathcal{A}$ and $\mathcal{B}$ are logically equivalent. What can you say about $\mathcal{A} \leftrightarrow \mathcal{B}$?
2. Suppose that $(\mathcal{A} \,\&\, \mathcal{B}) \rightarrow \mathcal{C}$ is contingent. What can you say about the argument "$\mathcal{A}$, $\mathcal{B}$, $\therefore \mathcal{C}$"?
3. Suppose that $\{\mathcal{A}, \mathcal{B}, \mathcal{C}\}$ is inconsistent. What can you say about $(\mathcal{A} \,\&\, \mathcal{B} \,\&\, \mathcal{C})$?
4. Suppose that $\mathcal{A}$ is a contradiction. What can you say about the argument "$\mathcal{A}$, $\mathcal{B}$, $\therefore \mathcal{C}$"?
5. Suppose that $\mathcal{C}$ is a tautology. What can you say about the argument "$\mathcal{A}$, $\mathcal{B}$, $\therefore \mathcal{C}$"?
6. Suppose that $\mathcal{A}$ and $\mathcal{B}$ are *not* logically equivalent. What can you say about $(\mathcal{A} \vee \mathcal{B})$?

Part F We could leave the biconditional ($\leftrightarrow$) out of the language. If we did that, we could still write "$A \leftrightarrow B$" so as to make sentences easier to read, but that would be shorthand for $(A \rightarrow B) \,\&\, (B \rightarrow A)$. The resulting language would be formally equivalent to SL, since $A \leftrightarrow B$ and $(A \rightarrow B) \,\&\, (B \rightarrow A)$ are logically equivalent in SL. If we valued formal simplicity over expressive richness, we could replace more of the connectives with notational conventions and still have a language equivalent to SL.

There are a number of equivalent languages with only two connectives. It would be enough to have only negation and the material conditional. Show this by writing sentences that are logically equivalent to each of the following using only parentheses, sentence letters, negation ($\sim$), and the material conditional ($\rightarrow$).

⋆ 1. $A \vee B$
⋆ 2. $A \,\&\, B$
⋆ 3. $A \leftrightarrow B$

We could have a language that is equivalent to SL with only negation and disjunction as connectives. Show this: Using only parentheses, sentence letters, negation ($\sim$), and disjunction ($\vee$), write sentences that are logically equivalent to each of the following.

4. $A \,\&\, B$
5. $A \rightarrow B$
6. $A \leftrightarrow B$

The *Sheffer stroke* is a logical connective with the following characteristic truth table:

$\mathcal{A}$	$\mathcal{B}$	$\mathcal{A}\vert\mathcal{B}$
T	T	F
T	F	T
F	T	T
F	F	T

7. Write a sentence using the connectives of SL that is logically equivalent to $(A\vert B)$.

Every sentence written using a connective of SL can be rewritten as a logically equivalent sentence using one or more Sheffer strokes. Using only the Sheffer stroke, write sentences that are equivalent to each of the following.

8. $\sim A$

9. $(A \,\&\, B)$
10. $(A \vee B)$
11. $(A \rightarrow B)$
12. $(A \leftrightarrow B)$

Chapter 4

Proofs in Sentential Logic

4.1 Substitution Instances and Proofs

Consider two arguments in SL:

Argument A

$$\begin{array}{ll} & P \vee Q \\ & {\sim}P \\ \therefore & \mathrm{Q} \end{array}$$

Argument B

$$\begin{array}{ll} & P \rightarrow Q \\ & P \\ \therefore & \mathrm{Q} \end{array}$$

These are both valid arguments. Go ahead and prove that for yourself by constructing the four-line truth tables. These particular valid arguments are examples of important kinds of arguments that are given special names. Argument A is an example of a kind of argument traditionally called *disjunctive syllogism*. In the system of proof we will develop later in the chapter, it will be given a newer name, *disjunction elimination* ($\vee$-E). Given a disjunction and the negation of one of the disjuncts, the other disjunct follows as a valid consequence. Argument B makes use of a different valid form: Given a conditional and its antecedent, the consequent follows as a valid consequence. This is traditionally called *modus ponens*. In our system it will be called *conditional elimination* ($\rightarrow$-E). Both of these arguments remain valid even if we substitute different sentence letters. You don't even need to run the truth tables again to see that these arguments are valid:

Argument A*

$$\begin{array}{rl} & A \vee B \\ & {\sim}A \\ \therefore & \mathrm{B} \end{array}$$

Argument B*

$$\begin{array}{rl} & A \rightarrow B \\ & A \\ \therefore & \mathrm{B} \end{array}$$

Replacing P with A and Q with B changes nothing (so long as we are sure to replace *every* P with an A and every Q with a B). What's more interesting is that we can replace the individual sentence letters in Argument A and Argument B with longer sentences in SL and the arguments will still be valid, as long as we do the substitutions consistently. Here are two more perfectly valid instances of disjunction and conditional elimination.

Argument A**

$$\begin{array}{rl} & (C \,\&\, D) \vee (E \vee F) \\ & {\sim}(C \,\&\, D) \\ \therefore & E \,\&\, F \end{array}$$

Argument B**

$$\begin{array}{rl} & (G \rightarrow H) \rightarrow (I \vee J) \\ & (G \rightarrow H) \\ \therefore & I \vee J \end{array}$$

Again, you can check these using truth tables, although the 16 line truth tables begin to get tiresome. All of these arguments are what we call *substitution instances* of the same two logical forms. We call them that because you get them by replacing the sentence letters with other sentences, either sentence letters or longer sentences in SL. A substitution instance cannot change the sentential connectives of a sentence, however. The sentential connectives are what makes the *logical form* of the sentence. We can write these logical forms using fancy script letters.

Disjunction Elimination
(Disjunctive Syllogism)

$$\begin{array}{rl} & \mathcal{A} \vee \mathcal{B} \\ & {\sim}\mathcal{A} \\ \therefore & \mathcal{B} \end{array}$$

Conditional Elimination
(Modus Ponens)

$$\begin{array}{rl} & \mathcal{A} \rightarrow \mathcal{B} \\ & \mathcal{A} \\ \therefore & \mathcal{B} \end{array}$$

As we explained in Chapter 2, the fancy script letters are *metavariables.* They are a part of our metalanguage and can refer to single sentence letters like P or longer sentences like $A \leftrightarrow (B \,\&\, (C \vee D))$.

Formally, we can define a SENTENCE FORM as a sentence in SL that contains one or more metavariables in place of sentence letters. A SUBSTITUTION INSTANCE of that sentence form is then a sentence created by consistently substituting sentences for one or more of the metavariables in the sentence form. Here "consistently substituting" means replacing all instances of the metavariable with the same sentence. You cannot replace instances of the same metavariable with different sentence, or leave a metavariable as it is, if you have replaced other metavariables of the same type. An ARGUMENT FORM is an argument that includes one or more sentence forms, and a SUBSTITUTION INSTANCE of the argument form is the argument obtained by consistently replacing the sentence forms in the argument form with their substitution instances.

Once we start identifying valid argument forms like this, we have a new way of showing that longer arguments are valid. Truth tables are fun, but doing the 1028 line truth table for an argument with 10 sentence letters would be tedious. Worse, we would never be sure we hadn't made a little mistake in all those Ts and Fs. Part of the problem is that we have no way of knowing *why* the argument is valid. The table gives you very little insight into how the premises work together.

The aim of a *proof system* is to show that particular arguments are valid in a way that allows us to understand the reasoning involved in the argument. Instead of representing all the premises and the conclusion in one table, we break the argument up into steps. Each step is a basic argument form of the sort we saw above, like disjunctive syllogism or modus ponens. Suppose we are given the premises $\sim L \rightarrow (J \vee L)$ and $\sim L$ and wanted to show J. We can break this up into two smaller arguments, each of which is a substitution inference of a form we know is correct.

Argument 1	Argument 2
$\sim L \rightarrow (J \vee L)$	$J \vee L$
$\sim L$	$\sim L$
$\therefore\ J \vee L$	$\therefore\ J$

The first argument is a substitution instance of modus ponens and the second is a substitution instance of disjunctive syllogism, so we know they are both valid. Notice also that the conclusion of the first argument is the first premise of the second, and the second premise is the same in both arguments. Together, these arguments are enough to get us from $\sim L \rightarrow (J \vee L)$ and $\sim L$ to J.

These two arguments take up a lot of space, though. To complete our proof system, we need a system for showing clearly how simple steps can combine to get us from premises to conclusions. The system we will use in this book was devised by the American logician Frederic Brenton Fitch (1908–1987). We begin by writing our premises on numbered lines with a bar on the left and a little bar underneath to represent the end of the premises. Then we write "Want" on the side followed by the conclusion we are trying to reach. If we wanted to write out arguments 1 and 2 above, we would begin like this.

$$\begin{array}{r|ll} 1 & {\sim}L \to (J \vee L) & \\ 2 & {\sim}L & \text{Want: } J \\ \hline \end{array}$$

We then add the steps leading to the conclusion below the horizontal line, each time explaining off to the right why we are allowed to write the new line. This explanation consists of citing a rule and the prior lines the rule is applied to. In the example we have been working with we would begin like this

$$\begin{array}{r|ll} 1 & {\sim}L \to (J \vee L) & \\ 2 & {\sim}L & \text{Want: } J \\ \hline 3 & J \vee L & {\to}\text{E } 1, 2 \end{array}$$

and then go like this

$$\begin{array}{r|ll} 1 & {\sim}L \to (J \vee L) & \\ 2 & {\sim}L & \text{Want: } J \\ \hline 3 & J \vee L & {\to}\text{E } 1, 2 \\ 4 & J & {\vee}\text{E } 2, 3 \end{array}$$

The little chart above is a *proof* that J follows from ${\sim}L \to (J \vee L)$ and ${\sim}L$. We will also call proofs like this *derivations*. Formally, a PROOF is a sequence of sentences. The first sentences of the sequence are assumptions; these are the premises of the argument. Every sentence later in the sequence follows from earlier sentences by one of the rules of proof. The final sentence of the sequence is the conclusion of the argument.

In the remainder of this chapter, we will develop a system for proving sentences in SL. Later, in Chapter 7, this will be expanded to cover Quantified Logic (QL). First, though, you should practice identifying substitution instances of sentences and longer rules.

Practice Exercises

Part A For each problem, a sentence form is given in sentence variables. Identify which of the sentences after it are legitimate substitution instances of that form.

(1) $\mathcal{A} \,\&\, \mathcal{B}$:

a. $P \vee Q$

b. $(A \rightarrow B) \,\&\, C$

c. $[(A \,\&\, B) \rightarrow (B \,\&\, A)] \,\&\, ({\sim}A \,\&\, {\sim}B)$

d. $[((A \,\&\, B) \,\&\, C) \,\&\, D] \,\&\, F$

e. $(A \,\&\, B) \rightarrow C$

(2) ${\sim}(\mathcal{P} \,\&\, \mathcal{Q})$

a. ${\sim}(A \,\&\, B)$

b. ${\sim}(A \,\&\, A)$

c. ${\sim}A \,\&\, B$

d. ${\sim}(({\sim}A \,\&\, B) \,\&\, (B \,\&\, {\sim}A))$

e. ${\sim}(A \rightarrow B)$

(3) ${\sim}\mathcal{A}$

a. ${\sim}A \rightarrow B$

b. ${\sim}(A \rightarrow B)$

c. ${\sim}[(G \rightarrow (H \vee I)) \rightarrow G]$

d. ${\sim}G \,\&\, ({\sim}B \,\&\, {\sim}H)$

e. ${\sim}(G \,\&\, (B \,\&\, H))$

(4) ${\sim}\mathcal{A} \rightarrow \mathcal{B}$

a. ${\sim}A \,\&\, B$

b. ${\sim}B \rightarrow A$

c. ${\sim}(X \,\&\, Y) \rightarrow (Z \vee B)$

d. ${\sim}(A \rightarrow B)$

e. $A \rightarrow {\sim}B$

(5) ${\sim}\mathcal{A} \leftrightarrow {\sim}\mathcal{Z}$

a. ${\sim}(P \leftrightarrow Q)$

b. ${\sim}(P \leftrightarrow Q) \leftrightarrow {\sim}(Q \leftrightarrow P)$

c. ${\sim}H \rightarrow {\sim}G$

d. ${\sim}(A \,\&\, B) \leftrightarrow C$

e. ${\sim}[{\sim}(P \leftrightarrow Q) \leftrightarrow R] \leftrightarrow {\sim}S$

(6) $(\mathcal{A} \,\&\, \mathcal{B}) \vee \mathcal{C}$

a. $(P \vee Q) \,\&\, R$

b. $({\sim}M \,\&\, {\sim}D) \vee C$

c. $(D \,\&\, R) \,\&\, (I \vee D)$

d. $[(D \rightarrow O) \vee A] \,\&\, D$

e. $[(A \,\&\, B) \,\&\, C] \vee (D \vee A)$

(7) $(\mathcal{A} \,\&\, \mathcal{B}) \vee \mathcal{A}$

a. $((C \rightarrow D) \,\&\, E) \vee A$

b. $(A \,\&\, A) \vee A$

c. $((C \rightarrow D) \,\&\, E) \vee (C \rightarrow D)$

d. $((G \,\&\, B) \,\&\, (Q \vee R)) \vee (G \,\&\, B)$

e. $(P \vee Q) \,\&\, P$

(8) $\mathcal{P} \rightarrow (\mathcal{P} \rightarrow \mathcal{Q})$

a. $A \rightarrow (B \rightarrow C)$

b. $(A \,\&\, B) \rightarrow [(A \,\&\, B) \rightarrow C]$

c. $(G \rightarrow B) \rightarrow [(G \rightarrow B) \rightarrow (G \rightarrow B)]$

d. $M \rightarrow [M \rightarrow (D \,\&\, (C \,\&\, M))]$

e. $(S \vee O) \rightarrow [(O \vee S) \rightarrow A]$

(9) ${\sim}\mathcal{A} \vee (\mathcal{B} \,\&\, {\sim}\mathcal{B})$

a. ${\sim}P \vee (Q \,\&\, {\sim}P)$

b. ${\sim}A \vee (A \,\&\, {\sim}A)$

c. $(P \rightarrow Q) \vee [(P \rightarrow Q) \,\&\, {\sim}R]$

d. ${\sim}E \,\&\, (F \,\&\, {\sim}F)$

e. ${\sim}G \vee [(H \rightarrow G) \,\&\, {\sim}(H \rightarrow G)]$

(10) $(\mathcal{P} \vee \mathcal{Q}) \rightarrow {\sim}(\mathcal{P} \,\&\, \mathcal{Q})$

a. $A \rightarrow {\sim}B$

b. $(A \vee B) \rightarrow {\sim}(A \,\&\, B)$

c. $(A \vee A) \rightarrow {\sim}(A \,\&\, A)$

d. $[(A \,\&\, B) \vee (D \rightarrow E)] \rightarrow$
${\sim}[(A \,\&\, B) \,\&\, (D \rightarrow E)]$

e. $(A \,\&\, B) \rightarrow {\sim}(A \vee B)$

Part B

Use the following symbolization key to create substitution instances of the sentences below.

$\mathscr{A} : B$
$\mathscr{B} : \sim C$
$\mathscr{C} : A \rightarrow B$
$\mathscr{D} : \sim(B \,\&\, C)$
$\mathscr{E} : D \leftrightarrow E$

(1) $\sim(\mathscr{A} \leftrightarrow \mathscr{B})$
(2) $\sim\mathscr{A} \rightarrow \sim\mathscr{B}$
(3) $(\mathscr{B} \rightarrow \mathscr{C}) \,\&\, \mathscr{D}$
(4) $\sim(\mathscr{B} \,\&\, \mathscr{D})$
(5) $\mathscr{D} \rightarrow (\mathscr{B} \,\&\, \sim\mathscr{B})$
(6) $(\mathscr{A} \rightarrow \mathscr{A}) \vee (\mathscr{C} \rightarrow \mathscr{A})$
(7) $\sim\sim(\mathscr{C} \vee \mathscr{E})$
(8) $[(\mathscr{A} \rightarrow \mathscr{B}) \rightarrow \mathscr{A}] \rightarrow \mathscr{A}$
(9) $\sim\mathscr{C} \leftrightarrow (\sim\sim\mathscr{D} \,\&\, \mathscr{E})$
(10) $\mathscr{A} \,\&\, (\mathscr{B} \,\&\, (\mathscr{C} \,\&\, (\mathscr{D} \,\&\, \mathscr{E})))$

Part C Decide whether the following are examples of →E (modus ponens).

(1) $A \rightarrow B$
$B \rightarrow C$
$\therefore\ A \rightarrow C$

(2) $C \rightarrow D$
$\therefore\ C$

(3) $P \,\&\, Q$
P
$\therefore\ Q$

(4) $P \rightarrow Q$
$\therefore\ Q$

(5) $(C \,\&\, L) \rightarrow (E \vee C)$
$C \,\&\, L$
$\therefore\ E \vee C$

(6) $\sim A \rightarrow B$
$\sim B$
$\therefore\ B$

(7) $D \rightarrow E$
E
$\therefore\ D$

(8) $X \rightarrow \sim Y$
$\sim Y$
$\therefore\ X$

(9)$(P \,\&\, Q) \rightarrow (Q \,\&\, V)$
$P \,\&\, Q$
$\therefore\ Q \,\&\, V$

(10) $G \rightarrow H$
$\sim H$
$\therefore\ \sim G$

Part D Decide whether the following are examples of disjunctive syllogism.

(1) $(A \rightarrow B) \vee (X \rightarrow Y)$
$\sim A$
$\therefore X \rightarrow Y$

(2) $(C \,\&\, D) \vee E$
$(C \,\&\, D)$
$\therefore E$

(3) $[(S \vee T) \vee U] \vee V$
$\sim[(S \vee T) \vee U]$
$\therefore V$

(4) $(P \vee Q) \rightarrow R$
$\sim(P \vee Q)$
$\therefore\ R$

(5) $P \vee Q$
P
$\therefore \sim Q$

(6) $X \vee (Y \rightarrow Z)$
$\sim X$
$\therefore\ Y \rightarrow Z$

(7) $\sim(A \vee B)$
$\sim A$
$\therefore B$

(8) $(P \vee Q) \vee R$
$\sim P$
$\therefore\ Q$

(9) $(P \vee Q) \vee R$
$\therefore\ R$

(10) $A \vee (B \vee C)$
$\sim A$
$\therefore\ B \vee C$

4.2 Basic Rules for Sentential Logic

In designing a proof system, we could just start with disjunctive syllogism and modus ponens. Whenever we discovered a valid argument that could not be proved with rules we already had, we could introduce new rules. Proceeding in this way, we would have an unsystematic grab bag of rules. We might accidentally add some strange rules, and we would surely end up with more rules than we need.

Instead, we will develop what is called a SYSTEM OF NATURAL DEDUCTION. In a natural deduction system, there will be two rules for each logical operator: an introduction rule that allows us to prove a sentence that has it as the main logical operator, and an elimination rule that allows us to prove something given a sentence that has it as the main logical operator.

In addition to the rules for each logical operator, we will also have a reiteration rule. If you already have shown something in the course of a proof, the reiteration rule allows you to repeat it on a new line. We can define the RULE OF REITERATION like this

Reiteration (R)

$$\begin{array}{r|ll} m & \mathcal{A} & \\ n & \mathcal{A} & \quad \text{R } m \end{array}$$

This diagram shows how you can add lines to a proof using the rule of reiteration. As before, the script letters represent sentences of any length. The upper line shows the sentence that comes earlier in the proof, and the bottom line shows the new sentence you are allowed to write and how you justify it. The reiteration rule above is justified by one line, the line that you are reiterating. So the "R m" on line 2 of the proof means that the line is justified by the reiteration rule (R) applied to line m. The letters m and n are variables, not real line numbers. In a real proof, they might be lines 5 and 7, or lines 1 and 2, or whatever. When we define the rule, however, we use variables to underscore the point that the rule may be applied to any line that is already in the proof.

Obviously, the reiteration rule will not allow us to show anything *new.* For that, we will need more rules. The remainder of this section will give six basic introduction and elimination rules. This will be enough to do some basic proofs in SL. Sections 4.3 through 4.5 will explain introduction rules involved in fancier kinds of derivation called conditional proof and indirect proof. The remaining sections of this chapter will develop our system of natural deduction further and give you tips for playing in it.

All of the rules introduced in this chapter are summarized starting on p. 169.

Conjunction

Think for a moment: What would you need to show in order to prove $E \,\&\, F$?

Of course, you could show $E \,\&\, F$ by proving E and separately proving F. This holds even if the two conjuncts are not atomic sentences. If you can prove $[(A \vee J) \to V]$ and $[(V \to L) \leftrightarrow (F \vee N)]$, then you have effectively proved

$$[(A \vee J) \to V] \,\&\, [(V \to L) \leftrightarrow (F \vee N)].$$

So this will be our conjunction introduction rule, which we abbreviate &I:

$$\begin{array}{r|ll} m & \mathcal{A} & \\ n & \mathcal{B} & \\ & \mathcal{A} \,\&\, \mathcal{B} & \&\text{I } m,\, n \end{array} \qquad\qquad \begin{array}{r|ll} m & \mathcal{A} & \\ n & \mathcal{B} & \\ & \mathcal{B} \,\&\, \mathcal{A} & \&\text{I } m,\, n \end{array}$$

A line of proof must be justified by some rule, and here we have "&I m, n." This means: Conjunction introduction applied to line m and line n. Again, these are variables, not real line numbers; m is some line and n is some other line. If you have K on line 8 and L on line 15, you can prove $(K \,\&\, L)$ at some later point in the proof with the justification "&I 8, 15."

We have written two versions of the rule to indicate that you can write the conjuncts in any order. Even though K occurs before L in the proof, you can derive $(L \,\&\, K)$ from them using the right-hand version &I. You do not need to mark this in any special way in the proof.

Now, consider the elimination rule for conjunction. What are you entitled to conclude from a sentence like $E \,\&\, F$? Surely, you are entitled to conclude E; if $E \,\&\, F$ were true, then E would be true. Similarly, you are entitled to conclude F. This will be our conjunction elimination rule, which we abbreviate &E:

$$\begin{array}{r|ll} m & \mathcal{A} \,\&\, \mathcal{B} & \\ & \mathcal{A} & \&\text{E } m \end{array} \qquad\qquad \begin{array}{r|ll} m & \mathcal{A} \,\&\, \mathcal{B} & \\ & \mathcal{B} & \&\text{E } m \end{array}$$

When you have a conjunction on some line of a proof, you can use &E to derive either of the conjuncts. Again, we have written two versions of the rule to indicate that it can be applied to either side of the conjunction. The &E rule requires only one sentence, so we write one line number as the justification for applying it. For example, both of these moves are acceptable in derivations.

4	$A \,\&\, (B \vee C)$		10	$A \,\&\, (B \vee C)$	
5	A	& E 4	. . .	. . .	
			15	$(B \,\&\, C)$	& E 10

Some textbooks will only let you use & E on one side of a conjunction. They then make you *prove* that it works for the other side. We won't do this, because it is a pain in the neck.

Even with just these two rules, we can provide some proofs. Consider this argument.

$$\begin{array}{rl} & [(A \vee B) \rightarrow (C \vee D)] \,\&\, [(E \vee F) \rightarrow (G \vee H)] \\ \therefore & [(E \vee F) \rightarrow (G \vee H)] \,\&\, [(A \vee B) \rightarrow (C \vee D)] \end{array}$$

The main logical operator in both the premise and conclusion is a conjunction. Since the conjunction is symmetric, the argument is obviously valid. In order to provide a proof, we begin by writing down the premise. After the premises, we draw a horizontal line—everything below this line must be justified by a rule of proof. So the beginning of the proof looks like this:

1	$[(A \vee B) \rightarrow (C \vee D)] \,\&\, [(E \vee F) \rightarrow (G \vee H)]$	

From the premise, we can get each of the conjuncts by & E. The proof now looks like this:

1	$[(A \vee B) \rightarrow (C \vee D)] \,\&\, [(E \vee F) \rightarrow (G \vee H)]$	
2	$[(A \vee B) \rightarrow (C \vee D)]$	& E 1
3	$[(E \vee F) \rightarrow (G \vee H)]$	& E 1

The rule & I requires that we have each of the conjuncts available somewhere in the proof. They can be separated from one another, and they can appear in any order. So by applying the & I rule to lines 3 and 2, we arrive at the desired conclusion. The finished proof looks like this:

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

This proof is trivial, but it shows how we can use rules of proof together to demonstrate the validity of an argument form. Also: 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.

Disjunction

If M were true, then $M \vee N$ would also be true. So the disjunction introduction rule ($\vee$I) allows us to derive a disjunction if we have one of the two disjuncts:

$$\begin{array}{r|ll} m & \mathcal{A} & \\ & \mathcal{A} \vee \mathcal{B} & \vee\text{I } m \end{array} \qquad \begin{array}{r|ll} m & \mathcal{A} & \\ & \mathcal{B} \vee \mathcal{A} & \vee\text{I } m \end{array}$$

Like the rule of conjunction elimination, this rule can be applied two ways. Also notice that $\mathcal{B}$ can be *any* sentence whatsoever. So the following is a legitimate proof:

$$\begin{array}{r|ll} 1 & M & \\ \cline{2-2} 2 & M \vee ([(A \leftrightarrow B) \rightarrow (C \,\&\, D)] \leftrightarrow [E \,\&\, F]) & \vee\text{I } 1 \end{array}$$

It may seem odd that just by knowing M we can derive a conclusion that includes sentences like A, B, and the rest—sentences that have nothing to do with M. Yet the conclusion follows immediately by $\vee$I. This is as it should be: The truth conditions for the disjunction mean that, if $\mathcal{A}$ is true, then $\mathcal{A} \vee \mathcal{B}$ is true regardless of what $\mathcal{B}$ is. So the conclusion cannot be false if the premise is true; the argument is valid.

Now consider the disjunction elimination rule. What can you conclude from $M \vee N$? You cannot conclude M. It might be M's truth that makes $M \vee N$ true, as in the example above, but it might not. From $M \vee N$ alone, you cannot conclude anything about either M or N specifically. If you also knew that N was false, however, then you would be able to conclude M.

$$\begin{array}{r|ll} m & \mathcal{A} \vee \mathcal{B} & \\ n & {\sim}\mathcal{B} & \\ & \mathcal{A} & \vee\text{E } m,\, n \end{array} \qquad \begin{array}{r|ll} m & \mathcal{A} \vee \mathcal{B} & \\ n & {\sim}\mathcal{A} & \\ & \mathcal{B} & \vee\text{E } m,\, n \end{array}$$

We've seen this rule before: it is just disjunctive syllogism. Now that we are using a system of natural deduction, we are going to make it our rule for disjunction elimination ($\vee$E). Once again, the rule works on both sides of the sentential connective.

Conditionals and biconditionals

The rule for conditional introduction is complicated because it requires a whole new kind of proof, called conditional proof. We will deal with this in the next section. For now, we will only use the rule of conditional elimination.

Nothing follows from $M \to N$ alone, but if we have both $M \to N$ and M, then we can conclude N. This is another rule we've seen before: modus ponens. It now enters our system of natural deduction as the conditional elimination rule ($\to$E).

$$\begin{array}{r|ll}
m & \mathcal{A} \to \mathcal{B} & \\
n & \mathcal{A} & \\
 & \mathcal{B} & \to\text{E } m,\, n
\end{array}$$

Biconditional elimination ($\leftrightarrow$E) will be a double-barreled version of conditional elimination. If you have the left-hand subsentence of the biconditional, you can derive the right-hand subsentence. If you have the right-hand subsentence, you can derive the left-hand subsentence. This is the rule:

$$\begin{array}{r|ll}
m & \mathcal{A} \leftrightarrow \mathcal{B} & \\
n & \mathcal{A} & \\
 & \mathcal{B} & \leftrightarrow\text{E } m,\, n
\end{array}
\qquad\qquad
\begin{array}{r|ll}
m & \mathcal{A} \leftrightarrow \mathcal{B} & \\
n & \mathcal{B} & \\
 & \mathcal{A} & \leftrightarrow\text{E } m,\, n
\end{array}$$

Invalid argument forms

In section 4.1, in the last two problem parts (p. 68), we saw that sometimes an argument looks like a legitimate substitution instance of a valid argument form, but really isn't. For instance, the problem set C asked you to identify instances of modus ponens. Below I'm giving you two of the answers.

(5) Modus ponens

$$\begin{array}{rl}
 & (C \,\&\, L) \to (E \vee C) \\
 & C \,\&\, L \\
\therefore & E \vee C
\end{array}$$

(7) *Not* modus ponens.

$$\begin{array}{rl}
 & D \to E \\
 & E \\
\therefore & D
\end{array}$$

The argument on the left is an example of a valid argument, because it is an instance of modus ponens, while the argument on the right is an example of an invalid argument, because it is not an example of modus ponens. (We originally defined the terms valid and invalid on p. 11). Arguments like the one on the right, which try to trick you into thinking that they are instances of valid arguments, are called DEDUCTIVE FALLACIES. The argument on the right is specifically called the fallacy of AFFIRMING THE CONSEQUENT. In the system of natural deduction we are using in this textbook, modus ponens has been renamed "conditional elimination," but it still works the same way. So you will need to be on the lookout for deductive fallacies like affirming the consequent as you construct proofs.

Practice Exercises

Part A Some of the following arguments are legitimate instances of our six basic inference rules. The others are either invalid arguments or valid arguments that are still illegitimate because they would take multiple steps using our basic inference rules. For those that are legitimate, mark the rule that they are instances of. Mark those that are not "invalid or illegitimate argument."

(1) $$\begin{array}{ll} & R \vee S \\ \therefore & S \end{array}$$

(2) $$\begin{array}{ll} & A \,\&\, B \\ \therefore & A \end{array}$$

(3) $$\begin{array}{ll} & (A \rightarrow B) \vee (B \rightarrow A) \\ & A \rightarrow B \\ \therefore & B \rightarrow A \end{array}$$

(4) $$\begin{array}{ll} & A \rightarrow (B \,\&\, (C \vee D)) \\ & A \\ \therefore & B \,\&\, (C \vee D) \end{array}$$

(5) $$\begin{array}{ll} & P \,\&\, (Q \vee R) \\ \therefore & R \end{array}$$

(6) $$\begin{array}{ll} & P \,\&\, (Q \,\&\, R) \\ \therefore & R \end{array}$$

(7) $$\begin{array}{ll} & P \,\&\, (Q \,\&\, R) \\ \therefore & P \end{array}$$

(8) $$\begin{array}{ll} & P \\ \therefore & P \vee [A \,\&\, (B \leftrightarrow C)] \end{array}$$

(9) $$\begin{array}{ll} & A \\ \therefore & P \,\&\, (Q \rightarrow A) \end{array}$$

(10) $$\begin{array}{ll} & M \\ & D \,\&\, C \\ \therefore & M \,\&\, (D \,\&\, C) \end{array}$$

(11) $$\begin{array}{ll} & A \\ & B \,\&\, C \\ \therefore & (A \,\&\, B) \,\&\, C \end{array}$$

(12) $$\begin{array}{ll} & (X \,\&\, Y) \rightarrow (Z \,\&\, W) \\ & Z \,\&\, W \\ \therefore & X \,\&\, Y \end{array}$$

(13) $$\begin{array}{ll} & (X \,\&\, Y) \leftrightarrow (Z \,\&\, W) \\ & Z \,\&\, W \\ \therefore & X \,\&\, Y \end{array}$$

(14) $$\begin{array}{ll} & (X \,\&\, Y) \rightarrow (Z \,\&\, W) \\ & {\sim}(X \,\&\, Y) \\ \therefore & {\sim}(Z \,\&\, W) \end{array}$$

(15) $$\begin{array}{ll} & ((L \rightarrow M) \rightarrow N) \rightarrow O \\ & L \\ \therefore & M \end{array}$$

(16) $$\begin{array}{ll} & ((L \rightarrow M) \rightarrow N) \rightarrow O \\ & (L \rightarrow M) \rightarrow N \\ \therefore & O \end{array}$$

Part B

Fill in the missing pieces in the following proofs. Some are missing the justification column on the right. Some are missing the left column that contains the actual steps, and some are missing lines from both columns.

(1)

1	$W \rightarrow {\sim}B$	
2	$A \,\&\, W$	
3	$B \vee (J \,\&\, K)$	Want: K
4	W	
5	${\sim}B$	
6	$J \,\&\, K$	
7	K	

(2)

1	$A \,\&\, {\sim}B$	
2	$A \rightarrow {\sim}C$	
3	$B \vee (C \vee D)$	Want: D
4		&E 1
5		&E 1
6		→E 2, 4
7		∨E 3, 5
8		∨E 6, 7

(3)

1	$W \,\&\, B$	
2	$E \,\&\, Z$	Want: $W \,\&\, Z$
3		&E 1
4		&E 2
5	$W \,\&\, Z$	

(4)

1	$W \vee V$	
2	$I \,\&\, ({\sim}Z \rightarrow {\sim}W)$	
3	$I \rightarrow {\sim}Z$	Want: $I \,\&\, V$
4		&E 2
5		&E 2
6	${\sim}Z$	
7		→E 5, 6
8	V	
9		&I 4, 8

(5)

1	$(A \,\&\, B) \,\&\, C$	Want: $A \,\&\, (B \,\&\, C)$
2	$A \,\&\, B$	
3	C	
4	A	
5	B	
6	$B \,\&\, C$	
7	$A \,\&\, (B \,\&\, C)$	

(6)

1	$X \,\&\, (Y \,\&\, Z)$	Want: $(X \vee A) \,\&\, [(Y \vee B) \,\&\, (Z \,\&\, C)]$
2		$\&$E 1
3		$\&$E 1
4		$\&$E 3
5		$\&$E 3
6		$\vee$I 2
7		$\vee$I 4
8		$\vee$I 5
9		$\&$I 7, 8
10		$\&$I 6, 9

Part C

Derive the following.

(1) $A \rightarrow B, A \therefore A \,\&\, B$

(2) $A \leftrightarrow D, C, [(A \leftrightarrow D) \,\&\, C] \rightarrow (C \leftrightarrow B) \therefore B$

(3) $A \leftrightarrow B, B \leftrightarrow C, C \rightarrow D, A \therefore D$

(4) $(A \rightarrow {\sim}B) \,\&\, A, B \vee C \therefore C$

(5) $(A \rightarrow B) \vee (C \rightarrow (D \,\&\, E)), {\sim}(A \rightarrow B), C \therefore D$

(6) $((A \rightarrow D) \vee B) \vee C, {\sim}C, {\sim}B, A \therefore D$

(7) $A \vee B, {\sim}A, {\sim}B \therefore C$

(8) $A \rightarrow B, A, {\sim}B \therefore C$

(9) $C \vee (B \,\&\, A), {\sim}C \therefore A \vee A$

(10) $(P \vee R) \,\&\, (S \vee R), {\sim}R \,\&\, Q \therefore P \,\&\, (Q \vee R)$

(11) $(X \,\&\, Y) \rightarrow Z, X \,\&\, W, W \rightarrow Y \therefore Z$

(12) $A \vee (B \vee G), A \vee (B \vee H), {\sim}A \,\&\, {\sim}B \therefore G \,\&\, H$

(13) $A \,\&\, B, B \rightarrow C \therefore A \,\&\, (B \,\&\, C)$

Part D Translate the following arguments into SL and then show that they are valid. Be sure to write out your dictionary.

(1) If Professor Plum did it, he did it with the rope in the kitchen. Either Professor Plum or Miss Scarlett did it, and it wasn't Miss Scarlett. Therefore the murder was in the kitchen.

(2) If you are going to replace the bathtub, you might as well redo the whole bathroom. If you redo the whole bathroom, you will have to replace all the plumbing on the north side of the house. You will spend a lot of money on this project if and only if you replace the plumbing on the north side of the house. You are definitely going to replace the bathtub. Therefore you will spend a lot of money on this project.

(3) Either Caroline is happy, or Joey is happy, but not both. If Joey teases Caroline, she is not happy. Joey is teasing Caroline. Therefore, Joey is happy.

(4) Either grass is green or one of two other things: the sky is blue or snow is white. If my lawn is brown, the sky is gray, and if the sky is gray, it is not blue. If my lawn is brown, then grass is not green, and on top of that my lawn is brown. Therefore snow is white.

4.3 Conditional Proof

So far we have introduced introduction and elimination rules for the conjunction and disjunction, and elimination rules for the conditional and biconditional, but we have no introduction rules for conditionals and biconditionals, and no rules at all for negations. That's because these other rules require fancy kinds of derivations that involve putting proofs inside proofs. In this section, we will look at one of these kinds of proof, called conditional proof.

Conditional introduction

Consider this argument:

$$\begin{array}{rl} & R \lor F \\ \therefore & {\sim}R \to F \end{array}$$

The argument is certainly a valid one. What should the conditional introduction rule be, such that we can draw this conclusion?

We begin the proof in the usual way, like this:

$$\begin{array}{r|l}
1 & R \lor F \\ \cline{2-2}
\end{array}$$

If we had ${\sim}R$ as a further premise, we could derive F by the $\lor$E rule. But sadly, we do not have ${\sim}R$ as a premise, and we can't derive it directly from the premise we do have—so we cannot simply prove F. What we will do instead is start a *subproof*, a proof within the main proof. When we start a subproof, we draw another vertical line to indicate that we are no longer in the main proof. Then we write in an assumption for the subproof. This can be anything we want. Here, it will be helpful to assume ${\sim}R$. Our proof now looks like this:

$$\begin{array}{r|c|l}
1 & \multicolumn{2}{l}{R \lor F} \\ \cline{2-3}
2 & & {\sim}R \\ \cline{3-3}
\end{array}$$

It is important to notice that we are not claiming to have proved ${\sim}R$. We do not need to write in any justification for the assumption line of a subproof. You can think of the subproof as posing the question: What could we show *if* ${\sim}R$ were true? For one thing, we can derive F. So we do:

$$\begin{array}{r|c|ll}
1 & \multicolumn{2}{l}{R \lor F} & \\ \cline{2-3}
2 & & {\sim}R & \\ \cline{3-3}
3 & & F & \lor\text{E } 1,\ 2
\end{array}$$

This has shown that *if* we had ${\sim}R$ as a premise, *then* we could prove F. In effect, we have proven ${\sim}R \to F$. So the conditional introduction rule ($\to$I) will allow us to close the subproof and derive ${\sim}R \to F$ in the main proof. Our final proof looks like this:

1	$R \lor F$	
2	$\quad \sim R$	
3	$\quad F$	$\lor$E 1, 2
4	$\sim R \to F$	$\to$I 2–3

Notice that the justification for applying the →I rule is the entire subproof. Usually that will be more than just two lines.

Now that we have that example, let's lay out more precisely the rules for subproofs and then give the formal schemes for the rule of conditional and biconditional introduction.

RULE 1 You can start a subproof on any line, except the last one, and introduce any assumptions with that subproof.

RULE 2 All subproofs must be closed by the time the proof is over.

RULE 3 Subproofs may closed at any time. Once closed, they can be used to justify →I, ↔I, ∼E, and ∼I.

RULE 4 Nested subproofs must be closed before the outer subproof is closed.

RULE 5 Once the subproof is closed, lines in the subproof cannot be used in later justifications.

Rule 1 gives you great power. You can assume anything you want, at any time. But with great power, comes great responsibility, and rules 2–5 explain what your responsibilities are. Making an assumption creates the burden of starting a subproof, and subproofs must end before the proof is done. (That's why we can't start a subproof on the last line.) Closing a subproof is called *discharging* the assumptions of that subproof. So we can summarize your responsibilities this way: You cannot complete a proof until you have discharged all of the assumptions introduced in subproofs. Once the assumptions are discharged, you can use the whole subproof as a justification, but not the individual lines. So you need to know going into the subproof what you are going to use it for once you get out. As in so many parts of life, you need an exit strategy.

With those rules for subproofs in mind, the →I rule looks like this:

m	$\quad \mathcal{A}$	want $\mathcal{B}$
n	$\quad \mathcal{B}$	
	$\mathcal{A} \to \mathcal{B}$	$\to$I m–n

You still might think this gives us too much power. In logic, the ultimate sign you have too much power is that given any premise $\mathcal{A}$ you can prove any conclusion $\mathcal{B}$. Fortunately, our rules for subproofs don't let us do this. Imagine a proof that looks like this:

1	$\mathcal{A}$		
2		$\mathcal{B}$	

It may seem as if a proof like this will let you reach any conclusion $\mathcal{B}$ from any premise $\mathcal{A}$. But this is not the case. By rule 2, in order to complete a proof, you must close all of the subproofs, and we haven't done that. A subproof is only closed when the vertical line for that subproof ends. To put it another way, you can't end a proof and still have two vertical lines going.

You still might think this system gives you too much power. Maybe we can try closing the subproof and writing $\mathcal{B}$ in the main proof, like this

1	$\mathcal{A}$		
2		$\mathcal{B}$	
3		$\mathcal{B}$	R 2
4	$\mathcal{B}$		R 3

But this is wrong, too. By rule 5, once you close a subproof, you cannot refer back to individual lines inside it.

Of course, it is legitimate to do this:

1	$\mathcal{A}$		
2		$\mathcal{B}$	
3		$\mathcal{B}$	R 2
4	$\mathcal{B} \rightarrow \mathcal{B}$		→I 2–3

This should not seem so strange, though. Since $\mathcal{B}\rightarrow\mathcal{B}$ is a tautology, no particular premises should be required to validly derive it. (Indeed, as we will see, a tautology follows from any premises.)

When we introduce a subproof, we typically write what we want to derive in the right column, just like we did for the main proof. This is just so that we do not forget why we started the subproof if it goes on for five or ten lines. There is no "want" rule. It is a note to ourselves and not formally part of the proof.

Having an exit strategy when you launch a subproof is crucial. Even if you discharge an assumption properly, you might wind up with a final line that doesn't do you any good. In order to derive a conditional by →I, for instance, you must assume the antecedent of the conditional in a subproof. The →I rule also requires that the consequent of the conditional be the first line after the subproof ends. Pick your assumptions so that you wind up with a conditional that you actually need. It is always permissible to close a subproof and discharge its assumptions, but it will not be helpful to do so until you get what you want.

Now that we have the rule for conditional introduction, consider this argument:

$$P \to Q$$

$$Q \to R$$

$$\therefore\ P \to R$$

We begin the proof by writing the two premises as assumptions. Since the main logical operator in the conclusion is a conditional, we can expect to use the →I rule. For that, we need a subproof—so we write in the antecedent of the conditional as an assumption of a subproof:

```
1 | P → Q
2 | Q → R
  |-------
3 |  | P
  |  |---
```

We made P available by assuming it in a subproof, allowing us to use →E on the first premise. This gives us Q, which allows us to use →E on the second premise. Having derived R, we close the subproof. By assuming P we were able to prove R, so we apply the →I rule and finish the proof.

```
1 | P → Q
2 | Q → R
  |-------
3 |  | P        want R
  |  |---
4 |  | Q        →E 1, 3
5 |  | R        →E 2, 4
6 | P → R       →I 3–5
```

Biconditional introduction

Just as the rule for biconditional elimination was a double-headed version of conditional elimination, our rule for biconditional introduction is a double-headed version of conditional introduction. In order to derive $W \leftrightarrow X$, for instance, you must be able to prove X by assuming W *and* prove W by assuming X. The biconditional introduction rule ($\leftrightarrow$I) requires two subproofs. The subproofs can come in any order, and the second subproof does not need to come immediately after the first—but schematically, the rule works like this:

$$\begin{array}{lll}
m & \vert\ \ \vert\ \underline{\mathcal{A}} & \text{want } \mathcal{B} \\
n & \vert\ \ \vert\ \mathcal{B} & \\
p & \vert\ \ \vert\ \underline{\mathcal{B}} & \text{want } \mathcal{A} \\
q & \vert\ \ \vert\ \mathcal{A} & \\
 & \vert\ \mathcal{A} \leftrightarrow \mathcal{B} & \leftrightarrow\text{I } m\text{–}n,\ p\text{–}q
\end{array}$$

We will call any proof that uses subproofs and either $\rightarrow$I or $\leftrightarrow$I CONDITIONAL PROOF. By contrast, the first kind of proof you learned, where you only use the six basic rules, will be called DIRECT PROOF. In section 4.3 we will learn the third and final kind of proof *indirect proof.* But for now you should practice conditional proof.

Practice Exercises

Part A Fill in the blanks in the following proofs. Be sure to include the "Want" line for each subproof.

1) $\sim P \rightarrow (Q \vee R), P \vee \sim Q, \therefore \sim P \rightarrow R$

$$\begin{array}{lll}
1 & \vert\ \sim P \rightarrow (Q \vee R) & \\
2 & \vert\ \underline{P \vee \sim Q} & \text{Want: } \sim\text{P} \rightarrow\text{R} \\
3 & \vert\ \ \vert\ \underline{\sim P} & \text{Want:} \\
4 & \vert\ \ \vert\ Q \vee R & \\
5 & \vert\ \ \vert\ & \vee\text{E 2, 3} \\
6 & \vert\ \ \vert\ R & \\
7 & \vert\ \sim P \rightarrow R &
\end{array}$$

2) $B \to {\sim}D, A \to (D \lor C) \therefore A \to (B \to C)$

1	$B \to {\sim}D$	
2	$A \to (D \lor C)$	Want:
3	A	
4		Want: C
5		→E 2, 3
6		→E 1, 4
7		∨E 5, 6
8		→I 4–7
9		→E 3–8

Part B Derive the following

1) $K \,\&\, L \therefore K \leftrightarrow L$

2) $A \to (B \to C) \therefore (A \,\&\, B) \to C$

3) $A \leftrightarrow B,\ B \leftrightarrow C \therefore A \leftrightarrow C$

4) $X \leftrightarrow (A \,\&\, B),\ B \leftrightarrow Y,\ B \to A \therefore X \leftrightarrow Y$

5) ${\sim}A,\ (B \,\&\, C) \to D \therefore (A \lor B) \to (C \to D)$

6) ${\sim}W \,\&\, {\sim}E, Q \leftrightarrow D \therefore (W \lor Q) \leftrightarrow (E \lor D)$

7) $(A \,\&\, B) \leftrightarrow D, D \leftrightarrow (X \,\&\, Y), C \leftrightarrow Z \therefore A \,\&\, (B \,\&\, C) \leftrightarrow X \,\&\, (Y \,\&\, Z)$

4.4 Indirect Proof

The last two rules we need to discuss are negation introduction (∼I) and negation elimination (∼E). As with the rules of conditional and biconditional introduction, we have put off explaining the rules, because they require launching subproofs. In the case of negation introduction and elimination, these subproofs are designed to let us perform a special kind of derivation classically known as *reductio ad absurdum*, or simply *reductio.*

A *reductio* in logic is a variation on a tactic we use in ordinary arguments all the time. In arguments we often stop to imagine, for a second, that what our opponent is saying is true, and then realize that it has unacceptable consequences.

In so-called "slippery slope" arguments or "arguments from consequences," we claim that doing one thing will will lead us to doing another thing which would be horrible. For instance, you might argue that legalizing physician assisted suicide for some patients might lead to the involuntary termination of lots of other sick people. These arguments are typically not very good, but they have a basic pattern whcih we can make rigorous in our logical system. These arguments say "if my opponent wins, all hell will break loose." In logic the equivalent of all hell breaking loose is asserting a contradiction. The worst thing you can do in logic is contradict yourself. The equivalent of our opponent being right in logic would be that the sentence we are trying to prove true turns out to be false, or vice versa, the sentence we are trying to prove false is true. So in developing the rules for reductio ad absurdum, we need to find a way to say "if this sentence we true (or false), we would have to assert a contradiction."

In our system of natural deduction, this kind of proof will be known as INDIRECT PROOF.The basic rules for negation will allow for arguments like this. If we assume something and show that it leads to contradictory sentences, then we have proven the negation of the assumption. This is the negation introduction (∼I) rule:

m		$\mathcal{A}$	for reductio	m		$\mathcal{A}$	for reductio
n		$\mathcal{B}$		n		$\sim\mathcal{B}$	
$n+1$		$\sim\mathcal{B}$		$n+1$		$\mathcal{B}$	
$n+2$	$\sim\mathcal{A}$		∼I m–$n+1$	$n+2$	$\sim\mathcal{A}$		∼I m–$n+1$

For the rule to apply, the last two lines of the subproof must be an explicit contradiction: either the second sentence is the direct negation of the first, or vice versa. We write "for reductio" as a note to ourselves, a reminder of why we started the subproof. It is not formally part of the proof, and you can leave it out if you find it distracting.

To see how the rule works, suppose we want to prove a law of double negation $A \therefore \sim\sim A$

1	A		Want: ∼∼A
2		$\sim A$	for reductio
3		A	R 1
4		$\sim A$	R 2
5	$\sim\sim A$		∼I 2–4

The ∼E rule will work in much the same way. If we assume $\sim\mathcal{A}$ and show that it leads to a contradiction, we have effectively proven $\mathcal{A}$. So the rule looks like this:

m	$\mid\ {\sim}\mathcal{A}$	for reductio
n	$\mid\ \mathcal{B}$	
$n+1$	$\mid\ {\sim}\mathcal{B}$	
$n+2$	$\mathcal{A}$	$\sim$E m–$n+1$

m	$\mid\ {\sim}\mathcal{A}$	for reductio
n	$\mid\ {\sim}\mathcal{B}$	
$n+1$	$\mid\ \mathcal{B}$	
$n+2$	$\mathcal{A}$	$\sim$E m–$n+1$

Here is a simple example of negation elimination at work. We can show $L \leftrightarrow {\sim}O, L \vee {\sim}O .\therefore. L$ by assuming $\sim$L, deriving a contradiction, and then using $\sim$E.

1	$L \leftrightarrow {\sim}O$	
2	$L \vee {\sim}O$	Want: L
3	$\mid\ {\sim}L$	for reductio
4	$\mid\ {\sim}O$	$\vee$E 2, 3
5	$\mid\ L$	$\leftrightarrow$E 1, 4
6	$\mid\ {\sim}L$	R 3
7	L	$\sim$E 3–6

With the addition of $\sim$E and $\sim$I, our system of natural deduction is complete. We can now prove that any valid argument is actually valid. This is really where the fun begins.

One important bit of strategy. Sometimes, you will launch a subproof right away by assuming the negation of the conclusion to the whole argument. Other times, you will use a subproof to get a piece of the conclusion you want, or some stepping stone to the conclusion you want. Here's a simple example. Suppose you were asked to show that this argument is valid: ${\sim}(A \vee B) .\therefore. {\sim}A \,\&\, {\sim}B$. (The argument, by the way, is part of DeMorgan's Laws, some very useful equivalences which we will see more of later on.)

You need to set up the proof like this.

1	${\sim}(A \vee B)$	Want ${\sim}A \,\&\, {\sim}B$

Since you are trying to show ${\sim}A \,\&\, {\sim}B$, you could open a subproof with ${\sim}({\sim}A \,\&\, {\sim}B)$ and try to derive a contradiction, but there is an easier way to do things. Since you are tying to prove a conjunction, you can set out to prove each conjunct separately. Each conjunct, then, would get its own reductio. Let's start by assuming A in order to show $\sim A$

1	$\sim(A \vee B)$	Want $\sim A \,\&\, \sim B$
2	$\mid A$	for reductio
3	$\mid A \vee B$	$\vee$I 2
4	$\mid \sim(A \vee B)$	R 1
5	$\sim A$	$\sim$I 2–4

We can then finish the proof by showing $\sim B$ and putting it to gether with $\sim A$ and conjunction introduction.

1	$\sim(A \vee B)$	Want $\sim A \,\&\, \sim B$
2	$\mid A$	for reductio
3	$\mid A \vee B$	$\vee$I 2
4	$\mid \sim(A \vee B)$	R 1
5	$\sim A$	$\sim$I 2–4
6	$\mid B$	for reductio
7	$\mid A \vee B$	$\vee$I 6
8	$\mid \sim(A \vee B)$	R 1
9	$\sim B$	$\sim$I 7–9
10	$\sim A \,\&\, \sim B$	&E 6, 9

Now that we have all of our rules for starting and ending subproofs, we can actually prove things without any premises at all. We learned early in the text that some sentences are *tautologies*, they are always true, no matter how the world is. If a statement is a tautology, we shouldn't need any premises to prove it, because its truth doesn't depend on anything else. With subproofs, we are now able to do this. We simply start a subproof at the beginning of the derivation. Consider this proof of the law of noncontradiction: $\sim(G \,\&\, \sim G)$.

1	$\mid G \,\&\, \sim G$	for reductio
2	$\mid G$	&E 1
3	$\mid \sim G$	&E 1
4	$\sim(G \,\&\, \sim G)$	$\sim$I 1–3

Practice Exercises

Part A Fill in the blanks in the following proofs.

1) $\sim A \,\&\, \sim B. \therefore \sim(A \vee B)$.

1	$\sim A \,\&\, \sim B$	Want:
2	$\sim A$	
3	$\sim B$	
4	$A \vee B$	for reductio
5	B	
6	$\sim B$	
7	$\sim(A \vee B)$	

2) $P \to Q. \therefore \sim P \vee Q$

1	$P \to Q$	Want: ∼P ∨Q
2		for reductio
3		for reductio
4		→E 1, 3
5		∨I 4
6		R 2
7	$\sim P$	∼I 3–6
8		∨I 7
9		R 2
10	$\sim P \vee Q$	∼E 2–9

3) $\sim A \vee \sim B \;.\!\therefore\! \sim(A \,\&\, B)$

1	$\vert\ \sim A \vee \sim B$	Want: ∼(A &B)
2	$\vert\ \vert\ \underline{A \,\&\, B}$	for reductio
3	$\vert\ \vert$	&E 2
4	$\vert\ \vert$	&E 2
5	$\vert\ \vert\ \vert\ \underline{\qquad}$	for reductio
6	$\vert\ \vert\ \vert$	R 3
7	$\vert\ \vert\ \vert$	R 5
8	$\vert\ \vert\ \sim\sim A$	
9	$\vert\ \vert\ B$	
10	$\vert\ \vert\ \sim B$	
11	$\vert\ \sim(A \,\&\, B$	

4) $(X \,\&\, Y) \vee (X \,\&\, Z)$, $\sim(X \,\&\, D)$, $D \vee M \;\therefore M$

1	$\mid\ (X \,\&\, Y) \vee (X \,\&\, Z)$	
2	$\mid\ \sim(X \,\&\, D)$	
3	$\mid\ D \vee M$	Want: M
4	$\mid\ \mid$	for reductio
5	$\mid\ \mid\ \underline{D}$	$\vee$E
6	$\mid\ \mid\ \mid$	for reductio
7	$\mid\ \mid\ \mid\ \sim X \vee \sim Y$	
8	$\mid\ \mid\ \mid\ \mid$	for reductio
9	$\mid\ \mid\ \mid\ \mid\ X$	
10	$\mid\ \mid\ \mid\ \mid\ \sim X$	
11	$\mid\ \mid\ \mid\ \sim(X \,\&\, Y)$	$\sim$I 8–10
12	$\mid\ \mid\ \mid$	$\vee$E 1, 11
13	$\mid\ \mid\ \mid$	& E 12
14	$\mid\ \mid\ \mid$	R 6
15	$\mid\ \mid\ X$	
16	$\mid\ \mid\ X \,\&\, D$	
17	$\mid\ \mid\ \sim(X \,\&\, D)$	
18	$\mid\ M$	$\sim$E 4–17

Part B Derive the following using indirect derivation. You may also have to use conditional derivation.

1) $\sim\sim A \;\therefore A$

2) $P \rightarrow Q, P \rightarrow \sim Q. \therefore \sim P$

3) $A \rightarrow B$, $\sim B \;\therefore \sim A$

4) $P \vee P. \therefore P$

5) $(C \,\&\, D) \vee E \;\therefore E \vee D$

6) $M \vee (N \rightarrow M)$, $\therefore \sim M \rightarrow \sim N$

7) $A \to ({\sim}B \vee {\sim}C) \mathrel{.\,\therefore} A \to {\sim}(B \,\&\, C)$

8) ${\sim}(A \,\&\, B) \mathrel{.\,\therefore} {\sim}A \vee {\sim}B$

9) $A \vee B,\ A \to C,\ B \to C \mathrel{.\,\therefore} C$

10) ${\sim}F \to G,\ F \to H, \mathrel{.\,\therefore} G \vee H$

Part C Prove each of the following tautologies

1) $O \to O$

2) $N \vee {\sim}N$

3) ${\sim}(A \to {\sim}C) \to (A \to C)$

4) $J \leftrightarrow [J \vee (L \,\&\, {\sim}L)]$

4.5 Derived Rules

Now that we have our five rules for introduction and our five rules for elimination, our system is complete. If an argument is valid, and you can symbolize that argument in SL, you can *prove* that the argument is valid using a derivation. (Later on we will prove that our system is complete in this sense. For now, you will just have to trust me on this.) Now that our system is complete, we can really begin to play around with it and explore the exciting logical world it creates.

There's an exciting logical world created by these ten rules? Yes, yes there is. You can begin to see this by noticing that there are a lot of other interesting rules that we could have used for our introduction and elimination rules, but didn't. In many textbooks, the system of natural deduction has a disjunction elimination rule that works like this:

$$\begin{array}{r|ll}
m & \mathcal{A} \vee \mathcal{B} & \\
n & \mathcal{A} \to \mathcal{C} & \\
o & \mathcal{B} \to \mathcal{C} & \\
 & \mathcal{C} & \qquad \vee{*}\ m,\ n,\ o
\end{array}$$

You might think our system is incomplete because it lacks this alternative rule of disjunction elimination. Yet this is not the case. If you can do a proof with this rule, you can do a proof with the basic rules of the natural deduction system. You actually proved this rule as problem (9) of part B in the last section's homework. Furthermore, once you have a proof of this rule, you can use it inside other proofs whenever you think you would need a rule like $\vee*$. Simply use the proof you gave in the last homework as a sort of recipe for generating a new series of steps to get you to a line saying $\mathcal{C} \vee \mathcal{D}$

But adding lines to a proof using this recipe all the time would be a pain in the neck. What's worse, there are dozens of interesting possible rules out there, which we could have used for our introduction and elimination rules, and which we now find ourselves replacing with recipes like the one above.

Fortunately our basic set of introduction and elimination rules was meant to be expanded on. That's part of the game we are playing here. The first system of deduction created in the Western tradition was the system of geometry created by Euclid (c 300 BCE). Euclid's *Elements* began with 10 basic laws, along with definitions of terms like "point," "line," and "plane." He then went on to prove hundreds of different theorems about geometry, and each time he proved a theorem he could use that theorem to help him prove later theorems.

We can do the same thing in our system of natural deduction. What we need is a rule that will allow us to make up new rules. The new rules we add to the system will be called DERIVED RULES. Our ten rules for adding and eliminating connectives are then the AXIOMS of SL.

To explain this we will need a few more symbols. Up to now we have been using the three dots, $\therefore$, to indicate a valid argument. Now we need to be more specific about how we know the argument is valid. We will use the symbol $\vdash$ to indicate that we can prove something using a derivation in our system of natural deduction for SL. This symbol is called the *single turnstile.* In contrast, we will use the *double turnstile*, $\models$, to indicate that an argument can be proven valid using truth tables and related methods we will look at later. If the proof can be done in both directions we will write $\dashv\vdash$.

If we write $\{A, B, C\} \vdash D$, that means there is a proof out there with the premises A, B and C and the conclusion D. It is important to note that if we know $\{A, B, C\} \vdash D$, then we know that for any substitution instance of A, B, C and D. So actually we know $\{\mathcal{A}_1, \mathcal{A}_2, \ldots\} \vdash \mathcal{B}$.

Now here is our rule for adding rules.

Rule of Derived Theorem Introduction: Given a derivation in SL of some argument $A_1 \ldots A_n \vdash B$, create the rule $\mathcal{A}_1 \ldots \mathcal{A}_n \vdash \mathcal{B}$ and assign a name to it of the form "T_n"., to be read "theorem n." Now given a derivation of some theorem T_m, where $n < m$, if $\mathcal{A}_1 \ldots \mathcal{A}_n$ occur as earlier lines in a proof $x_1 \ldots x_n$, one may infer $\mathcal{B}$, and justify it "T_n, $x_1 \ldots x_n$", so long as lines none of lines $x_1 \ldots x_n$ are in a closed subproof.

Let's make our rule $\vee*$ above our first theorem. The proof of T_1 is derived simply from the recipe above.

T_1 **(Constructive Dilemma, CD):** $\{\mathcal{A} \vee \mathcal{B}, \mathcal{A} \rightarrow \mathcal{C}, \mathcal{B} \rightarrow \mathcal{C}\} \vdash \mathcal{C}$

Proof:

1	$A \vee B$			
2	$A \rightarrow C$			
3	$B \rightarrow C$			want: C
4		$\sim C$		for reductio
5			$\sim A$	for reductio
6			B	$\vee$E 2, 5
7			C	$\rightarrow$I 3, 6
8			$\sim C$	R 4
9		A		$\sim$I 5–8
10		C		$\rightarrow$E 2, 10
11		$\sim C$		R 4
12	C			$\sim$I 4–12

Informally, we will refer to T_1 as "Constructive Dilemma" or by the abbreviation "CD." Most theorems will have names and easy abbreviations like this. We will generally use the abbreviations to refer to the proofs when we use them in derivations, because they are easier to remember.

Several other important theorems have already appeard as examples or in homework problems. We'll talk about most of them in the next section, when we discuss rules of replacement. In the meantime, there is one important one we need to introduce now

T$_2$ (Modus Tollens, MT): $\{\mathcal{A} \rightarrow \mathcal{B}, \sim\mathcal{B}\} \vdash \sim\mathcal{A}$

Proof: See page 89

Now that we have some theorems, let's close by looking at how they can be used in a proof.

T$_3$ (Destructive Dilemma, DD): $\{\mathcal{A} \rightarrow \mathcal{B}, \mathcal{A} \rightarrow \mathcal{C}, \sim\mathcal{B} \vee \sim\mathcal{C}\} \vdash \sim\mathcal{A}$

1	$\vert\ A \rightarrow B$	
2	$\vert\ A \rightarrow C$	
3	$\vert\ {\sim}B \vee {\sim}C$	Want: ∼A
4	$\vert\ \vert\ A$	for reductio
5	$\vert\ \vert\ B$	→E 1, 4
6	$\vert\ \vert\ \vert\ {\sim}B$	For reductio
7	$\vert\ \vert\ \vert\ B$	R 5
8	$\vert\ \vert\ \vert\ {\sim}B$	R 6
9	$\vert\ \vert\ {\sim}{\sim}B$	∼I 6–8
10	$\vert\ \vert\ {\sim}C$	∨E 3, 9
11	$\vert\ \vert\ A$	R 4
12	$\vert\ \vert\ {\sim}A$	MT 2, 10
13	$\vert\ {\sim}A$	∼I 4–12

Practice Exercises

Part A Prove the follwoing theorems

1) T_4 (Hypothetical Syllogism, HS): $\{\mathcal{A} \rightarrow \mathcal{B}, \mathcal{B} \rightarrow \mathcal{C}\} \vdash \mathcal{A} \rightarrow \mathcal{C}$

2) T_5 (Idempotence of ∨, Idem∨): $\mathcal{A} \vee \mathcal{A} \vdash \mathcal{A}$

3) T_6 (Idempotence of & , Idem &): $\mathcal{A} \vdash \mathcal{A} \,\&\, \mathcal{A}$

Part B Provide proofs using both axioms and derived rules to show each of the following.

1) $M \,\&\, ({\sim}N \rightarrow {\sim}M) \vdash (N \,\&\, M) \vee {\sim}M$

2) $\{C \rightarrow (E \,\&\, G), {\sim}C \rightarrow G\} \vdash G$

3) $\{(Z \,\&\, K) \leftrightarrow (Y \,\&\, M), D \,\&\, (D \rightarrow M)\} \vdash Y \rightarrow Z$

4) $\{(W \vee X) \vee (Y \vee Z), X \rightarrow Y, {\sim}Z\} \vdash W \vee Y$

Part C

1) If you know that $\mathcal{A} \vdash \mathcal{B}$, what can you say about $(\mathcal{A} \,\&\, \mathcal{C}) \vdash \mathcal{B}$? Explain your answer.

2) If you know that $\mathcal{A} \vdash \mathcal{B}$, what can you say about $(\mathcal{A} \vee \mathcal{C}) \vdash \mathcal{B}$? Explain your answer.

4.6 Rules of Replacement

Very often in a derivation, you have probably been tempted to apply a rule to a part of a line. For instance, if you knew $F \rightarrow (G \,\&\, H)$ and wanted $F \rightarrow G$, you would be tempted to apply &E to just the $G \,\&\, H$ part of $F \rightarrow (G \,\&\, H)$. But, of course you aren't allowed to do that. We will now introduce some new derived rules where you can do that. These are called RULES OF REPLACEMENT, because they can be used to replace part of a sentence with a logically equivalent expression. What makes the rules of replacement different from other derived rules is that they draw on only one previous line and are symmetrical, so that you can reverse premise and conclusion and still have a valid argument. Someof the most simple examples are Theorems $7-9$, the rules of commuitivity for &, ∨, and ↔.

T$_7$ (Commutivity of &, Comm&): $(\mathcal{A} \,\&\, \mathcal{B}) \dashv\vdash (\mathcal{B} \,\&\, \mathcal{A})$
T$_8$ (Commutivity of ∨, Comm∨): $(\mathcal{A} \vee \mathcal{B}) \dashv\vdash (\mathcal{B} \vee \mathcal{A})$
T$_9$ (Commutivity of ↔, Comm↔): $(\mathcal{A} \leftrightarrow \mathcal{B}) \dashv\vdash (\mathcal{B} \leftrightarrow \mathcal{A})$

You will be asked to prove these in the homework. In the meantime, let's see an example of how they work in a proof. Suppose you wanted to prove $(M \vee P) \rightarrow (P \,\&\, M)$, $\therefore$ $(P \vee M) \rightarrow (M \,\&\, P)$ You could do it using only the basic rules, but it will be long and inconvenient. With the Comm rules, we can provide a proof easily:

1	$(M \vee P) \rightarrow (P \,\&\, M)$	
2	$(P \vee M) \rightarrow (P \,\&\, M)$	Comm& 1
3	$(P \vee M) \rightarrow (M \,\&\, P)$	Comm& 2

Formally, we can put our rule for deploying rules of replacement like this

Inserting rules of replacement: Given a theorem T of the form $\mathcal{A} \dashv\vdash \mathcal{B}$ and a line in a derivation $\mathcal{C}$ which contains in it a sentence $\mathcal{D}$, where $\mathcal{D}$ is a substitution instance of either $\mathcal{A}$ or $\mathcal{B}$, replace $\mathcal{D}$ with the equivalent substitution instance of the other side of theorem T.

Here are some other important theorems that can act as rules of replacement. Some are theorems we have already proved, while you will be asked to prove others in the homework.

$\mathbf{T}_{10}$ (Double Negation, DN): $\mathcal{A} \dashv\vdash {\sim}{\sim}\mathcal{A}$

Proof: See pages 84 and 89.

$\mathbf{T}_{11}$: ${\sim}(\mathcal{A} \vee \mathcal{B}) \dashv\vdash {\sim}\mathcal{A} \,\&\, {\sim}\mathcal{B}$

Proof: See page 86

$\mathbf{T}_{12}$: ${\sim}(\mathcal{A} \,\&\, \mathcal{B}) \dashv\vdash {\sim}\mathcal{A} \vee {\sim}\mathcal{B}$

Proof: See pages 87 and 90.

T_{11} and T_{12} are collectively known as DEMORGAN'S LAWS, and we will use the abbreviation DeM to refer to either of them in proofs.

$\mathbf{T}_{13}$: $(\mathcal{A} \rightarrow \mathcal{B}) \dashv\vdash ({\sim}\mathcal{A} \vee \mathcal{B})$

$\mathbf{T}_{14}$: $(\mathcal{A} \vee \mathcal{B}) \dashv\vdash ({\sim}\mathcal{A} \rightarrow \mathcal{B})$

T_{13} and T_{14} are collectively known as the rule of Material Conditional (MC). You will prove them in the homework.

$\mathbf{T}_{15}$ (Biconditional Exportation, ex): $\mathcal{A} \leftrightarrow \mathcal{B} \dashv\vdash (\mathcal{A} \rightarrow \mathcal{B}) \,\&\, (\mathcal{B} \rightarrow \mathcal{A})$

Proof: See the homework.

$\mathbf{T}_{16}$ (Transposition, trans): $\mathcal{A} \rightarrow \mathcal{B} \dashv\vdash {\sim}\mathcal{B} \rightarrow {\sim}\mathcal{A}$

Proof: See the homework.

To see how much these theorems can help us, consider this argument: ${\sim}(P \rightarrow Q)$, $\therefore P \,\&\, {\sim}Q$

As always, we could prove this argument using only the basic rules. With rules of replacement, though, the proof is much simpler:

1	${\sim}(P \rightarrow Q)$	
2	${\sim}({\sim}P \vee Q)$	MC 1
3	${\sim}{\sim}P \,\&\, {\sim}Q$	DeM 2
4	$P \,\&\, {\sim}Q$	DN 3

Practice Exercises

Part A Prove T_7 through T_9. You may use T_1 through T_6 in your proofs.

Part B Prove T_{13} through T_{16}. You may use T_1 through T_{12} in your proofs.

4.7 Proof Strategy

There is no simple recipe for proofs, and there is no substitute for practice. Here, though, are some rules of thumb and strategies to keep 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 logical 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. Ask what you could do to derive this new goal. For example: If your conclusion is a conditional $\mathcal{A} \rightarrow \mathcal{B}$, plan to use the →I rule. This requires starting a subproof in which you assume $\mathcal{A}$. In the subproof, you want to derive $\mathcal{B}$. Similarly, if your conclusion is a biconditional, $\mathcal{A} \leftrightarrow \mathcal{B}$, plan on using ↔I and be prepared to launch two subproofs. If you are trying to prove a single sentence letter or a negated single sentence letter, you might plan on using indirect proof.

Work forwards from what you have. When you are starting a proof, look at the premises; later, look at the sentences that you have derived so far. Think about the elimination rules for the main operators of these sentences. These will tell you what your options are. For example: If you have $A \,\&\, B$ use &E to get A and B separately. If you have $A \vee B$ see if you can find the negation of either A or B and use ∨E.

Repeat as necessary. 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 the target sentences again and ask how you might reach them. Remember, a long proof is formally 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.

Change what you are looking at. Replacement rules can often make your life easier. If a proof seems impossible, try out some different substitutions.For example: It is often difficult to prove a disjunction using the basic rules. If you want to show $\mathcal{A} \vee \mathcal{B}$, it is often easier to show $\sim\mathcal{A} \rightarrow \mathcal{B}$ and use the MC rule. Some replacement rules should become second nature. If you see a negated disjunction, for instance, you should immediately think of DeMorgan's rule.

When all else fails, try indirect proof. If you cannot find a way to show something directly, try assuming its negation. Remember that most proofs can be done either indirectly or directly. One way might be easier—or perhaps one sparks your imagination more than the other—but either one is formally legitimate.

Take a break If you are completely stuck, put down your pen and paper, get up from your computer, and do something completely different for a while. Walk the dog. Do the dishes. Take a shower. I find it especially helpful to do something physically active. Doing other desk work or watching TV doesn't have the same effect. When you come back to the problem, everything will seem clearer. Of course, if you are in a testing situation, taking a break to walk around might not be advisible. Instead, switch to another problem.

A lot of times, when you are stuck, your mind keeps trying the same solution again and again, even though you know it won't work. "If I only knew $Q \rightarrow R$," you say to yourself, "it would all work. Why can't I derive $Q \rightarrow R$!" If you go away from a problem and then come back, you might not be as focused on That One Thing that you were sure you needed, and you can find a different approach.

Practice Exercises

Part A

Show the following theorems are valid. Feel free to use T_1 through T_{16}

1) T_{17} (Weakening, WK): $\mathcal{A} \vdash \mathcal{B} \rightarrow \mathcal{A}$

2) T_{18} (Associativity of &, Ass&): $(\mathcal{A} \,\&\, \mathcal{B}) \,\&\, \mathcal{C} \dashv\vdash \mathcal{A} \,\&\, (\mathcal{B} \,\&\, \mathcal{C})$

3) T_{19} (Associativity of ∨, Ass∨): $(\mathcal{A} \vee \mathcal{B}) \vee \mathcal{C} \dashv\vdash \mathcal{A} \vee (\mathcal{B} \vee \mathcal{C})$

4) T_{20} (Associativity of ↔, Ass↔): $(\mathcal{A} \leftrightarrow \mathcal{B}) \leftrightarrow \mathcal{C} \dashv\vdash \mathcal{A} \leftrightarrow (\mathcal{B} \leftrightarrow \mathcal{C})$

4.8 Proof-Theoretic Concepts and Truth Tables

We have seen that a THEOREM is a sentence that is derivable without any premises; i.e., $\mathcal{T}$ is a theorem if and only if $\vdash \mathcal{T}$. It is not too hard to show that something is a theorem—you just have to give a proof of it. How could you show that something is *not* a theorem? If its negation is a theorem, then you could provide a proof. For example, it is easy to prove $\sim(P \,\&\, {\sim}P)$, which shows that $(P \,\&\, {\sim}P)$ cannot be a theorem. Some statements, however, are neither theorems nor the negations of theorems. Proving that something belongs to this category is more difficult. You would have to demonstrate not just that certain proof strategies fail, but that no proof is possible. Even if you fail in trying to prove a sentence in a thousand different ways, perhaps the proof is just too long and complex for you to make out.

Fortunately, we can get around this problem because there is a connection between *theorems* and *tautologies.* In Chapter 1 (page 6) we defined a tautology as a sentence that must be true because of the meaning of the words in it, no matter what the world was like. In Chapter 3 we formalized this definition for SL by saying that a sentence is a tautology if it has a T under its main connective on every line of its truth table. This corresponded to the idea that the tautology must be true no matter what the world is like. But we also have a way to show that something is *not* a tautology. If we see an F anywhere under that main connective, we know that the sentence is not a tautology. If we could show that *all* theorems are tautologies and that *only* theorems are tautologies, we can then use this strength in the system of truth tables to make up for a gap in the power of the system of natural deduction.

Similar things are true for other concepts relating to our system of natural deduction. Two sentences $\mathcal{A}$ and $\mathcal{B}$ are PROVABLY EQUIVALENT if and only if each can be derived from the other; i.e., $\mathcal{A} \vdash \mathcal{B}$ and $\mathcal{B} \vdash \mathcal{A}$ It is relatively easy to show that two sentences are provably equivalent—it just requires a pair of proofs. Showing that sentences are *not* provably equivalent would be much harder. It would be just as hard as showing that a sentence is not a theorem. (In fact, these problems are interchangeable. Can you think of a sentence that would be a theorem if and only if $\mathcal{A}$ and $\mathcal{B}$ were provably equivalent?) Here again there is an equivalent concept from the world of truth tables. With the truth tables, we saw that two sentences were *logically equivalent* if the columns under the main connectives were identical. The truth tables gave us a straightforward method for showing either that two sentences were logically equivalent or that they are not logically equivalent. So just as with theorems and tautologies, techniques for showing logical equivalence can make up for weaknesses in techniques for showing provable equivalence.

The set of sentences $\{\mathcal{A}_1, \mathcal{A}_2, \ldots\}$ is PROVABLY INCONSISTENT if and only if a contradiction is derivable from it; i.e., for some sentence $\mathcal{B}$, $\{\mathcal{A}_1, \mathcal{A}_2, \ldots\} \vdash \mathcal{B}$ and $\{\mathcal{A}_1, \mathcal{A}_2, \ldots\} \vdash {\sim}\mathcal{B}$. It is easy to show that a set is provably inconsistent: You just need to assume the sentences in the set and prove a contradiction. Showing that a set is provably *consistent* will be much harder. It would require more than just providing a proof or two; it would require showing that proofs of a certain kind are *impossible.* Truth tables can save us here again. We saw in Chapter 3 that a set of sentences is consistent if there is one line where they all are true.

Finally we have said that an argument $\mathcal{A}_1 \ldots \mathcal{A}_n \vdash \mathcal{B}$ is derivable if there is a proof of $\mathcal{B}$ using the premises $\mathcal{A}_1 \ldots \mathcal{A}_n$. Showing that an argument is not derivable is just as difficult as showing that a sentence is not a theorem. (And again, these problems are iterchangable. What sentence would a be a theorem if and only if $\mathcal{A} \vdash \mathcal{B}$ is derivable?) Here the concept of validity in truth tables will save us. An argument is valid if there is no line in the truth table where the premises are true and the conclusion is false. Because we've also used the word "valid" to talk about arguments in ordinary English and derivable arguments in SL, we should establish a specific term for arguments that are shown valid by truth tables. Let's call this notion truth table validty, or TT-validity. Again, the techniques for showing TT-validty compensate for weaknesses in dervivations.

Fortunately, for us all the equivalences we've been talking about are true. A sentence is a theorem if and only if it is a tautology. Two sentences are provably equivalent if and only if they are logically equivalent. A set of senteces is provably consistent if and only if it is consistent by the truth tables. An argument is derivable if and only if it is TT-valid. In general $\vdash$ and $\models$ are completely interchangable.

You can pick and choose when to think in terms of proofs and when to think in terms of truth tables, doing whichever is easier for a given task. Table 4.1 summarizes when it is best to give proofs and when it is best to give truth tables. In this way, proofs and truth tables give us a versatile toolkit for working with arguments. If we can translate an argument into SL, then we can measure its logical weight in a purely formal way. If it is deductively valid, we can give a formal proof; if it is invalid, we can provide a truth table.

This toolkit is incredibly convenient. It is also intuitive, because it seems natural that provability and semantic entailment should agree. Yet, do not be fooled by the similarity of the symbols "$\models$" and "$\vdash$." The fact that these two are really interchangeable is not a simple thing to prove.

Why should we think that an argument that *can be proven* is necessarily a *valid* argument? That is, why think that $\mathcal{A} \vdash \mathcal{B}$ implies $\mathcal{A} \models \mathcal{B}$?

This is the problem of SOUNDNESS. A proof system is SOUND if there are no proofs of tt-invalid arguments. Demonstrating that the proof system is sound would require showing that *any* possible proof is the proof of a valid argument. It would not be enough simply to succeed when trying to prove many valid arguments and to fail when trying to prove invalid ones.

A full-fledged proof of the soundness of SL is beyond the scope of this book, however we can give a sketch of it here. When we defined a well formed formula in SL, we used a recursive definition (see p. **??**). We could have also used recursive definitions to define a proper proof in SL and a proper truth table. (Although we didn't.) If we had these definitions, we could then use a *recursive*

	YES	NO
Is $\mathcal{A}$ a tautology?	prove $\vdash \mathcal{A}$	find the false line in the truth table for $\mathcal{A}$
Is $\mathcal{A}$ a contradiction?	prove $\vdash {\sim}\mathcal{A}$	find the true line in the truth table for $\mathcal{A}$
Is $\mathcal{A}$ contingent?	find a false line and a true line in the truth table for $\mathcal{A}$	prove $\vdash \mathcal{A}$ or $\vdash {\sim}\mathcal{A}$
Are $\mathcal{A}$ and $\mathcal{B}$ equivalent?	prove $\mathcal{A} \vdash \mathcal{B}$ and $\mathcal{B} \vdash \mathcal{A}$	find a line in the truth table for $\mathcal{A}$ and $\mathcal{B}$ where they have different truth values
Is the set $\mathbb{A}$ consistent?	find a line in truth table for $\mathbb{A}$ where all the sentences are true	Prove $\mathcal{B}$ and ${\sim}\mathcal{B}$ given the sentences in $\mathbb{A}$
Is the argument "$\mathcal{P}$, $\therefore$ $\mathcal{C}$" valid?	prove $\mathcal{P} \vdash \mathcal{C}$	give a a truth table line in which $\mathcal{P}$ is true and $\mathcal{C}$ is false

Table 4.1: When to provide a truth table and when to provide a proof.

proof to show the soundness of SL. A recursive proof works the same way a recursive definition does.With the recursive definition, we identified a group of base elements that were stipulated to be examples of the thing we were trying to define. In the case of a well formed formula, the base class was the set of sentence letters A, B, C .... We just announced that these were wffs. The second step of a recursive definition is to say that anything that is built up from your base class using certain rules also counts as an example of the thing you are defining. In the case of a definition of a wff, the rules corresponded to the five sentential connectives (see p. **??**). Once you have established a recursive definition, you can use that definition to show that all the members of the class you have defined have a certain property. You simply prove that the property is true of the members of the base class, and then you prove that the rules for extending the base class don't change the property. This is what it means to give a RECURSIVE PROOF.

Even though we don't have a recursive definition of a proof in SL, we can sketch how a recursive proof of the soundness of SL would go. Imagine a base class of one-line proofs, one for each of our eleven rules of inference. The members of this class would look like this $\{\mathcal{A}, \mathcal{B}\} \vdash \mathcal{A} \,\&\, \mathcal{B}$; $\mathcal{A} \,\&\, \mathcal{B} \vdash \mathcal{A}$; $\{\mathcal{A} \vee \mathcal{B}, {\sim}\mathcal{A}\} \vdash \mathcal{B}$ ... etc. Since some rules have a couple different forms, we would have to have add some members to this base class, for instance $\mathcal{A} \,\&\, \mathcal{B} \vdash \mathcal{B}$ Notice that these are all statements in the metalanguage. The proof that SL is sound is not a part of SL, because SL does not have the power to talk about itself.

You can use truth tables to prove to yourself that each of these one-line proofs in this base class is tt-valid. For instance the proof $\{\mathcal{A}, \mathcal{B}\} \vdash \mathcal{A} \,\&\, \mathcal{B}$ corresponds to a truth table that shows $\{\mathcal{A}, \mathcal{B}\} \models \mathcal{A} \,\&\, \mathcal{B}$ This estalishes the first part of our recursive proof.

The next step is to show that adding lines to any proof will never change a tt-valid proof into a tt-invalid one. We would need to this for each of our eleven basic rules of inference. So, for instance, for &I we need to show that for any proof $\mathcal{A}_1 \ldots \mathcal{A}_n \vdash \mathcal{B}$ anding a line where we use &I to infer $\mathcal{C} \,\&\, \mathcal{D}$, where $\mathcal{C} \,\&\, \mathcal{D}$ can be legitimately inferred from $\{\mathcal{A}_1 \ldots \mathcal{A}_n, \mathcal{B}\}$, would not change a valid proof into an invalid proof. But wait, if we can legimately derive $\mathcal{C} \,\&\, \mathcal{D}$ from the3se premises, then $\mathcal{C} \,\&\, \mathcal{D}$ must be alread available in the proof. They are either members of $\{\mathcal{A}_1 \ldots \mathcal{A}_n, \mathcal{B}\}$ or can be legitimately derived from them. As such, any truth table line in which the premises are true must be a truth table line in which $\mathcal{C}$ and $\mathcal{D}$ are true. According to the characterstic truth table for &, this means that $\mathcal{C} \,\&\, \mathcal{D}$ is also true on that line. Therefore, $\mathcal{C} \,\&\, \mathcal{D}$ validly follows from the premises. This means that using the &E rule to extend a valid proof produces another valid proof.

In order to show that the proof system is sound, we would need to show this for the other inference rules. Since the derived rules are consequences of the basic rules, it would suffice to provide similar arguments for the 11 other basic rules. This tedious exercise falls beyond the scope of this book.

So we have shown that $\mathcal{A} \vdash \mathcal{B}$ implies $\mathcal{A} \models \mathcal{B}$. What about the other direction, that is why think that *every* tt-valid argument is an argument that can be proven?

This is the problem of COMPLETENESS. A proof system is COMPLETE if there is a proof of every valid argument. Proving that a system is complete is generally harder than proving that it is sound. Proving that a system is sound amounts to showing that all of the rules of your proof system work the way they are supposed to. Showing that a system is complete means showing that you have included *all* the rules you need, that you haven't left any out. Showing this is beyond the scope of this book. The important point is that, happily, the proof system for SL is both sound and complete. This is not the case for all proof systems and all formal languages. Because it is true of SL, we can choose to give proofs or give truth tables—whichever is easier for the task at hand.

Summary of definitions

- ▷ A sentence $\mathcal{A}$ is a THEOREM if and only if $\vdash \mathcal{A}$.
- ▷ Two sentences $\mathcal{A}$ and $\mathcal{B}$ are PROVABLY EQUIVALENT if and only if $\mathcal{A} \vdash \mathcal{B}$ and $\mathcal{B} \vdash \mathcal{A}$.

- ▷ $\{\mathcal{A}_1, \mathcal{A}_2, \ldots\}$ is PROVABLY INCONSISTENT if and only if, for some sentence $\mathcal{B}$, $\{\mathcal{A}_1, \mathcal{A}_2, \ldots\} \vdash (\mathcal{B} \& {\sim}\mathcal{B})$.
- ▷ Sentential Logic is SOUND if $\vdash$ implies $\models$.
- ▷ Sentential Logic is COMPLETE if $\models$ implies $\vdash$.

Practice Exercises

Part A Use either a derivation or a truth table for each of the following.

1) Show that the sentence $A \rightarrow {\sim}A$ is not a contradiction.
2) Show that the sentence $A \leftrightarrow {\sim}A$ is a contradiction.
3) Show that $A \rightarrow (B \rightarrow A)$ is a tautology
4) Show that $A \rightarrow [((B \& C) \vee D) \rightarrow A$ is a tautology.
5) Show that $A \rightarrow (A \rightarrow B)$ is not a tautology
6) Show that the set $\{{\sim}(A \vee B), C, C \rightarrow A\}$ is inconsistent.
7) Show that the set {∼(A ∨B), ∼B, B →A} is consistent
8) Show that ${\sim}(A \vee (B \vee C)) \vdash {\sim}C$ is valid.
9) Show that ${\sim}(A \& (B \vee C)) \vdash {\sim}C$ is invalid.

Chapter 5

Quantified Logic

This chapter introduces a logical language called QL. It is a version of *quantified logic*, because it introduces words like *all* and *some*, which tell us about quantities. Quantified logic is also sometimes called *predicate logic*, because the basic units of the language are predicates and terms.

5.1 From Sentences to Predicates

Consider the following argument, which is obviously valid in English:

> If everyone knows logic, then either no one will be confused or everyone will. Everyone will be confused only if we try to believe a contradiction. Everyone knows logic.
>
> ∴ If we don't try to believe a contradiction, then no one will be confused.

In order to symbolize this in SL, we will need a symbolization key.

L: Everyone knows logic.
N: No one will be confused.
E: Everyone will be confused.
B: We try to believe a contradiction.

Notice that N and E are both about people being confused, but they are two separate sentence letters. We could not replace E with $\sim N$. Why not? $\sim N$

means “It is not the case that no one will be confused.” This would be the case if even one person were confused, so it is a long way from saying that *everyone* will be confused.

Once we have separate sentence letters for N and E, however, we erase any connection between the two. They are just two atomic sentences which might be true or false independently. In English, it could never be the case that both no one and everyone was confused. As sentences of SL, however, there is a truth value assignment for which N and E are both true.

Expressions like “no one”, “everyone”, and “anyone” are called *quantifiers.* By translating N and E as separate atomic sentences, we leave out the *quantifier structure* of the sentences. Fortunately, the quantifier structure is not what makes this argument valid. As such, we can safely ignore it. To see this, we translate the argument to SL:

$L \rightarrow (N \vee E)$
$E \rightarrow B$
L
$\therefore\ {\sim}B \rightarrow N$

This is a valid argument in SL. (You can do a truth table to check this.)

Now consider another argument. This one is also valid in English.

> Willard is a logician. All logicians wear funny hats.
> ∴ Willard wears a funny hat.

To symbolize it in SL, we define a symbolization key:

L: Willard is a logician.
A: All logicians wear funny hats.
F: Willard wears a funny hat.

Now we symbolize the argument:

L
A
$\therefore\ F$

This is *invalid* in SL. (Again, you can confirm this with a truth table.) There is something very wrong here, because this is clearly a valid argument in English.

The symbolization in SL leaves out all the important structure. Once again, the translation to SL overlooks quantifier structure: The sentence "All logicians wear funny hats" is about both logicians and hat-wearing. By not translating this structure, we lose the connection between Willard's being a logician and Willard's wearing a hat.

Some arguments with quantifier structure can be captured in SL, like the first example, even though SL ignores the quantifier structure. Other arguments are completely botched in SL, like the second example. Notice that the problem is not that we have made a mistake while symbolizing the second argument. These are the best symbolizations we can give for these arguments *in SL*.

Generally, if an argument containing quantifiers comes out *valid in SL*, then the English language argument is valid. If it comes out *invalid in SL*, then we cannot say the English language argument is invalid. The argument might be valid because of quantifier structure which the natural language argument has and which the argument in SL lacks.

Similarly, if a sentence with quantifiers comes out as a *tautology in SL*, then the English sentence is logically true. If comes out as *contingent in SL*, then this might be because of the structure of the quantifiers that gets removed when we translate into the formal language.

In order to symbolize arguments that rely on quantifier structure, we need to develop a different logical language. We will call this language quantified logic, QL.

5.2 Building Blocks of Quantified Logic

Just as sentences were the basic unit of sentential logic, predicates will be the basic unit of quantified logic. A predicate is an expression like "is a dog." This is not a sentence on its own. It is neither true nor false. In order to be true or false, we need to specify something: Who or what is it that is a dog?

The details of this will be explained in the rest of the chapter, but here is the basic idea: In QL, we will represent predicates with capital letters. For instance, we might let D stand for "_____ is a dog." We will use lower-case letters as the names of specific things. For instance, we might let b stand for Bertie. The expression Db will be a sentence in QL. It is a translation of the sentence "Bertie is a dog."

In order to represent quantifier structure, we will also have symbols that represent quantifiers. For instance, "$\exists$" will mean "There is some_____." So to say that there is a dog, we can write $\exists x Dx$; that is: There is some x such that x is a dog.

That will come later. We start by defining singular terms and predicates.

Singular Terms

In English, a SINGULAR TERM is a word or phrase that refers to a *specific* person, place, or thing. The word "dog" is not a singular term, because there are a great many dogs. The phrase "Philip's dog Bertie" is a singular term, because it refers to a specific little terrier.

A PROPER NAME is a singular term that picks out an individual without describing it. The name "Emerson" is a proper name, and the name alone does not tell you anything about Emerson. Of course, some names are traditionally given to boys and other are traditionally given to girls. If "Jack Hathaway" is used as a singular term, you might guess that it refers to a man. However, the name does not necessarily mean that the person referred to is a man—or even that the creature referred to is a person. Jack might be a giraffe for all you could tell just from the name. There is a great deal of philosophical action surrounding this issue, but the important point here is that a name is a singular term because it picks out a single, specific individual.

Other singular terms more obviously convey information about the thing to which they refer. For instance, you can tell without being told anything further that "Philip's dog Bertie" is a singular term that refers to a dog. A DEFINITE DESCRIPTION picks out an individual by means of a unique description. In English, definite descriptions are often phrases of the form "the such-and-so." They refer to *the* specific thing that matches the given description. For example, "the tallest member of Monty Python" and "the first emperor of China" are definite descriptions. A description that does not pick out a specific individual is not a definite description. "A member of Monty Python" and "an emperor of China" are not definite descriptions.

We can use proper names and definite descriptions to pick out the same thing. The proper name "Mount Rainier" names the location picked out by the definite description "the highest peak in Washington state." The expressions refer to the same place in different ways. You learn nothing from my saying that I am going to Mount Rainier, unless you already know some geography. You could guess that it is a mountain, perhaps, but even this is not a sure thing; for all you know it might be a college, like Mount Holyoke. Yet if I were to say that I was going to the highest peak in Washington state, you would know immediately that I was going to a mountain in Washington state.

In English, the specification of a singular term may depend on context; "Willard" means a specific person and not just someone named Willard; "P.D. Magnus" as a logical singular term means *me* and not the other P.D. Magnus. We live with this kind of ambiguity in English, but it is important to keep in mind that singular terms in QL must refer to just one specific thing.

In QL, we will symbolize singular terms with lower-case letters a through w. We can add subscripts if we want to use some letter more than once. So $a, b, c, \ldots w, a_1, f_{32}, j_{390}$, and m_{12} are all terms in QL.

Singular terms are called CONSTANTS because they pick out specific individuals. Note that x, y, and z are not constants in QL. They will be VARIABLES, letters which do not stand for any specific thing. We will need them when we introduce quantifiers.

Predicates

The simplest predicates are properties of individuals. They are things you can say about an object. "_____ is a dog" and "_____ is a member of Monty Python" are both predicates. In translating English sentences, the term will not always come at the beginning of the sentence: "A piano fell on _____" is also a predicate. Predicates like these are called ONE-PLACE or MONADIC, because there is only one blank to fill in. A one-place predicate and a singular term combine to make a sentence.

Other predicates are about the *relation* between two things. For instance, "_____ is bigger than _____", "_____ is to the left of _____", and "_____ owes money to _____." These are TWO-PLACE or DYADIC predicates, because they need to be filled in with two terms in order to make a sentence.

In general, you can think about predicates as schematic sentences that need to be filled out with some number of terms. Conversely, you can start with sentences and make predicates out of them by removing terms. Consider the sentence, "Vinnie borrowed the family car from Nunzio." By removing a singular term, we can recognize this sentence as using any of three different monadic predicates:

_____borrowed the family car from Nunzio.

Vinnie borrowed _____from Nunzio.

Vinnie borrowed the family car from _____.

By removing two singular terms, we can recognize three different dyadic predicates:

Vinnie borrowed _____ from _____.

_____ borrowed the family car from _____.

_____ borrowed _____ from Nunzio.

By removing all three singular terms, we can recognize one three-place or triadic predicate:

_____ borrowed _____ from _____.

If we are translating this sentence into QL, should we translate it with a one-, two-, or three-place predicate? It depends on what we want to be able to say. If the only thing that we will discuss being borrowed is the family car, then the generality of the three-place predicate is unnecessary. If the only borrowing we need to symbolize is different people borrowing the family car from Nunzio, then a one-place predicate will be enough.

In general, we can have predicates with as many places as we need. Predicates with more than one place are called polyadic. Predicates with n places, for some number n, are called n-place or n-adic.

In QL, we symbolize predicates with capital letters A through Z, with or without subscripts. When we give a symbolization key for predicates, we will not use blanks; instead, we will use variables. By convention, constants are listed at the end of the key. So we might write a key that looks like this:

Ax: x is angry.
Hx: x is happy.
$\mathbf{T_1xy}$: x is as tall or taller than y.
$\mathbf{T_2xy}$: x is as tough or tougher than y.
Bxyz: y is between x and z.
d: Donald
g: Gregor
m: Marybeth

We can symbolize sentences that use any combination of these predicates and terms. For example:

1. Donald is angry.
2. If Donald is angry, then so are Gregor and Marybeth.
3. Marybeth is at least as tall and as tough as Gregor.
4. Donald is shorter than Gregor.
5. Gregor is between Donald and Marybeth.

Sentence 1 is straightforward: Ad. The "x" in the key entry "Ax" is just a placeholder; we can replace it with other terms when translating.

Sentence 2 can be paraphrased as, "If Ad, then Ag and Am." QL has all the truth-functional connectives of SL, so we translate this as $Ad \rightarrow (Ag \,\&\, Am)$.

Sentence 3 can be translated as $T_1mg \,\&\, T_2mg$.

Sentence 4 might seem as if it requires a new predicate. If we only needed to symbolize this sentence, we could define a predicate like Sxy to mean “x is shorter than y.” However, this would ignore the logical connection between “shorter”’ and “taller.” Considered only as symbols of QL, there is no connection between S and T_1. They might mean anything at all. Instead of introducing a new predicate, we paraphrase sentence 4 using predicates already in our key: “It is not the case that Donald is as taller or taller than Gregor.” We can translate it as $\sim T_1dg$.

Sentence 5 requires that we pay careful attention to the order of terms in the key. It becomes $Bdgm$.

Practice Exercises

Part A

1) Which of the following are terms in English?

   a) Bob Barker

   b) some bees

   c) the oldest person alive now

   d) my dog Edie

   e) numbers

   f) the person with the highest grade in this class

   g) Barak Obama

   h) the fast lane on I-90 eastbound

   i) a herd of elephants

   j) a herd of Asian elephants.

2) Of the singular terms in (1), which were proper names and which were definite descriptins?

Part B Given the following translation key, translate the sentences below.

Fxy:	x is a friend of y
Sxy:	x is a sibling of y.
Oxy:	x is older than y
Txy:	x is taller than y.
Cx:	x is a child
Ax:	x is an adult.
c:	Caroline
j:	Joey
a:	Allison

o: Jane
m: Molly
r: Rob

1) Joey is a child

2) Rob is taller than Joey

3) Joey is shorter or the same height as Rob

4) Joey and Alison are friends.

5) Caroline and Joey are siblings

6) Caroline and Joey are not friends

7) Alison and Jane are friends and siblings.

8) Molly is an adult

9) Molly is not a child

10) Molly is neither older nor taller than Rob.

5.3 Quantifiers

We are now ready to introduce quantifiers. Consider these sentences:

6. Everyone is happy.
7. Everyone is at least as tough as Donald.
8. Someone is angry.

It might be tempting to translate sentence 6 as $Hd \,\&\, Hg \,\&\, Hm$. Yet this would only say that Donald, Gregor, and Marybeth are happy. We want to say that *everyone* is happy, even if we have not defined a constant to name them. In order to do this, we introduce the $\forall$ symbol. This is called the UNIVERSAL QUANTIFIER.

A quantifier must always be followed by a variable and a formula that includes that variable. We can translate sentence 6 as $\forall x Hx$. Paraphrased in English, this means "For all x, x is happy." We call $\forall x$ an *x-quantifier*. The formula that follows the quantifier is called the *scope* of the quantifier. We will give a formal definition of scope later, but intuitively it is the part of the sentence that the quantifier quantifies over. In $\forall x Hx$, the scope of the universal quantifier is Hx.

Sentence 7 can be paraphrased as, “For all x, x is at least as tough as Donald.” This translates as $\forall x T_2 x d$.

In these quantified sentences, the variable x is serving as a kind of placeholder. The expression $\forall x$ means that you can pick anyone and put them in as x. There is no special reason to use x rather than some other variable. The sentence $\forall x Hx$ means exactly the same thing as $\forall y Hy$, $\forall z Hz$, and $\forall x_5 H x_5$.

To translate sentence 8, we introduce another new symbol: the EXISTENTIAL QUANTIFIER, $\exists$. Like the universal quantifier, the existential quantifier requires a variable. Sentence 8 can be translated as $\exists x Ax$. This means that there is some x which is angry. More precisely, it means that there is *at least one* angry person. Once again, the variable is a kind of placeholder; we could just as easily have translated sentence 8 as $\exists z Az$.

Consider these further sentences:

9. No one is angry.
10. There is someone who is not happy.
11. Not everyone is happy.

Sentence 9 can be paraphrased as, “It is not the case that someone is angry.” This can be translated using negation and an existential quantifier: $\sim\exists x Ax$. Yet sentence 9 could also be paraphrased as, “Everyone is not angry.” With this in mind, it can be translated using negation and a universal quantifier: $\forall x \sim Ax$. Both of these are acceptable translations, because they are logically equivalent. The critical thing is whether the negation comes before or after the quantifier.

In general, $\forall x \mathcal{A}$ is logically equivalent to $\sim\exists x \sim\mathcal{A}$. This means that any sentence which can be symbolized with a universal quantifier can be symbolized with an existential quantifier, and vice versa. One translation might seem more natural than the other, but there is no logical different in translating with one quantifier rather than the other. For some sentences, it will simply be a matter of taste.

Sentence 10 is most naturally paraphrased as, “There is some x such that x is not happy.” This becomes $\exists x \sim Hx$. Equivalently, we could write $\sim\forall x Hx$.

Sentence 11 is most naturally translated as $\sim\forall x Hx$. This is logically equivalent to sentence 10 and so could also be translated as $\exists x \sim Hx$.

Although we have two quantifiers in QL, we could have an equivalent formal language with only one quantifier. We could proceed with only the universal quantifier, for instance, and treat the existential quantifier as a notational convention. We use square brackets [ ] to make some sentences more readable, but we know that these are really just parentheses ( ). In the same way, we could write “$\exists x$” knowing that this is just shorthand for “$\sim\forall x\sim$.” There is a choice between making logic formally simple and making it expressively simple. With QL, we opt for expressive simplicity. Both $\forall$ and $\exists$ will be symbols of QL.

Universe of Discourse

Given the symbolization key we have been using, $\forall x Hx$ means "Everyone is happy." Who is included in this *everyone*? When we use sentences like this in English, we usually do not mean everyone now alive on the Earth. We certainly do not mean everyone who was ever alive or who will ever live. We mean something more modest: everyone in the building, everyone in the class, or everyone in the room.

In order to eliminate this ambiguity, we will need to specify a UNIVERSE OF DISCOURSE—abbreviated UD. The UD is the set of things that we are talking about. So if we want to talk about people in Chicago, we define the UD to be people in the Chicago. We write this at the beginning of the symbolization key, like this:

UD: people in Chicago

The quantifiers *range over* the universe of discourse. Given this UD, $\forall x$ means "Everyone in Chicago" and $\exists x$ means "Someone in Chicago." Each constant names some member of the UD, so we can only use this UD with the symbolization key above if Donald, Gregor, and Marybeth are all in Chicago. If we want to talk about people in places besides Chicago, then we need to include those people in the UD.

In QL, the UD must be *non-empty*; that is, it must include at least one thing. It is possible to construct formal languages that allow for empty UDs, but this introduces complications.

Even allowing for a UD with just one member can produce some strange results. Suppose we have this as a symbolization key:

UD: the Eiffel Tower
Px: x is in Paris.

The sentence $\forall x Px$ might be paraphrased in English as "Everything is in Paris." Yet that would be misleading. It means that everything *in the UD* is in Paris. This UD contains only the Eiffel Tower, so with this symbolization key $\forall x Px$ just means that the Eiffel Tower is in Paris.

Non-referring terms

In QL, each constant must pick out exactly one member of the UD. A constant cannot refer to more than one thing—it is a *singular* term. Each constant must still pick out *something*. This is connected to a classic philosophical problem: the so-called problem of non-referring terms.

Medieval philosophers typically used sentences about the *chimera* to exemplify this problem. Chimera is a mythological creature; it does not really exist. Consider these two sentences:

12. Chimera is angry.
13. Chimera is not angry.

It is tempting just to define a constant to mean "chimera." The symbolization key would look like this:

UD: creatures on Earth
Ax: x is angry.
c: chimera

We could then translate sentence 12 as Ac and sentence 13 as $\sim Ac$.

Problems will arise when we ask whether these sentences are true or false.

One option is to say that sentence 12 is not true, because there is no chimera. If sentence 12 is false because it talks about a non-existent thing, then sentence 13 is false for the same reason. Yet this would mean that Ac and $\sim Ac$ would both be false. Given the truth conditions for negation, this cannot be the case.

Since we cannot say that they are both false, what should we do? Another option is to say that sentence 12 is *meaningless* because it talks about a non-existent thing. So Ac would be a meaningful expression in QL for some interpretations but not for others. Yet this would make our formal language hostage to particular interpretations. Since we are interested in logical form, we want to consider the logical force of a sentence like Ac apart from any particular interpretation. If Ac were sometimes meaningful and sometimes meaningless, we could not do that.

This is the *problem of non-referring terms*, and we will return to it later (see p. 136.) The important point for now is that each constant of QL *must* refer to something in the UD, although the UD can be any set of things that we like. If we want to symbolize arguments about mythological creatures, then we must define a UD that includes them. This option is important if we want to consider the logic of stories. We can translate a sentence like "Sherlock Holmes lived at

221B Baker Street" by including fictional characters like Sherlock Holmes in our UD.

5.4 Translating to Quantified Logic

We now have all of the pieces of QL. Translating more complicated sentences will only be a matter of knowing the right way to combine predicates, constants, quantifiers, and connectives. Consider these sentences:

14. Every coin in my pocket is a quarter.
15. Some coin on the table is a dime.
16. Not all the coins on the table are dimes.
17. None of the coins in my pocket are dimes.

In providing a symbolization key, we need to specify a UD. Since we are talking about coins in my pocket and on the table, the UD must at least contain all of those coins. Since we are not talking about anything besides coins, we let the UD be all coins. Since we are not talking about any specific coins, we do not need to define any constants. So we define this key:

UD: all coins
Px: x is in my pocket.
Tx: x is on the table.
Qx: x is a quarter.
Dx: x is a dime.

Sentence 14 is most naturally translated with a universal quantifier. The universal quantifier says something about everything in the UD, not just about the coins in my pocket. Sentence 14 means that, for any coin, *if* that coin is in my pocket *then* it is a quarter. So we can translate it as $\forall x(Px \rightarrow Qx)$.

Since sentence 14 is about coins that are both in my pocket *and* that are quarters, it might be tempting to translate it using a conjunction. However, the sentence $\forall x(Px \,\&\, Qx)$ would mean that everything in the UD is both in my pocket and a quarter: All the coins that exist are quarters in my pocket. This is would be a crazy thing to say, and it means something very different than sentence 14.

Sentence 15 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 $\exists x(Tx \,\&\, Dx)$.

Notice that we needed to use a conditional with the universal quantifier, but we used a conjunction with the existential quantifier. What would it mean to write $\exists x(Tx \rightarrow Dx)$? Probably not what you think. It means that there is some member of the UD which would satisfy the subformula; roughly speaking, there is some a such that $(Ta \rightarrow Da)$ is true. In SL, $\mathcal{A} \rightarrow \mathcal{B}$ is logically equivalent to $\sim\mathcal{A} \vee \mathcal{B}$, and this will also hold in QL. So $\exists x(Tx \rightarrow Dx)$ is true if there is some a such that $(\sim Ta \vee Da)$; i.e., it is true if some coin is *either* not on the table *or* is a dime. Of course there is a coin that is not the table—there are coins lots of other places. So $\exists x(Tx \rightarrow Dx)$ is trivially true. A conditional will usually be the natural connective to use with a universal quantifier, but a conditional within the scope of an existential quantifier can do very strange things. As a general rule, do not put conditionals in the scope of existential quantifiers unless you are sure that you need one.

Sentence 16 can be paraphrased as, "It is not the case that every coin on the table is a dime." So we can translate it as $\sim\forall x(Tx \rightarrow Dx)$. You might look at sentence 16 and paraphrase it instead as, "Some coin on the table is not a dime." You would then translate it as $\exists x(Tx \,\&\, \sim Dx)$. Although it is probably not obvious, these two translations are logically equivalent. (This is due to the logical equivalence between $\sim\forall x\mathcal{A}$ and $\exists x\sim\mathcal{A}$, along with the equivalence between $\sim(\mathcal{A} \rightarrow \mathcal{B})$ and $\mathcal{A} \,\&\, \sim\mathcal{B}$.)

Sentence 17 can be paraphrased as, "It is not the case that there is some dime in my pocket." This can be translated as $\sim\exists x(Px \,\&\, Dx)$. It might also be paraphrased as, "Everything in my pocket is a non-dime," and then could be translated as $\forall x(Px \rightarrow \sim Dx)$. Again the two translations are logically equivalent. Both are correct translations of sentence 17.

We can now translate the argument from p. 104, the one that motivated the need for quantifiers:

Willard is a logician. All logicians wear funny hats.
∴ Willard wears a funny hat.

UD: people
Lx: x is a logician.
Fx: x wears a funny hat.
w: Willard

Translating, we get:

Lw
$\forall x(Lx \rightarrow Fx)$
$\therefore Fw$

This captures the structure that was left out of the SL translation of this argument, and this is a valid argument in QL.

Empty predicates

A predicate need not apply to anything in the UD. A predicate that applies to nothing in the UD is called an EMPTY PREDICATE.

Suppose we want to symbolize these two sentences:

18. Every monkey knows sign language.
19. Some monkey knows sign language.

It is possible to write the symbolization key for these sentences in this way:

UD: animals
Mx: x is a monkey.
Sx: x knows sign language.

Sentence 18 can now be translated as $\forall x(Mx \rightarrow Sx)$.

Sentence 19 becomes $\exists x(Mx \,\&\, Sx)$.

It is tempting to say that sentence 18 entails sentence 19; that is: if every monkey knows sign language, then it must be that some monkey knows sign language. This is a valid inference in Aristotelean logic: All Ms are S, $\therefore$ some M is S. However, the entailment does not hold in QL. It is possible for the sentence $\forall x(Mx \rightarrow Sx)$ to be true even though the sentence $\exists x(Mx \,\&\, Sx)$ is false.

How can this be? The answer comes from considering whether these sentences would be true or false *if there were no monkeys.*

We have defined $\forall$ and $\exists$ in such a way that $\forall\mathcal{A}$ is equivalent to $\sim\exists\sim\mathcal{A}$. As such, the universal quantifier doesn't involve the existence of anything—only non-existence. If sentence 18 is true, then there are *no* monkeys who don't know sign language. If there were no monkeys, then $\forall x(Mx \rightarrow Sx)$ would be true and $\exists x(Mx \,\&\, Sx)$ would be false.

We allow empty predicates because we want to be able to say things like, "I do not know if there are any monkeys, but any monkeys that there are know sign language." That is, we want to be able to have predicates that do not (or might not) refer to anything.

- ▷ A UD must have *at least* one member.
- ▷ A predicate may apply to some, all, or no members of the UD.
- ▷ A constant must pick out *exactly* one member of the UD.
  A member of the UD may be picked out by one constant, many constants, or none at all.

What happens if we add an empty predicate R to the interpretation above? For example, we might define Rx to mean "x is a refrigerator." Now the sentence $\forall x(Rx \rightarrow Mx)$ will be true. This is counterintuitive, since we do not want to say that there are a whole bunch of refrigerator monkeys. It is important to remember, though, that $\forall x(Rx \rightarrow Mx)$ means that any member of the UD that is a refrigerator is a monkey. Since the UD is animals, there are no refrigerators in the UD and so the setence is trivially true.

If you were actually translating the sentence "All refrigerators are monkeys", then you would want to include appliances in the UD. Then the predicate R would not be empty and the sentence $\forall x(Rx \rightarrow Mx)$ would be false.

Picking a Universe of Discourse

The appropriate symbolization of an English language sentence in QL will depend on the symbolization key. In some ways, this is obvious: It matters whether Dx means "x is dainty" or "x is dangerous." The meaning of sentences in QL also depends on the UD.

Let Rx mean "x is a rose," let Tx mean "x has a thorn," and consider this sentence:

20. Every rose has a thorn.

It is tempting to say that sentence 20 should be translated as $\forall x(Rx \rightarrow Tx)$. If the UD contains all roses, that would be correct. Yet if the UD is merely *things on my kitchen table*, then $\forall x(Rx \rightarrow Tx)$ would only mean that every rose on my kitchen table has a thorn. If there are no roses on my kitchen table, the sentence would be trivially true.

The universal quantifier only ranges over members of the UD, so we need to include all roses in the UD in order to translate sentence 20. We have two options. First, we can restrict the UD to include all roses but *only* roses. Then sentence 20 becomes $\forall xTx$. This means that everything in the UD has a thorn;

since the UD just is the set of roses, this means that every rose has a thorn. This option can save us trouble if every sentence that we want to translate using the symbolization key is about roses.

Second, we can let the UD contain things besides roses: rhododendrons, rats, rifles, and whatall else. Then sentence 20 must be $\forall x(Rx \rightarrow Tx)$.

If we wanted the universal quantifier to mean *every* thing, without restriction, then we might try to specify a UD that contains everything. This would lead to problems. Does “everything” include things that have only been imagined, like fictional characters? On the one hand, we want to be able to symbolize arguments about Hamlet or Sherlock Holmes. So we need to have the option of including fictional characters in the UD. On the other hand, we never need to talk about every thing that does not exist. That might not even make sense. There are philosophical issues here that we will not try to address. We can avoid these difficulties by always specifying the UD. For example, if we mean to talk about plants, people, and cities, then the UD might be “living things and places.”

Suppose that we want to translate sentence 20 and, with the same symbolization key, translate these sentences:

21. Esmerelda has a rose in her hair.
22. Everyone is cross with Esmerelda.

We need a UD that includes roses (so that we can symbolize sentence 20) and a UD that includes people (so we can translate sentence 21–22.) Here is a suitable key:

UD:	people and plants
Px:	x is a person.
Rx:	x is a rose.
Tx:	x has a thorn.
Cxy:	x is cross with y.
Hxy:	x has y in their hair.
e:	Esmerelda

Since we do not have a predicate that means “. . . has a rose in her hair”, translating sentence 21 will require paraphrasing. The sentence says that there is a rose in Esmerelda’s hair; that is, there is something which is both a rose and is in Esmerelda’s hair. So we get: $\exists x(Rx \,\&\, Hex)$.

It is tempting to translate sentence 22 as $\forall x Cxe$. Unfortunately, this would mean that every member of the UD is cross with Esmerelda—both people and plants. It would mean, for instance, that the rose in Esmerelda's hair is cross with her. Of course, sentence 22 does not mean that.

"Everyone" means every person, not every member of the UD. So we can paraphrase sentence 22 as, "Every person is cross with Esmerelda." We know how to translate sentences like this: $\forall x(Px \rightarrow Cxe)$

In general, the universal quantifier can be used to mean "everyone" if the UD contains only people. If there are people and other things in the UD, then "everyone" must be treated as "every person."

Practice Exercises

Part A Using the symbolization key given, translate each English-language sentence into QL.

UD: all animals
Ax: x is an alligator.
Mx: x is a monkey.
Rx: x is a reptile.
Zx: x lives at the zoo.
Lxy: x loves y.
a: Amos
b: Bouncer
c: Cleo

1. Amos, Bouncer, and Cleo all live at the zoo.
2. Bouncer is a reptile, but not an alligator.
3. If Cleo loves Bouncer, then Bouncer is a monkey.
4. If both Bouncer and Cleo are alligators, then Amos loves them both.
5. Some reptile lives at the zoo.
6. Every alligator is a reptile.
7. Any animal that lives at the zoo is either a monkey or an alligator.
8. There are reptiles which are not alligators.
9. Cleo loves a reptile.
10. Bouncer loves all the monkeys that live at the zoo.
11. All the monkeys that Amos loves love him back.
12. If any animal is an alligator, then it is a reptile.
13. Every monkey that Cleo loves is also loved by Amos.
14. There is a monkey that loves Bouncer, but sadly Bouncer does not reciprocate this love.

Part B Our language QL is an expanded and more powerful way of dealing with a class of arguments called categorical syllogisms, which were originally identified by Aristotle and were the focus of study by Christian, Jewish and Islamic scholars in the middle ages. Christan monks gave them names who letters encoded different properties of the arguments. The vowels coded for how the sentence was quantified, and the consonants coded for things like the order of the predicates. Here are twelve of them stated as the medievals studied them. Translate each into QL.

1. Barbara: All Ms are Ps. All Ss are Ms. ∴ All Ss are Ps.
2. Cesare: No Ps are Ms. All Ss are Ms. ∴ NoSs are Ps.
3. Datisi: All Ms are Ps. Some Ms are Ss. ∴ Some Ss are Ps.
4. Calemes: All Ps are Ms. No Ms are Ss. ∴ NoSs are Ps.
5. Celarent: No Ms are Ps. All Ss are Ms. ∴ No Ss are Ps.
6. Camestres: All Ps are Ms. No Ss are Ms. ∴ No Ss are Ps.
7. Disamis: Some Ms are Ps. All Ms are Ss ∴ Some Ss are Ps.
8. Dimatis: Some Ps are Ms. All Ms are Ss. ∴ Some Ss are Ps.
9. Darii: All Ms are Ps. Some Ss are Ms. ∴ Some Ss are Ps.
10. Festino: No Ps are Ms. Some Ss are Ms. ∴ Some Ss are not Ps.
11. Ferison No Ms are Ps. Some Ms are Ss. ∴ Some Ss are not Ps.
12. Fresison No Ps are Ms Some Ms are Ss. ∴ Some Ss are not Ps.

Part C Here are the remaining 12 syllogisms studied by Aristotle and his followers.Translate each into QL.

1. Ferio: No Ms are Ps. Some Ss are Ms ∴ Some Ss are not Ps.
2. Baroco: All Ps are Ms. Some Ss are not Ms. ∴ Some Ss are not Ps.
3. Bocardo: Some Ms are not Ps. All Ms are Ss. ∴ Some Ss are not Ps.
4. Calemos: All Ps are Ms. And no Ms are Ss. ∴ Some Ss are not Ps.
5. Barbar:i All Ms are Ps. All Ss are Ms. ∴ Some Ss are Ps.
6. Camestros: All Ps are Ms. No Ss are Ms.∴ Some Ss are not Ps.
7. Darapti: All Ms are Ps. All Ms are Ss. ∴ Some Ss are Ps.

8. Fesapo: No Ps are Ms. All Ms are Ss. $\therefore$ Some Ss are not Ps.

9. Celaront: No Ms are Ps. All Ss are Ms. $\therefore$ Some Ss are not Ps.

10. Cesaro: No Ps are Ms. All Ss are Ms $\therefore$ Some Ss are not Ps.

11. Felapton: No Ms are Ps. All Ms are Ss. $\therefore$ Some Ss are not Ps.

12. Bamalip: All Ps are Ms. All Ms are Ss. $\therefore$ Some Ss are Ps.

5.5 Sentences of Quantified Logic

In this section, we provide a formal definition for a *well-formed formula* (wff) and *sentence* of QL.

Expressions

There are six kinds of symbols in QL:

predicates with subscripts, as needed	$A, B, C, \ldots, Z$ $A_1, B_1, Z_1, A_2, A_{25}, J_{375}, \ldots$
constants with subscripts, as needed	$a, b, c, \ldots, w$ $a_1, w_4, h_7, m_{32}, \ldots$
variables with subscripts, as needed	x, y, z $x_1, y_1, z_1, x_2, \ldots$
connectives	$\sim$, & $, \vee, \rightarrow, \leftrightarrow$
parentheses	(,)
quantifiers	$\forall, \exists$

We define an expression of ql as any string of symbols of SL. Take any of the symbols of QL and write them down, in any order, and you have an expression.

Well-formed formulae

By definition, a term of ql is either a constant or a variable.

An atomic formula of ql is an n-place predicate followed by n terms.

Just as we did for SL, we will give a *recursive* definition for a wff of QL. In fact, most of the definition will look like the definition of a wff of SL: Every atomic formula is a wff, and you can build new wffs by applying the sentential connectives.

We could just add a rule for each of the quantifiers and be done with it. For instance: If $\mathcal{A}$ is a wff, then $\forall x\mathcal{A}$ and $\exists x\mathcal{A}$ are wffs. However, this would allow for bizarre sentences like $\forall x\exists xDx$ and $\forall xDw$. What could these possibly mean? We could adopt some interpretation of such sentences, but instead we will write the definition of a wff so that such abominations do not even count as well-formed.

In order for $\forall x\mathcal{A}$ to be a wff, $\mathcal{A}$ must contain the variable x and must not already contain an x-quantifier. $\forall xDw$ will not count as a wff because "x" does not occur in Dw, and $\forall x\exists xDx$ will not count as a wff because $\exists xDx$ contains an x-quantifier

1. Every atomic formula is a wff.
2. If $\mathcal{A}$ is a wff, then $\sim\mathcal{A}$ is a wff.
3. If $\mathcal{A}$ and $\mathcal{B}$ are wffs, then $(\mathcal{A}\,\&\,\mathcal{B})$, is a wff.
4. If $\mathcal{A}$ and $\mathcal{B}$ are wffs, $(\mathcal{A}\vee\mathcal{B})$ is a wff.
5. If $\mathcal{A}$ and $\mathcal{B}$ are wffs, then $(\mathcal{A}\rightarrow\mathcal{B})$ is a wff.
6. If $\mathcal{A}$ and $\mathcal{B}$ are wffs, then $(\mathcal{A}\leftrightarrow\mathcal{B})$ is a wff.
7. If $\mathcal{A}$ is a wff, χ is a variable, $\mathcal{A}$ contains at least one occurrence of χ, and $\mathcal{A}$ contains no χ-quantifiers, then $\forall\chi\mathcal{A}$ is a wff.
8. If $\mathcal{A}$ is a wff, χ is a variable, $\mathcal{A}$ contains at least one occurrence of χ, and $\mathcal{A}$ contains no χ-quantifiers, then $\exists\chi\mathcal{A}$ is a wff.
9. All and only wffs of QL can be generated by applications of these rules.

Notice that the "χ" that appears in the definition above is not the variable x. It is a *meta-variable* that stands in for any variable of QL. So $\forall xAx$ is a wff, but so are $\forall yAy$, $\forall zAz$, $\forall x_4Ax_4$, and $\forall z_9Az_9$.

We can now give a formal definition for scope: The SCOPE of a quantifier is the subformula for which the quantifier is the main logical operator.

Sentences

A sentence is something that can be either true or false. In SL, every wff was a sentence. This will not be the case in QL. Consider the following symbolization key:

UD: people

Lxy: x loves y

b: Boris

Consider the expression Lzz. It is an atomic forumula: a two-place predicate followed by two terms. All atomic formula are wffs, so Lzz is a wff. Does it mean anything? You might think that it means that z loves himself, in the same way that Lbb means that Boris loves himself. Yet z is a variable; it does not name some person the way a constant would. The wff Lzz does not tell us how to interpret z. Does it mean everyone? anyone? someone? If we had a z-quantifier, it would tell us how to interpret z. For instance, $\exists zLzz$ would mean that someone loves themself.

Some formal languages treat a wff like Lzz as implicitly having a universal quantifier in front. We will not do this for QL. If you mean to say that everyone loves themself, then you need to write the quantifier: $\forall zLzz$

In order to make sense of a variable, we need a quantifier to tell us how to interpret that variable. The scope of an x-quantifier, for instance, is the the part of the formula where quantifier tells how to interpret x.

In order to be precise about this, we define a BOUND VARIABLE to be an occurrence of a variable χ that is within the scope of an χ-quantifier. A FREE VARIABLE is an occurance of a variable that is not bound.

For example, consider the wff $\forall x(Ex \vee Dy) \rightarrow \exists z(Ex \rightarrow Lzx)$. The scope of the universal quantifier $\forall x$ is $(Ex \vee Dy)$, so the first x is bound by the universal quantifier but the second and third xs are free. There is not y-quantifier, so the y is free. The scope of the existential quantifier $\exists z$ is $(Ex \rightarrow Lzx)$, so both occurrences of z are bound by it.

We define a SENTENCE of QL as a wff of QL that contains no free variables.

Notational conventions

We will adopt the same notational conventions that we did for SL (p. 39.) First, we may leave off the outermost parentheses of a formula. Second, we will use square brackets [and] in place of parentheses to increase the readability of formulae. Third, we will leave out parentheses between each pair of conjuncts when writing long series of conjunctions. Fourth, we will leave out parentheses between each pair of disjuncts when writing long series of disjunctions.

Substitution instance

If $\mathcal{A}$ is a wff, c a constant, and x a variable, then $\mathcal{A}[c|x]$ is the wff made by replacing each occurance of x in $\mathcal{A}$ with c. This is called a SUBSTITUTION INSTANCE of $\forall x\mathcal{A}$ and $\exists x\mathcal{A}$; c is called the INSTANTIATING CONSTANT.

For example: $Aa \rightarrow Ba$, $Af \rightarrow Bf$, and $Ak \rightarrow Bk$ are all substitution instances of $\forall x(Ax \rightarrow Bx)$; the instantiating constants are a, f, and k, respectively. Raj, Rdj, and Rjj are substitution instances of $\exists zRzj$; the instantiating constants are a, d, and j, respectively.

This definition will be useful later, when we define truth and derivability in QL. If $\forall xPx$ is true, then every substitution instance Pa, Pb, $Pc\ldots$ is true. To put the point informally, if everything is a P, then a is a P, b is a P, c is a P, and so on. Conversely, if some substitution instance of $\exists xPx$ such as Pa is true, then $\exists xPx$ must be true. Informally, if some specific a is a P, then there is some P.

On this definition, a substitution instance is formed by replacing just *one* variable with a constant. Also, quantifiers must be removed starting from the left. So substitutions instances of the sentence $\forall x\forall yPxy$ would include $\forall yPay$ and $\forall yPgy$, but a sentence like Pab would actually be a substitution instance of a substitution instance of $\forall x\forall yPxy$. To form it, you would first have to create the substitution instance $\forall yPay$ and then take a substitution instance of that newer sentence. The sentence $\forall xPxb$ does not count as a substitution instance of $\forall x\forall yPxy$, although it would follow from it. Setting things up this way will make this rule consistent with the rules of existential and universal elimination in Chapter 7.

Practice Exercises

Part A Identify which variables are bound and which are free.

1. $\exists xLxy \,\&\, \forall yLyx$
2. $\forall xAx \,\&\, Bx$
3. $\forall x(Ax \,\&\, Bx) \,\&\, \forall y(Cx \,\&\, Dy)$
4. $\forall x\exists y[Rxy \rightarrow (Jz \,\&\, Kx)] \vee Ryx$
5. $\forall x_1(Mx_2 \leftrightarrow Lx_2x_1) \,\&\, \exists x_2Lx_3x_2$

Part B

1. Identify which of the following are substitution instances of $\forall xRcx$: Rac, Rca, Raa, Rcb, Rbc, Rcc, Rcd, Rcx

2. Identify which of the following are substitution instances of $\exists x \forall y Lxy$: $\forall y Lby$, $\forall x Lbx$, Lab, $\exists x Lxa$

5.6 Tricky Translations

Ambiguous predicates

Suppose we just want to translate this sentence:

23. Adina is a skilled surgeon.

Let the UD be people, let Kx mean "x is a skilled surgeon", and let a mean Adina. Sentence 23 is simply Ka.

Suppose instead that we want to translate this argument:

> The hospital will only hire a skilled surgeon. All surgeons are greedy. Billy is a surgeon, but is not skilled. Therefore, Billy is greedy, but the hospital will not hire him.

We need to distinguish being a *skilled surgeon* from merely being a *surgeon*. So we define this symbolization key:

UD: people
Gx: x is greedy.
Hx: The hospital will hire x.
Rx: x is a surgeon.
Kx: x is skilled.
b: Billy

Now the argument can be translated in this way:

$$\begin{array}{rl} & \forall x\big[{\sim}(Rx \,\&\, Kx) \rightarrow {\sim}Hx\big] \\ & \forall x(Rx \rightarrow Gx) \\ & Rb \,\&\, {\sim}Kb \\ \therefore & Gb \,\&\, {\sim}Hb \end{array}$$

Next suppose that we want to translate this argument:

> Carol is a skilled surgeon and a tennis player. Therefore, Carol is a surgeon and a skilled tennis player.

If we start with the symbolization key we used for the previous argument, we could add a predicate (let Tx mean "x is a tennis player") and a constant (let c mean Carol). Then the argument becomes:

$$\begin{array}{ll} & (Rc \,\&\, Kc) \,\&\, Tc \\ \therefore & Tc \,\&\, Kc \end{array}$$

This translation is a disaster! It takes what in English is a terrible argument and translates it as a valid argument in QL. The problem is that there is a difference between being *skilled as a surgeon* and *skilled as a tennis player*. Translating this argument correctly requires two separate predicates, one for each type of skill. If we let K_1x mean "x is skilled as a surgeon" and K_2x mean "x is skilled as a tennis player," then we can symbolized the argument in this way:

$$\begin{array}{ll} & (Rc \,\&\, K_1c) \,\&\, Tc \\ \therefore & Tc \,\&\, K_2c \end{array}$$

Like the English language argument it translates, this is invalid.

The moral of these examples is that you need to be careful of symbolizing predicates in an ambiguous way. Similar problems can arise with predicates like *good*, *bad*, *big*, and *small*. Just as skilled surgeons and skilled tennis players have different skills, big dogs, big mice, and big problems are big in different ways.

Is it enough to have a predicate that means "x is a skilled surgeon", rather than two predicates "x is skilled" and ‘'x is a surgeon"? Sometimes. As sentence 23 shows, sometimes we do not need to distinguish between skilled surgeons and other surgeons.

Must we always distinguish between different ways of being skilled, good, bad, or big? No. As the argument about Billy shows, sometimes we only need to talk about one kind of skill. If you are translating an argument that is just about dogs, it is fine to define a predicate that means "x is big." If the UD includes dogs and mice, however, it is probably best to make the predicate mean "x is big for a dog."

Multiple quantifiers

Consider this following symbolization key and the sentences that follow it:

UD: People and dogs
Dx: x is a dog.
Fxy: x is a friend of y.

Oxy: x owns y.
f: Fifi
g: Gerald

24. Fifi is a dog.
25. Gerald is a dog owner.
26. Someone is a dog owner.
27. All of Gerald's friends are dog owners.
28. Every dog owner is the friend of a dog owner.

Sentence 24 is easy: Df.

Sentence 25 can be paraphrased as, "There is a dog that Gerald owns." This can be translated as $\exists x(Dx \,\&\, Ogx)$.

Sentence 26 can be paraphrased as, "There is some y such that y is a dog owner." The subsentence "y is a dog owner" is just like sentence 25, except that it is about y rather than being about Gerald. So we can translate sentence 26 as $\exists y \exists x(Dx \,\&\, Oyx)$.

Sentence 27 can be paraphrased as, "Every friend of Gerald is a dog owner." Translating part of this sentence, we get $\forall x(Fxg \rightarrow$ "x is a dog owner'). Again, it is important to recognize that "x is a dog owner" is structurally just like sentence 25. Since we already have an x-quantifier, we will need a different variable for the existential quantifier. Any other variable will do. Using z, sentence 27 can be translated as $\forall x\big[Fxg \rightarrow \exists z(Dz \,\&\, Oxz)\big]$.

Sentence 28 can be paraphrased as "For any x that is a dog owner, there is a dog owner who is x's friend." Partially translated, this becomes

$$\forall x\big[x \text{ is a dog owner} \rightarrow \exists y(y \text{ is a dog owner} \,\&\, Fxy\big].$$

Completing the translation, sentence 28 becomes

$$\forall x\big[\exists z(Dz \,\&\, Oxz) \rightarrow \exists y(\exists z(Dz \,\&\, Oyz) \,\&\, Fxy\big].$$

Consider this symbolization key and these sentences:

UD: people
Lxy: x likes y.
i: Imre.
k: Karl.

29. Imre likes everyone that Karl likes.
30. There is someone who likes everyone who likes everyone that he likes.

Sentence 29 can be partially translated as $\forall x$(Karl likes $x \rightarrow$ Imre likes x). This becomes $\forall x(Lkx \rightarrow Lix)$.

Sentence 30 is almost a tongue-twister. There is little hope of writing down the whole translation immediately, but we can proceed by small steps. An initial, partial translation might look like this:

$$\exists x \text{ everyone who likes everyone that } x \text{ likes is liked by } x$$

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 29, with y and x in place of Imre and Karl. So sentence 30 can be completely translated in this way

$$\exists x \forall y \big[\forall z(Lxz \rightarrow Lyz) \rightarrow Lxy\big]$$

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 symbolized in QL. Then translate piecemeal, replacing the daunting task of translating a long sentence with the simpler task of translating shorter formulae.

Translating pronouns

Let's look at another situation where it helps to translate in small steps. For the next several examples, we will use this symbolization key:

UD: people
Gx: x can play bass guitar.
Rx: x is a rock star.
l: Lemmy

Now consider these sentences:

31. If Lemmy can play bass guitar, then he is a rock star.
32. If a person can play bass guitar, then he is a rock star.

Sentence 31 and sentence 32 have the same consequent ("... he is a rock star"), but they cannot be translated in the same way. It helps to paraphrase the original sentences, replacing pronouns with explicit references.

Sentence 31 can be paraphrased as, "If Lemmy can play bass guitar, then *Lemmy* is a rockstar." This can obviously be translated as $Gl \rightarrow Rl$.

Sentence 32 must be paraphrased differently: "If a person can play bass guitar, then *that person* is a rock star." This sentence is not about any particular person, so we need a variable. Translating halfway, we can paraphrase the sentence as, "For any person x, if x can play bass guitar, then x is a rockstar." Now this can be translated as $\forall x(Gx \rightarrow Rx)$. This is the same as, '"Everyone who can play bass guitar is a rock star."

Consider these further sentences:

33. If anyone can play bass guitar, then Lemmy can.
34. If anyone can play bass guitar, then he or she is a rock star.

These two sentences have the same antecedent ("If anyone can play guitar..."), but they have different logical structures.

Sentence 33 can be paraphrased, "If someone can play bass guitar, then Lemmy can play bass guitar." The antecedent and consequent are separate sentences, so it can be symbolized with a conditional as the main logical operator: $\exists xGx \rightarrow Gl$.

Sentence 34 can be paraphrased, "For anyone, if that one can play bass guitar, then that one is a rock star." It would be a mistake to symbolize this with an existential quantifier, because it is talking about everybody. The sentence is equivalent to "All bass guitar players are rock stars." It is best translated as $\forall x(Gx \rightarrow Rx)$.

The English words "any" and "anyone" should typically be translated using quantifiers. As these two examples show, they sometimes call for an existential quantifier (as in sentence 33) and sometimes for a universal quantifier (as in sentence 34). If you have a hard time determining which is required, paraphrase the sentence with an English language sentence that uses words besides "any" or "anyone."

Quantifiers and scope

In the sentence $\exists xGx \rightarrow Gl$, the scope of the existential quantifier is the expression Gx. Would it matter if the scope of the quantifier were the whole sentence? That is, does the sentence $\exists x(Gx \rightarrow Gl)$ mean something different?

With the key given above, $\exists xGx \rightarrow Gl$ means that if there is some bass guitarist, then Lemmy is a bass guitarist. $\exists x(Gx \rightarrow Gl)$ would mean that there is some person such that if that person were a bass guitarist, then Lemmy would be a bass guitarist. Recall that the conditional here is a material conditional; the conditional is true if the antecedent is false. Let the constant p denote the author of this book, someone who is certainly not a guitarist. The sentence $Gp \rightarrow Gl$ is true because Gp is false. Since someone (namely p) satisfies the sentence, then $\exists x(Gx \rightarrow Gl)$ is true. The sentence is true because there is a non-guitarist, regardless of Lemmy's skill with the bass guitar.

Something strange happened when we changed the scope of the quantifier, because the conditional in QL is a material conditional. In order to keep the meaning the same, we would have to change the quantifier: $\exists x Gx \rightarrow Gl$ means the same thing as $\forall x(Gx \rightarrow Gl)$, and $\exists x(Gx \rightarrow Gl)$ means the same thing as $\forall x Gx \rightarrow Gl$.

This oddity does not arise with other connectives or if the variable is in the consequent of the conditional. For example, $\exists x Gx \,\&\, Gl$ means the same thing as $\exists x(Gx \,\&\, Gl)$, and $Gl \rightarrow \exists x Gx$ means the same things as $\exists x(Gl \rightarrow Gx)$.

Practice Exercises

Part A Using the symbolization key given, translate each English-language sentence into QL.

UD: all animals
Dx: x is a dog.
Sx: x likes samurai movies.
Lxy: x is larger than y.
b: Bertie
e: Emerson
f: Fergis

1. Bertie is a dog who likes samurai movies.
2. Bertie, Emerson, and Fergis are all dogs.
3. Emerson is larger than Bertie, and Fergis is larger than Emerson.
4. All dogs like samurai movies.
5. Only dogs like samurai movies.
6. There is a dog that is larger than Emerson.
7. If there is a dog larger than Fergis, then there is a dog larger than Emerson.
8. No animal that likes samurai movies is larger than Emerson.
9. No dog is larger than Fergis.
10. Any animal that dislikes samurai movies is larger than Bertie.
11. There is an animal that is between Bertie and Emerson in size.
12. There is no dog that is between Bertie and Emerson in size.

13. No dog is larger than itself.
14. Every dog is larger than some dog.
15. There is an animal that is smaller than every dog.
16. If there is an animal that is larger than any dog, then that animal does not like samurai movies.

Part B For each argument, write a symbolization key and translate the argument into QL.

1. Nothing on my desk escapes my attention. There is a computer on my desk. As such, there is a computer that does not escape my attention.
2. All my dreams are black and white. Old TV shows are in black and white. Therefore, some of my dreams are old TV shows.
3. Neither Holmes nor Watson has been to Australia. A person could see a kangaroo only if they had been to Australia or to a zoo. Although Watson has not seen a kangaroo, Holmes has. Therefore, Holmes has been to a zoo.
4. No one expects the Spanish Inquisition. No one knows the troubles I've seen. Therefore, anyone who expects the Spanish Inquisition knows the troubles I've seen.
5. An antelope is bigger than a bread box. I am thinking of something that is no bigger than a bread box, and it is either an antelope or a cantaloupe. As such, I am thinking of a cantaloupe.
6. All babies are illogical. Nobody who is illogical can manage a crocodile. Berthold is a baby. Therefore, Berthold is unable to manage a crocodile.

Part C Using the symbolization key given, translate each English-language sentence into QL.

UD: candies
Cx: x has chocolate in it.
Mx: x has marzipan in it.
Sx: x has sugar in it.
Tx: Boris has tried x.
Bxy: x is better than y.

1. Boris has never tried any candy.
2. Marzipan is always made with sugar.
3. Some candy is sugar-free.
4. The very best candy is chocolate.
5. No candy is better than itself.
6. Boris has never tried sugar-free chocolate.
7. Boris has tried marzipan and chocolate, but never together.
8. Any candy with chocolate is better than any candy without it.

9. Any candy with chocolate and marzipan is better than any candy that lacks both.

Part D Using the symbolization key given, translate each English-language sentence into QL.

UD:	people and dishes at a potluck
Rx:	x has run out.
Tx:	x is on the table.
Fx:	x is food.
Px:	x is a person.
Lxy:	x likes y.
e:	Eli
f:	Francesca
g:	the guacamole

1. All the food is on the table.
2. If the guacamole has not run out, then it is on the table.
3. Everyone likes the guacamole.
4. If anyone likes the guacamole, then Eli does.
5. Francesca only likes the dishes that have run out.
6. Francesca likes no one, and no one likes Francesca.
7. Eli likes anyone who likes the guacamole.
8. Eli likes anyone who likes the people that he likes.
9. If there is a person on the table already, then all of the food must have run out.

Part E Using the symbolization key given, translate each English-language sentence into QL.

UD:	people
Dx:	x dances ballet.
Fx:	x is female.
Mx:	x is male.
Cxy:	x is a child of y.
Sxy:	x is a sibling of y.
e:	Elmer
j:	Jane
p:	Patrick

1. All of Patrick's children are ballet dancers.
2. Jane is Patrick's daughter.
3. Patrick has a daughter.
4. Jane is an only child.

5. All of Patrick's daughters dance ballet.
6. Patrick has no sons.
7. Jane is Elmer's niece.
8. Patrick is Elmer's brother.
9. Patrick's brothers have no children.
10. Jane is an aunt.
11. Everyone who dances ballet has a sister who also dances ballet.
12. Every man who dances ballet is the child of someone who dances ballet.

5.7 Identity

Consider this sentence:

35. Pavel owes money to everyone else.

Let the UD be people; this will allow us to translate "everyone" as a universal quantifier. Let Oxy mean "x owes money to y," and let p mean Pavel. Now we can symbolize sentence 35 as $\forall x Opx$. Unfortunately, this translation has some odd consequences. It says that Pavel owes money to every member of the UD, including Pavel; it entails that Pavel owes money to himself. However, sentence 35 does not say that Pavel owes money to himself; he owes money to everyone *else*. This is a problem, because $\forall x Opx$ is the best translation we can give of this sentence into QL.

The solution is to add another symbol to QL. The symbol $=$ is a two-place predicate. Since it has a special logical meaning, we write it a bit differently: For two terms t_1 and t_2, $t_1 = t_2$ is an atomic formula.

The predicate $x = y$ means "x is identical to y." This does not mean merely that x and y are indistinguishable or that all of the same predicates are true of them. Rather, it means that x and y are the very same thing.

When we write $x \neq y$, we mean that x and y are not identical. There is no reason to introduce this as an additional predicate. Instead, $x \neq y$ is an abbreviation of $\sim(x = y)$.

Now suppose we want to symbolize this sentence:

36. Pavel is Mister Checkov.

Let the constant c mean Mister Checkov. Sentence 36 can be symbolized as $p = c$. This means that the constants p and c both refer to the same guy.

This is all well and good, but how does it help with sentence 35? That sentence can be paraphrased as, "Everyone who is not Pavel is owed money by Pavel." This is a sentence structure we already know how to symbolize: "For all x, if x is not Pavel, then x is owed money by Pavel." In QL with identity, this becomes $\forall x(x \neq p \rightarrow Opx)$.

In addition to sentences that use the word "else," identity will be helpful when symbolizing some sentences that contain the words "besides" and "only." Consider these examples:

37. No one besides Pavel owes money to Hikaru.
38. Only Pavel owes Hikaru money.

We add the constant h, which means Hikaru.

Sentence 37 can be paraphrased as, "No one who is not Pavel owes money to Hikaru." This can be translated as $\sim\exists x(x \neq p \,\&\, Oxh)$.

Sentence 38 can be paraphrased as, "Pavel owes Hikaru *and* no one besides Pavel owes Hikaru money." We have already translated one of the conjuncts, and the other is straightforward. Sentence 38 becomes $Oph \,\&\, \sim\exists x(x \neq p \,\&\, Oxh)$.

Expressions of quantity

We can also use identity to say how many things there are of a particular kind. For example, consider these sentences:

39. There is at least one apple on the table.
40. There are at least two apples on the table.
41. There are at least three apples on the table.

Let the UD be *things on the table*, and let Ax mean "x is an apple."

Sentence 39 does not require identity. It can be translated adequately as $\exists xAx$: There is some apple on the table—perhaps many, but at least one.

It might be tempting to also translate sentence 40 without identity. Yet consider the sentence $\exists x\exists y(Ax \,\&\, Ay)$. It means that there is some apple x in the UD and some apple y in the UD. Since nothing precludes x and y from picking out the same member of the UD, this would be true even if there were only one apple. In order to make sure that there are two *different* apples, we need an identity predicate. Sentence 40 needs to say that the two apples that exist are not identical, so it can be translated as $\exists x\exists y(Ax \,\&\, Ay \,\&\, x \neq y)$.

Sentence 41 requires talking about three different apples. It can be translated as $\exists x \exists y \exists z(Ax \,\&\, Ay \,\&\, Az \,\&\, x \neq y \,\&\, y \neq z \,\&\, x \neq z)$.

Continuing in this way, we could translate "There are at least n apples on the table." There is a summary of how to symbolize sentences like these on p. 168.

Now consider these sentences:

42. There is at most one apple on the table.
43. There are at most two apples on the table.

Sentence 42 can be paraphrased as, "It is not the case that there are at least *two* apples on the table." This is just the negation of sentence 40:

$$\sim\exists x \exists y(Ax \,\&\, Ay \,\&\, x \neq y)$$

Sentence 42 can also be approached in another way. It means that any apples that there are on the table must be the selfsame apple, so it can be translated as $\forall x \forall y\big[(Ax \,\&\, Ay) \rightarrow x = y\big]$. The two translations are logically equivalent, so both are correct.

In a similar way, sentence 43 can be translated in two equivalent ways. It can be paraphrased as, "It is not the case that there are *three* or more distinct apples", so it can be translated as the negation of sentence 41. Using universal quantifiers, it can also be translated as

$$\forall x \forall y \forall z\big[(Ax \,\&\, Ay \,\&\, Az) \rightarrow (x = y \vee x = z \vee y = z)\big].$$

See p. 168 for the general case.

The examples above are sentences about apples, but the logical structure of the sentences translates mathematical inequalities like $a \geq 3$, $a \leq 2$, and so on. We also want to be able to translate statements of equality which say exactly how many things there are. For example:

44. There is exactly one apple on the table.
45. There are exactly two apples on the table.

Sentence 44 can be paraphrased as, "There is *at least* one apple on the table, and there is *at most* one apple on the table." This is just the conjunction of sentence 39 and sentence 42: $\exists x Ax \,\&\, \forall x \forall y\big[(Ax \,\&\, Ay) \rightarrow x = y\big]$. This is a somewhat complicated way of going about it. It is perhaps more straightforward to paraphrase sentence 44 as "There is a thing which is the only apple on the table." Thought of in this way, the sentence can be translated $\exists x\big[Ax \,\&\, \sim\exists y(Ay \,\&\, x \neq y)\big]$.

Similarly, sentence 45 may be paraphrased as, "There are two different apples on the table, and these are the only apples on the table." This can be translated as $\exists x\exists y\big[Ax \,\&\, Ay \,\&\, x \neq y \,\&\, {\sim}\exists z(Az \,\&\, x \neq z \,\&\, y \neq z)\big]$.

Finally, consider this sentence:

46. There are at most two things on the table.

It might be tempting to add a predicate so that Tx would mean "x is a thing on the table." However, this is unnecessary. Since the UD is the set of things on the table, all members of the UD are on the table. If we want to talk about a *thing on the table*, we need only use a quantifier. Sentence 46 can be symbolized like sentence 43 (which said that there were at most two apples), but leaving out the predicate entirely. That is, sentence 46 can be translated as $\forall x\forall y\forall z(x = y \vee x = z \vee y = z)$.

Techniques for symbolizing expressions of quantity ("at most", "at least", and "exactly") are summarized on p. 168.

Definite descriptions

Recall that a constant of QL must refer to some member of the UD. This constraint allows us to avoid the problem of non-referring terms. Given a UD that included only actually existing creatures but a constant c that meant "chimera" (a mythical creature), sentences containing c would become impossible to evaluate.

The most widely influential solution to this problem was introduced by Bertrand Russell in 1905. Russell asked how we should understand this sentence:

47. The present king of France is bald.

The phrase "the present king of France" is supposed to pick out an individual by means of a definite description. However, there was no king of France in 1905 and there is none now. Since the description is a non-referring term, we cannot just define a constant to mean "the present king of France" and translate the sentence as Kf.

Russell's idea was that sentences that contain definite descriptions have a different logical structure than sentences that contain proper names, even though they share the same grammatical form. What do we mean when we use an unproblematic, referring description, like "the highest peak in Washington state"? We mean that there is such a peak, because we could not talk about it otherwise. We also mean that it is the only such peak. If there was another peak in Washington state of exactly the same height as Mount Rainier, then Mount Rainier would not be *the* highest peak.

According to this analysis, sentence 47 is saying three things. First, it makes an *existence* claim: There is some present king of France. Second, it makes a *uniqueness* claim: This guy is the only present king of France. Third, it makes a claim of *predication*: This guy is bald.

In order to symbolize definite descriptions in this way, we need the identity predicate. Without it, we could not translate the uniqueness claim which (according to Russell) is implicit in the definite description.

Let the UD be *people actually living*, let Fx mean "x is the present king of France," and let Bx mean "x is bald." Sentence 47 can then be translated as $\exists x\big[Fx \,\&\, {\sim}\exists y(Fy \,\&\, x \neq y) \,\&\, Bx\big]$. This says that there is some guy who is the present king of France, he is the only present king of France, and he is bald.

Understood in this way, sentence 47 is meaningful but false. It says that this guy exists, but he does not.

The problem of non-referring terms is most vexing when we try to translate negations. So consider this sentence:

48. The present king of France is not bald.

According to Russell, this sentence is ambiguous in English. It could mean either of two things:

48a. It is not the case that the present king of France is bald.
48b. The present king of France is non-bald.

Both possible meanings negate sentence 47, but they put the negation in different places.

Sentence 48a is called a WIDE-SCOPE NEGATION, because it negates the entire sentence. It can be translated as ${\sim}\exists x\big[Fx \,\&\, {\sim}\exists y(Fy \,\&\, x \neq y) \,\&\, Bx\big]$. This does not say anything about the present king of France, but rather says that some sentence about the present king of France is false. Since sentence 47 if false, sentence 48a is true.

Sentence 48b says something about the present king of France. It says that he lacks the property of baldness. Like sentence 47, it makes an existence claim and a uniqueness claim; it just denies the claim of predication. This is called NARROW-SCOPE NEGATION. It can be translated as $\exists x\big[Fx \,\&\, {\sim}\exists y(Fy \,\&\, x \neq y) \,\&\, {\sim}Bx\big]$. Since there is no present king of France, this sentence is false.

Russell's theory of definite descriptions resolves the problem of non-referring terms and also explains why it seemed so paradoxical. Before we distinguished between the wide-scope and narrow-scope negations, it seemed that sentences like 48 should be both true and false. By showing that such sentences are ambiguous, Russell showed that they are true understood one way but false understood another way.

For a more detailed discussion of Russell's theory of definite descriptions, including objections to it, see the Peter Ludlow's entry “descriptions”’ in *The Stanford Encyclopedia of Philosophy*: Summer 2005 edition, edited by Edward N. Zalta, `http://plato.stanford.edu/archives/sum2005/entries/descriptions/`

Practice Exercises

Part A Using the symbolization key given, translate each English-language sentence into QL with identity. The last sentence is ambiguous and can be translated two ways; you should provide both translations. (Hint: Identity is only required for the last four sentences.)

UD: people
Kx: x knows the combination to the safe.
Sx: x is a spy.
Vx: x is a vegetarian.
Txy: x trusts y.
h: Hofthor
i: Ingmar

1. Hofthor is a spy, but no vegetarian is a spy.
2. No one knows the combination to the safe unless Ingmar does.
3. No spy knows the combination to the safe.
4. Neither Hofthor nor Ingmar is a vegetarian.
5. Hofthor trusts a vegetarian.
6. Everyone who trusts Ingmar trusts a vegetarian.
7. Everyone who trusts Ingmar trusts someone who trusts a vegetarian.
8. Only Ingmar knows the combination to the safe.
9. Ingmar trusts Hofthor, but no one else.
10. The person who knows the combination to the safe is a vegetarian.
11. The person who knows the combination to the safe is not a spy.

Part B Using the symbolization key given, translate each English-language sentence into QL with identity. The last two sentences are ambiguous and can be translated two ways; you should provide both translations for each.

UD: cards in a standard deck
Bx: x is black.
Cx: x is a club.
Dx: x is a deuce.
Jx: x is a jack.
Mx: x is a man with an axe.
Ox: x is one-eyed.
Wx: x is wild.

1. All clubs are black cards.
2. There are no wild cards.
3. There are at least two clubs.
4. There is more than one one-eyed jack.
5. There are at most two one-eyed jacks.
6. There are two black jacks.
7. There are four deuces.
8. The deuce of clubs is a black card.
9. One-eyed jacks and the man with the axe are wild.
10. If the deuce of clubs is wild, then there is exactly one wild card.
11. The man with the axe is not a jack.
12. The deuce of clubs is not the man with the axe.

Part C Using the symbolization key given, translate each English-language sentence into QL with identity. The last two sentences are ambiguous and can be translated two ways; you should provide both translations for each.

UD: animals in the world
Bx: x is in Farmer Brown's field.
Hx: x is a horse.
Px: x is a Pegasus.
Wx: x has wings.

1. There are at least three horses in the world.
2. There are at least three animals in the world.
3. There is more than one horse in Farmer Brown's field.
4. There are three horses in Farmer Brown's field.
5. There is a single winged creature in Farmer Brown's field; any other creatures in the field must be wingless.
6. The Pegasus is a winged horse.
7. The animal in Farmer Brown's field is not a horse.
8. The horse in Farmer Brown's field does not have wings.

Chapter 6

Semantics for Quantified Logic

6.1 Creating models in Quantified Logic

In Chapter 3 we developed the truth table system, a semantic method for evaluating sentences and arguments in SL. The method was called semantic, because it involved establishing meanings for all the non-logical parts of a sentence, in this case, the sentence letters. These were called non-logical because their meaning was not fixed by the rules of SL. The logical part of the language was the system of connectives and parentheses. As it turned out, the logical parts of our language were truth functional: the meaning of the larger sentences built up with the connectives and parentheses was simply a function of the truth of the smaller parts. Because the system of sentential connectives and parentheses is truth functional, our semantic system only needed to look at one aspect of the meaning of the sentence letters, their truth value. Thus an interpretation of a sentence in SL simply turned out to be a truth assignment on the sentence letters. That's why you spent Chapter 3 merrily writing Ts and Fs under sentence letters. Each little row of Ts and Fs represented one way of assigning meaning to a sentence or argument, and all the little rows in the truth table represented all the possible ways of assigning meaning to the sentence or argument (at least as far as truth was concerned.)

In this chapter we are going to develop similar methods for QL. Because QL is a more complicated system than SL developing semantics for QL will be much more complicated, and there will be greater limitations for what we can do with these methods.

Basic Models

The first thing we need to do is to find an equivalent in QL of a little row of Ts and Fs beneath a sentence in SL. We called this little row of Ts and Fs an interpretation, because it gave one aspect of the meaning of the parts of the sentence, namely their truth values. To come up with an interpretation of a sentence in QL, we will need to look at more than the truth or falsity of sentences, though. We have, after all, broken open the atomic sentence to look at its subatomic parts, and now we will need to assign meaning to these parts. The way to start is to look at the symbolization keys we created when we were translating sentences in and out of QL. The symbolization key contained a universe of discourse (UD), a meaning for each predicate, and an object picked out by each term. For example:

UD:	comic book characters
Fx:	x fights crime.
b:	Batman
w:	Bruce Wayne

Given this symbolization key, we can translate sentences like this

Fb: Batman fights crime.
Fw: Bruce Wayne fights crime.
$Fb \rightarrow Fw$: If Batman fights crime, then Bruce Wayne fights crime.

This symbolization key, along with some basic knowledge of superhero stories, gives us enough information to figure out that the sentence Fb is true. (Notice that the sentence Fb is not true *just because* of the interpretation. The way we interpret the sentence, plus the way the world is, makes the sentence true.)

This is the information we need to develop an interpretation of sentences in QL. We are not just assigning truth values anymore. We need to get deeper into meaning here, and look at the *reference* of the parts of the sentence. REFERENCE is an aspect of meaning that deals with the way bits of language pick out or identify objects in the real world. For our purposes the real world is the universe of discourse, the set of objects we are talking about. To explain how a singular term like b refers, we need only point out the member of the UD is refers to, in this case, Batman. In more technical terms, Batman is the *referent* of B. For our purposes it is enough to define a referent like this: the REFERENT of a term in QL is the unique object picked out by that term.

(Notice that here to explain what b refers to, I have been forced to simply use another singular term in another language, the name "Batman." To really pick out the object referred to, I would need to draw your attention to something in the actual world, for instance by holding up a comic book and pointing at the character on the cover.)

We want to use similar methods to talk about the meaning of predicates. To do this we will talk about the *extension* of a predicate. The EXTENSION of a predicate is the set of objects in the UD that the predicate applies to. So if b is in the extension of the predicate F then the sentence Fb is true.

(Identifying the extension of a predicate like F again forces us to rely on another language, English. We can say that the extension of F includes Batman, Superman, Green Lantern, etc. The situation is worse because F has an indefinitely large extension, so we have relied on the English predicate "fights crime" to describe it.)

All this means that we are able to talk about the meaning of sentences in QL entirely in terms of sets. We use curly brackets { and } to denote sets. The members of the set can be listed in any order, separated by commas. The fact that sets can be in any order is important, because it means that {foo, bar} and {bar, foo} are the same set. It is possible to have a set with no members in it. This is called the EMPTY SET. The empty set is sometimes written as {}, but usually it is written as the single symbol $\emptyset$.

We are now able to give the equivalent of a line in a truth table for QL. An interpretation in QL will be called a *model.* A MODEL of sentences or arguments in QL consists of a set that is the universe of discourse, individual members of that set that are the referents of the singular terms in the sentences or arguments, and subsets of the universe of discourse which are the extensions of the predicates used in the sentences or arguments.

To see how this works imagine I have a bunch of sentences in QL, which include the predicate H and the singular term f. Now consider this symbolization key:

UD: People who played as part of the Three Stooges
Hx: x had head hair.
f: Mister Fine

What is the model that corresponds to this interpretation? There were six people who played as part of the Three Stooges over the years, so the UD will have six members: Larry Fine, Moe Howard, Curly Howard, Shemp Howard, Joe Besser, and Curly Joe DeRita. Curly, Joe, and Curly Joe were the only completely bald stooges. The result is this model:

$$\begin{aligned} \text{UD} &= \{\text{Larry, Curly, Moe, Shemp, Joe, Curly Joe}\} \\ \text{extension}(H) &= \{\text{Larry, Moe, Shemp}\} \\ \text{referent}(f) &= \text{Larry} \end{aligned}$$

You do not need to know anything about the Three Stooges in order to evaluate whether sentences are true or false in this *model.* Hf is true, since the referent of f (Larry) is in the extension of H. Both $\exists x Hx$ and $\exists x{\sim}Hx$ are true, since there is at least one member of the UD that is in the extension of H and at least one member that is not in the extension of H. In this way, the model captures all of the formal significance of the interpretation.

Models for multiplace predicates

Now consider this interpretation:

UD: whole numbers less than 10
Ex: x is even.
Nx: x is negative.
Lxy: x is less than y.
Txyz: x times y equals z.

What is the model that goes with this interpretation? The UD is the set $\{1, 2, 3, 4, 5, 6, 7, 8, 9\}$.

The extension of a one-place predicate like E or N is just the subset of the UD of which the predicate is true. The extension of E is the subset $\{2, 4, 6, 8\}$. There are many even numbers besides these four, but these are the only members of the UD that are even. There are no negative numbers in the UD, so N has an empty extension; i.e. extension$(N) = \emptyset$.

The extension of a two-place predicate like L is somewhat vexing. It seems as if the extension of L ought to contain 1, since 1 is less than all the other numbers; it ought to contain 2, since 2 is less than all of the other numbers besides 1; and so on. Every member of the UD besides 9 is less than some member of the UD. What would happen if we just wrote extension$(L) = \{1, 2, 3, 4, 5, 6, 7, 8\}$?

The problem is that sets can be written in any order, so this would be the same as writing extension$(L) = \{8, 7, 6, 5, 4, 3, 2, 1\}$. This does not tell us which of the members of the set are less than which other members.

We need some way of showing that 1 is less than 8 but that 8 is not less than 1. The solution is to have the extension of L consist of pairs of numbers. An ORDERED PAIR is like a set with two members, except that the order *does* matter. We write ordered pairs with angle brackets < and >. The ordered pair <foo, bar> is different than the ordered pair <bar, foo>. The extension of L is a collection of ordered pairs, all of the pairs of numbers in the UD such that the first number is less than the second. Writing this out completely:

> extension(L) = {<1,2>, <1,3>, <1,4>, <1,5>, <1,6>, <1,7>, <1,8>, <1,9>, <2,3>, <2,4>, <2,5>, <2,6>, <2,7>, <2,8>, <2,9>, <3,4>, <3,5>, <3,6>, <3,7>, <3,8>, <3,9>, <4,5>, <4,6>, <4,7>, <4,8>, <4,9>, <5,6>, <5,7>, <5,8>, <5,9>, <6,7>, <6,8>, <6,9>, <7,8>, <7,9>, <8,9>}

Three-place predicates will work similarly; the extension of a three-place predicate is a set of ordered triples where the predicate is true of those three things *in that order*. So the extension of T in this model will contain ordered triples like <2,4,8>, because $2 \times 4 = 8$.

Generally, the extension of an n-place predicate is a set of all ordered n-tuples $<a_1, a_2, \ldots, a_n>$ such that a_1–a_n are members of the UD and the predicate is true of a_1–a_n in that order.

Models for identity

Identity is a special predicate of QL. We write it a bit differently than other two-place predicates: $x = y$ instead of Ixy. We also do not need to include it in a symbolization key. The sentence $x = y$ always means "x is identical to y," and it cannot be interpreted to mean anything else. In the same way, when you construct a model, you do not get to pick and choose which ordered pairs go into the extension of the identity predicate. It always contains just the ordered pair of each object in the UD with itself.

The sentence $\forall x Ixx$, which contains an ordinary two-place predicate, is contingent. Whether it is true for an interpretation depends on how you interpret I, and whether it is true in a model depends on the extension of I.

The sentence $\forall x\ x = x$ is a tautology. The extension of identity will always make it true.

Notice that although identity always has the same interpretation, it does not always have the same extension. The extension of identity depends on the UD. If the UD in a model is the set {Doug}, then extension(=) in that model is {<Doug, Doug>}. If the UD is the set {Doug, Omar}, then extension(=) in that model is {<Doug, Doug>, <Omar, Omar>}. And so on.

If the referent of two constants is the same, then anything which is true of one is true of the other. For example, if referent(a) = referent(b), then $Aa \leftrightarrow Ab$, $Ba \leftrightarrow Bb$, $Ca \leftrightarrow Cb$, $Rca \leftrightarrow Rcb$, $\forall x Rxa \leftrightarrow \forall x Rxb$, and so on for any two sentences containing a and b. In metaphysics, this is called principle of the indiscernibility of identicals

In our system, the reverse of this principle is not true. It is possible that anything which is true of a is also true of b, yet for a and b still to have different referents. This may seem puzzling, but it is easy to construct a model that shows this. Consider this model:

$$\begin{aligned}
\text{UD} &= \{\text{Rosencrantz, Guildenstern}\}\\
\text{referent}(a) &= \text{Rosencrantz}\\
\text{referent}(b) &= \text{Guildenstern}\\
\text{for all predicates } \mathscr{P}\text{, extension}(\mathscr{P}) &= \emptyset\\
\text{extension}(=) &= \{<\text{Rosencrantz, Rosencrantz}>,\\
&\quad\; <\text{Guildenstern, Guildenstern}>\}
\end{aligned}$$

This specifies an extension for every predicate of QL: All the infinitely-many predicates are empty. This means that both Aa and Ab are false, and they are equivalent; both Ba and Bb are false; and so on for any two sentences that contain a and b. Yet a and b refer to different things. We have written out the extension of identity to make this clear: The ordered pair $<\text{referent}(a), \text{referent}(b)>$ is not in it. In this model, $a = b$ is false and $a \neq b$ is true.

Practice Exercises

Part A Determine whether each sentence is true or false in the model given.

$$\begin{aligned}
\text{UD} &= \{\text{Corwin, Benedict}\}\\
\text{extension}(A) &= \{\text{Corwin, Benedict}\}\\
\text{extension}(B) &= \{\text{Benedict}\}\\
\text{extension}(N) &= \emptyset\\
\text{referent}(c) &= \text{Corwin}
\end{aligned}$$

1. Bc
2. $Ac \leftrightarrow {\sim}Nc$
3. $Nc \rightarrow (Ac \vee Bc)$
4. $\forall x Ax$
5. $\forall x {\sim} Bx$
6. $\exists x(Ax \,\&\, Bx)$
7. $\exists x(Ax \rightarrow Nx)$
8. $\forall x(Nx \vee {\sim}Nx)$
9. $\exists x Bx \rightarrow \forall x Ax$

Part B Determine whether each sentence is true or false in the model given.

$$\begin{aligned}
\text{UD} &= \{\text{Waylan, Willy, Johnny}\}\\
\text{extension}(H) &= \{\text{Waylan, Willy, Johnny}\}\\
\text{extension}(W) &= \{\text{Waylan, Willy}\}\\
\text{extension}(R) &= \{<\text{Waylan, Willy}>,<\text{Willy, Johnny}>,<\text{Johnny, Waylan}>\}\\
\text{referent}(m) &= \text{Johnny}
\end{aligned}$$

1. $\exists x(Rxm \,\&\, Rmx)$
2. $\forall x(Rxm \vee Rmx)$
3. $\forall x(Hx \leftrightarrow Wx)$
4. $\forall x(Rxm \rightarrow Wx)$
5. $\forall x\big[Wx \rightarrow (Hx \,\&\, Wx)\big]$
6. $\exists xRxx$
7. $\exists x\exists yRxy$
8. $\forall x\forall yRxy$
9. $\forall x\forall y(Rxy \vee Ryx)$
10. $\forall x\forall y\forall z\big[(Rxy \,\&\, Ryz) \rightarrow Rxz\big]$

Part C Determine whether each sentence is true or false in the model given.

$$\begin{aligned}
\text{UD} &= \{\text{Lemmy, Courtney, Eddy}\}\\
\text{extension}(G) &= \{\text{Lemmy, Courtney, Eddy}\}\\
\text{extension}(H) &= \{\text{Courtney}\}\\
\text{extension}(M) &= \{\text{Lemmy, Eddy}\}\\
\text{referent}(c) &= \text{Courtney}\\
\text{referent}(e) &= \text{Eddy}
\end{aligned}$$

1. Hc
2. He
3. $Mc \vee Me$
4. $Gc \vee {\sim}Gc$
5. $Mc \rightarrow Gc$
6. $\exists xHx$
7. $\forall xHx$
8. $\exists x{\sim}Mx$
9. $\exists x(Hx \,\&\, Gx)$
10. $\exists x(Mx \,\&\, Gx)$
11. $\forall x(Hx \vee Mx)$
12. $\exists xHx \,\&\, \exists xMx$
13. $\forall x(Hx \leftrightarrow {\sim}Mx)$
14. $\exists xGx \,\&\, \exists x{\sim}Gx$
15. $\forall x\exists y(Gx \,\&\, Hy)$

Part D Write out the model that corresponds to the interpretation given.

UD: natural numbers from 10 to 13
Ox: x is odd.
Sx: x is less than 7.
Tx: x is a two-digit number.
Ux: x is thought to be unlucky.
Nxy: x is the next number after y.

6.2 Working with Models

We can use models in some of the ways we used truth tables to evaluate sentences and arguments. We cannot construct a finite number of models to show conclusively that a sentence is a tautology or a contradiction. As we shall see, this is because there are infinitely many ways to model a sentence, and no single way to talk about all of them, the way we talked about all the lines in a truth table. We can, however, construct two models to show conclusively that a sentence is neither a tautology nor a contradiction, and is instead contingent. Similarly, we cannot construct models to prove two sentences equivalent, but we can use them to show that two sentences are not equivalent. We cannot use models to show that a set of sentences is inconsistent, but we can use models to show that a set of sentences is consistent. Finally, we cannot construct a finite number of models to show that an argument is valid, but we can use one to show that an argument is invalid.

We will use the double turnstile symbol for QL much as we did for SL. "$\mathcal{A} \models \mathcal{B}$"' means that "$\mathcal{A}$ entails $\mathcal{B}$": When $\mathcal{A}$ and $\mathcal{B}$ are two sentences of QL, $\mathcal{A} \models \mathcal{B}$ means that there is no model in which $\mathcal{A}$ is true and $\mathcal{B}$ is false. $\models \mathcal{A}$ means that $\mathcal{A}$ is true in every model.

This allows us to give definitions for various concepts in QL. Because we are using the same symbol, these definitions will look similar to the definitions in SL. Remember, however, that the definitions in QL are in terms of *models* rather than in terms of truth value assignments.

> A TAUTOLOGY IN QL is a sentence $\mathcal{A}$ that is true in every model; i.e., $\models \mathcal{A}$.
>
> A CONTRADICTION IN QL is a sentence $\mathcal{A}$ that is false in every model; i.e., $\models {\sim}\mathcal{A}$.
>
> A sentence is CONTINGENT IN QL if and only if it is neither a tautology nor a contradiction.
>
> Two sentences $\mathcal{A}$ and $\mathcal{B}$ are LOGICALLY EQUIVALENT IN QL if and only if both $\mathcal{A} \models \mathcal{B}$ and $\mathcal{B} \models \mathcal{A}$.
>
> The set $\{\mathcal{A}_1, \mathcal{A}_2, \mathcal{A}_3, \cdots\}$ is CONSISTENT IN QL if and only if there is at least one model in which all of the sentences are true. The set is INCONSISTENT IN QL if and if only there is no such model.
>
> An argument " $\mathcal{P}_1, \mathcal{P}_2, \cdots, \therefore \mathcal{C}$ " is VALID IN QL if and only if there is no model in which all of the premises are true and the conclusion is false; i.e., $\{\mathcal{P}_1, \mathcal{P}_2, \cdots\} \models \mathcal{C}$. It is INVALID IN QL otherwise.

Models to show contingency

Suppose we want to show that $\forall x Axx \rightarrow Bd$ is *not* a tautology. This requires showing that the sentence is not true in every model; i.e., that it is false in some model. If we can provide just one model in which the sentence 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.

To construct such a model, we start with a UD. It will be easier to specify extensions for predicates if we have a small UD, so start with a UD that has just one member. Formally, this single member might be anything, so let's just call it α.

We want $\forall x Axx$ to be true, so we want all members of the UD to be paired with themselves in the extension of A; this means that the extension of A must be $\{< \alpha, \alpha >\}$.

We want Bd to be false, so the referent of d must not be in the extension of B. We give B an empty extension.

Since α is the only member of the UD, it must be the referent of d. The model we have constructed looks like this:

$$\begin{aligned}
\text{UD} &= \{\alpha\} \\
\text{extension}(A) &= \{< \alpha, \alpha >\} \\
\text{extension}(B) &= \emptyset \\
\text{referent}(d) &= \alpha
\end{aligned}$$

Strictly speaking, a model specifies an extension for *every* predicate of QL and a referent for *every* constant. As such, it is generally impossible to write down a complete model. That would require writing down infinitely many extensions and infinitely many referents. However, we do not need to consider every predicate in order to show that there are models in which $\forall x Axx \rightarrow Bd$ is false. Predicates like H and constants like f_{13} make no difference to the truth or falsity of this sentence. It is enough to specify extensions for A and B and a referent for d, as we have done. This provides a *partial model* in which the sentence is false.

Perhaps you are wondering: What is α? What does the predicate A mean in English? The partial model could correspond to an interpretation like this one:

UD: Paris
Axy**:** x is in the same country as y.
Bx**:** x was founded in the 20th century.
d**:** the City of Lights

However, we don't have to say that this model corresponds to any particular interpretation of the sentence in English in order to know that the sentence $\forall x Axx \rightarrow Bd$ is not a tautology. We could have made our one-object universe of discourse contain only Mahatma Gandhi, or a pebble on a beach in Africa, or the number 2. As long as the predicate and terms were given the right reference, the sentence would come out false. Thus in the future we can evalaute sentences and arguments using UDs with arbitrarily named elements, like α, β, γ, etc.

We use the same method to 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 one such partial model:

$$\begin{aligned} \text{UD} &= \{\alpha\} \\ \text{extension}(A) &= \{<\alpha, \alpha>\} \\ \text{extension}(B) &= \{\alpha\} \\ \text{referent}(d) &= \alpha \end{aligned}$$

We have now shown that $\forall x Axx \rightarrow Bd$ is neither a tautology nor a contradiction. By the definition of "contingent in QL," this means that $\forall x Axx \rightarrow Bd$ is contingent. In general, 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.

Notice, however, that we cannot show that a sentence actually is a tautology or a contradiction using one or two models this way. For a sentence to be a tautology, it must be true in every possible model. Similarly, if a sentence is a contradiction, it is false in all possible models. But there are infinitely many possible models for any sentence, and we don't have any tools in this text that will let us reason about all of them at once.

Models to show non-equivalence

Suppose 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 want one of them to be true and the other to be false. We start by specifying a UD. Again, we make the UD small so that we can specify extensions easily. We will need at least two members. Let the UD be $\{\alpha, \beta\}$. (If we chose a UD with only one member, the two sentences would end up with the same truth value. In order to see why, try constructing some partial models with one-member UDs.)

We can make $\exists x Sx$ true by including something in the extension of S, and we can make $\forall x Sx$ false by leaving something out of the extension of S. It does not matter which one we include and which one we leave out. Making α the only S, we get a partial model that looks like this:

$$\begin{aligned} \text{UD} &= \{\alpha, \beta\} \\ \text{extension}(S) &= \{\alpha\} \end{aligned}$$

This partial model shows that the two sentences are *not* logically equivalent.

Notice, though, that we cannot show that two sentences *are* logically equivalent by simply producing a model. If we claim that two sentences are logically equivalent, we are once again making a claim about every possible model.

Models to show consistency

Suppose I wanted to show that the set of sentences $\{\exists x Fx, \exists x{\sim}Fx, \forall x Gx\}$ is consistent. For this to the case, we need at least one model were all three sentences are true. In thise case that means having one object in our UD that is F and one that is not F, and we need both of them to be G. This would do the trick.

$$\begin{aligned} \text{UD} &= \{\alpha, \beta\} \\ \text{extension}(F) &= \{\alpha\} \\ \text{extension}(G) &= \{\alpha, \beta\} \end{aligned}$$

We cannot show a set of sentences to be inconsistent this way, because as before, that would mean creating an infinite number of models.

Models to show invalidity

Back on p. 126, we said that this argument would be invalid in QL:

$$\begin{array}{ll} & (Rc \,\&\, K_1c) \,\&\, Tc \\ \therefore & Tc \,\&\, K_2c \end{array}$$

In order to show that it is invalid, we need to show that there is some model in which the premises are true and the conclusion is false. We can construct such a model deliberately. Here is one way to do it:

$$\begin{aligned} \text{UD} &= \{\alpha\} \\ \text{extension}(T) &= \{\alpha\} \\ \text{extension}(K_1) &= \{\alpha\} \\ \text{extension}(K_2) &= \emptyset \\ \text{extension}(R) &= \{\alpha\} \\ \text{referent}(c) &= \alpha \end{aligned}$$

As you have probably guessed, we cannot show an argument valid with models so simply.

Practice Exercises

Part A Show that each of the following is contingent.

⋆ 1. $Da \,\&\, Db$
⋆ 2. $\exists x Txh$
⋆ 3. $Pm \,\&\, {\sim}\forall x Px$
4. $\forall z Jz \leftrightarrow \exists y Jy$
5. $\forall x(Wxmn \vee \exists y Lxy)$
6. $\exists x(Gx \rightarrow \forall y My)$

Part B Show that the following pairs of sentences are not logically equivalent.

1. Ja, Ka
2. $\exists x Jx$, Jm
3. $\forall x Rxx$, $\exists x Rxx$
4. $\exists x Px \rightarrow Qc$, $\exists x(Px \rightarrow Qc)$
5. $\forall x(Px \rightarrow {\sim}Qx)$, $\exists x(Px \,\&\, {\sim}Qx)$
6. $\exists x(Px \,\&\, Qx)$, $\exists x(Px \rightarrow Qx)$
7. $\forall x(Px \rightarrow Qx)$, $\forall x(Px \,\&\, Qx)$
8. $\forall x \exists y Rxy$, $\exists x \forall y Rxy$
9. $\forall x \exists y Rxy$, $\forall x \exists y Ryx$

Part C Show that the following sets of sentences are consistent.

1. $\{\mathrm{Ma}, {\sim}\mathrm{Na}, \mathrm{Pa}, {\sim}\mathrm{Qa}\}$
2. $\{Lee, Lef, {\sim}Lfe, {\sim}Lff\}$
3. $\{{\sim}(Ma \,\&\, \exists x Ax), Ma \vee Fa, \forall x(Fx \rightarrow Ax)\}$
4. $\{Ma \vee Mb, Ma \rightarrow \forall x{\sim}Mx\}$
5. $\{\forall y Gy, \forall x(Gx \rightarrow Hx), \exists y{\sim}Iy\}$
6. $\{\exists x(Bx \vee Ax), \forall x{\sim}Cx, \forall x\big[(Ax \,\&\, Bx) \rightarrow Cx\big]\}$
7. $\{\exists x Xx, \exists x Yx, \forall x(Xx \leftrightarrow {\sim}Yx)\}$
8. $\{\forall x(Px \vee Qx), \exists x{\sim}(Qx \,\&\, Px)\}$
9. $\{\exists z(Nz \,\&\, Ozz), \forall x \forall y(Oxy \rightarrow Oyx)\}$
10. $\{{\sim}\exists x \forall y Rxy, \forall x \exists y Rxy\}$

Part D Construct models to show that the following arguments are invalid.

1. $\forall x(Ax \rightarrow Bx)$, $\therefore\ \exists x Bx$
2. $\forall x(Rx \rightarrow Dx)$, $\forall x(Rx \rightarrow Fx)$, $\therefore\ \exists x(Dx \,\&\, Fx)$
3. $\exists x(Px \rightarrow Qx)$, $\therefore \exists x Px$
4. $Na \,\&\, Nb \,\&\, Nc$, $\therefore\ \forall x Nx$

5. Rde, $\exists x Rxd$, $\therefore$ Red
6. $\exists x(Ex \,\&\, Fx)$, $\exists x Fx \rightarrow \exists x Gx$, $\therefore$ $\exists x(Ex \,\&\, Gx)$
7. $\forall x Oxc$, $\forall x Ocx$, $\therefore$ $\forall x Oxx$
8. $\exists x(Jx \,\&\, Kx)$, $\exists x{\sim}Kx$, $\exists x{\sim}Jx$, $\therefore$ $\exists x({\sim}Jx \,\&\, {\sim}Kx)$
9. $Lab \rightarrow \forall x Lxb$, $\exists x Lxb$, $\therefore$ Lbb

Part E

⋆ 1. Show that $\{{\sim}Raa, \forall x(x = a \vee Rxa)\}$ is consistent.
⋆ 2. Show that $\{\forall x \forall y \forall z(x = y \vee y = z \vee x = z), \exists x \exists y\ x \neq y\}$ is consistent.
⋆ 3. Show that $\{\forall x \forall y\ x = y, \exists x\ x \neq a\}$ is inconsistent.
4. Show that $\exists x(x = h \,\&\, x = i)$ is contingent.
5. Show that $\{\exists x \exists y(Zx \,\&\, Zy \,\&\, x = y), {\sim}Zd, d = s\}$ is consistent.
6. Show that $\forall x(Dx \rightarrow \exists y Tyx)$ $\therefore$ $\exists y \exists z\ y \neq z$ is invalid.

Chapter 7

Proofs in Quantified Logic

7.1 Rules for Quantifiers

For proofs in QL, we use all of the basic rules of SL plus four new basic rules: both introduction and elimination rules for each of the quantifiers.

Since all of the derived rules of SL are derived from the basic rules, they will also hold in QL. We will add another derived rule, a replacement rule called quantifier negation.

Universal elimination

If you have $\forall x A x$, it is legitimate to infer that anything is an A. You can infer Aa, Ab, Az, Ad_3— in short, you can infer $A\mathscr{c}$ for any constant $\mathscr{c}$. This is the general form of the universal elimination rule ($\forall$E):

$$\begin{array}{r|ll} m & \forall \mathscr{x} \mathscr{A} & \\ & \mathscr{A}[\mathscr{c}|\mathscr{x}] & \forall\text{E } m \end{array}$$

$\mathscr{A}[\mathscr{c}|\mathscr{x}]$ is a substitution instance of $\forall \mathscr{x} \mathscr{A}$. The symbols for a substitution instance are not symbols of QL, so you cannot write them in a proof. Instead, you write the subsituted sentence with the constant $\mathscr{c}$ replacing all occurrences of the variable $\mathscr{x}$ in $\mathscr{A}$. For example:

1	$\forall x(Mx \to Rxd)$	
2	$Ma \to Rad$	$\forall$E 1
3	$Md \to Rdd$	$\forall$E 1

Existential introduction

When is it legitimate to infer $\exists xAx$? If you know that something is an A— for instance, if you have Aa available in the proof.

This is the existential introduction rule ($\exists$I):

m	$\mathcal{A}$	
	$\exists\mathcal{x}\mathcal{A}[\mathcal{x}\|\|\mathcal{c}]$	$\exists$I m

It is important to notice that $\mathcal{A}[\mathcal{x}||\mathcal{c}]$ is not the same as a substitution instance. We write it with two bars to show that the variable $\mathcal{x}$ does not need to replace all occurrences of the constant $\mathcal{c}$. You can decide which occurrences to replace and which to leave in place. For example:

1	$Ma \to Rad$	
2	$\exists x(Ma \to Rax)$	$\exists$I 1
3	$\exists x(Mx \to Rxd)$	$\exists$I 1
4	$\exists x(Mx \to Rad)$	$\exists$I 1
5	$\exists y\exists x(Mx \to Ryd)$	$\exists$I 4
6	$\exists z\exists y\exists x(Mx \to Ryz)$	$\exists$I 5

Universal introduction

A universal claim like $\forall xPx$ would be proven if every substitution instance of it had been proven, if every sentence Pa, Pb, ... were available in a proof. Alas, there is no hope of proving *every* substitution instance. That would require proving Pa, Pb, ..., Pj_2, ..., Ps_7, ..., and so on to infinity. There are infinitely many constants in QL, and so this process would never come to an end.

Consider a simple argument: $\forall xMx$, $\therefore$ $\forall yMy$

It makes no difference to the meaning of the sentence whether we use the variable

x or the variable y, so this argument is obviously valid. Suppose we begin in this way:

1	$\forall x Mx$	want $\forall y My$
2	Ma	$\forall$E 1

We have derived Ma. Nothing stops us from using the same justification to derive Mb, …, Mj_2, …, Ms_7, …, and so on until we run out of space or patience. We have effectively shown the way to prove $M\mathscr{c}$ for any constant $\mathscr{c}$. From this, $\forall x Mx$ follows.

1	$\forall x Mx$	
2	Ma	$\forall$E 1
3	$\forall y My$	$\forall$I 2

It is important here that a was just some arbitrary constant. We had not made any special assumptions about it. If Ma were a premise of the argument, then this would not show anything about *all* y. For example:

1	$\forall x Rxa$	
2	Raa	$\forall$E 1
3	$\forall y Ryy$	not allowed!

This is the schematic form of the universal introduction rule ($\forall$I):

m	$\mathscr{A}$	
	$\forall \mathscr{x}\mathscr{A}[\mathscr{x}\vert\mathscr{c}]^*$	$\forall$I m

* $\mathscr{c}$ must not occur in any undischarged assumptions.

Note that we can do this for any constant that does not occur in an undischarged assumption and for any variable.

Note also that the constant may not occur in any *undischarged* assumption, but it may occur as the assumption of a subproof that we have already closed. For example, we can prove $\forall z(Dz \rightarrow Dz)$ without any premises.

1	Df	want Df
2	Df	R 1
3	$Df \to Df$	→I 1–2
4	$\forall z(Dz \to Dz)$	∀I 3

Existential elimination

A sentence with an existential quantifier tells us that there is *some* member of the UD that satisfies a formula. For example, $\exists x Sx$ tells us (roughly) that there is at least one S. It does not tell us *which* member of the UD satisfies S, however. We cannot immediately conclude Sa, Sf_{23}, or any other substitution instance of the sentence. What can we do?

Suppose that we knew both $\exists x Sx$ and $\forall x(Sx \to Tx)$. We could reason in this way:

> Since $\exists x Sx$, there is something that is an S. We do not know which constants refer to this thing, if any do, so call this thing Ω. From $\forall x(Sx \to Tx)$, it follows that if Ω is an S, then it is a T. Therefore Ω is a T. Because Ω is a T, we know that $\exists x Tx$.

In this paragraph, we introduced a name for the thing that is an S. We called it Ω, so that we could reason about it and derive some consequences from there being an S. Since Ω is just a bogus name introduced for the purpose of the proof and not a genuine constant, we could not mention it in the conclusion. Yet we could derive a sentence that does not mention Ω; namely, $\exists x Tx$. This sentence does follow from the two premises.

We want the existential elimination rule to work in a similar way. Yet since Greek letters like Ω are not symbols of QL, we cannot use them in formal proofs. Instead, we will use constants of QL which do not otherwise appear in the proof. A constant that is used to stand in for whatever it is that satisfies an existential claim is called a PROXY. Reasoning with the proxy must all occur inside a subproof, and the proxy cannot be a constant that is doing work elsewhere in the proof.

This is the schematic form of the existential elimination rule (∃E):

$$\begin{array}{r|lll}
m & \exists \mathcal{x}\mathcal{A} & & \\
n & \quad \vert\ \underline{\mathcal{A}[c|\mathcal{x}]^{*}} & & \\
p & \quad \vert\ \mathcal{B} & & \\
 & \mathcal{B} & & \exists\text{E } m,\ n\text{–}p
\end{array}$$

* The constant c must not appear in $\exists \mathcal{x}\mathcal{A}$, in $\mathcal{B}$, or in any undischarged assumption.

With this rule, we can give a formal proof that $\exists xSx$ and $\forall x(Sx \rightarrow Tx)$ together entail $\exists xTx$. The structure of the proof is effectively the same as the English-language argument with which we began, except that the subproof uses the constant "a" rather than the bogus name Ω.

$$\begin{array}{r|ll}
1 & \exists xSx & \\
2 & \underline{\forall x(Sx \rightarrow Tx)} & \text{want } \exists xTx \\
3 & \quad \vert\ \underline{Sa} & \\
4 & \quad \vert\ Sa \rightarrow Ta & \forall\text{E } 2 \\
5 & \quad \vert\ Ta & \rightarrow\text{E } 3,\ 4 \\
6 & \quad \vert\ \exists xTx & \exists\text{I } 5 \\
7 & \exists xTx & \exists\text{E } 1,\ 3\text{–}6
\end{array}$$

Quantifier negation

When translating from English to QL, we noted that $\sim\exists x\sim\mathcal{A}$ is logically equivalent to $\forall x\mathcal{A}$. In QL, they are provably equivalent. We can prove one half of the equivalence with a rather gruesome proof:

$$\begin{array}{rll}
1 & \vert\ \underline{\forall x Ax} & \text{want } {\sim}\exists x{\sim}Ax \\
2 & \vert\ \ \vert\ \underline{\exists{\sim}Ax} & \text{for reductio} \\
3 & \vert\ \ \vert\ \ \vert\ \underline{{\sim}Ac} & \text{for } \exists\text{E} \\
4 & \vert\ \ \vert\ \ \vert\ \ \vert\ \underline{\forall x Ax} & \text{for reductio} \\
5 & \vert\ \ \vert\ \ \vert\ \ \vert\ Ac & \forall\text{E } 1 \\
6 & \vert\ \ \vert\ \ \vert\ \ \vert\ {\sim}Ac & \text{R } 3 \\
7 & \vert\ \ \vert\ \ \vert\ {\sim}\forall x Ax & {\sim}\text{I } 4\text{–}6 \\
8 & \vert\ \ \vert\ \forall x Ax & \text{R } 1 \\
9 & \vert\ \ \vert\ {\sim}\forall x Ax & \exists\text{E } 3\text{–}7 \\
10 & \vert\ {\sim}\exists{\sim}Ax & {\sim}\text{I } 2\text{–}9
\end{array}$$

In order to show that the two sentences are genuinely equivalent, we need a second proof that assumes ${\sim}\exists x{\sim}\mathcal{A}$ and derives $\forall x\mathcal{A}$. We leave that proof as an exercise for the reader.

It will often be useful to translate between quantifiers by adding or subtracting negations in this way, so we add two derived rules for this purpose. These rules are called quantifier negation (QN):

$$\begin{array}{ll}
{\sim}\forall\mathcal{X}\mathcal{A} \Longleftrightarrow \exists\mathcal{X}{\sim}\mathcal{A} & \\
{\sim}\exists\mathcal{X}\mathcal{A} \Longleftrightarrow \forall\mathcal{X}{\sim}\mathcal{A} & \text{QN}
\end{array}$$

Since QN is a replacement rule, it can be used on whole sentences or on subformulae.

Practice Exercises

Part A Provide a justification (rule and line numbers) for each line of proof that requires one.

$$\begin{array}{rl}
1 & \vert\ \forall x\exists y(Rxy \vee Ryx) \\
2 & \vert\ \underline{\forall x{\sim}Rmx} \\
3 & \vert\ \exists y(Rmy \vee Rym) \\
4 & \vert\ \ \vert\ \underline{Rma \vee Ram} \\
5 & \vert\ \ \vert\ {\sim}Rma \\
6 & \vert\ \ \vert\ Ram \\
7 & \vert\ \ \vert\ \exists xRxm \\
8 & \vert\ \exists xRxm
\end{array}$$

$$\begin{array}{r|l}
1 & \forall x(\exists y Lxy \to \forall z Lzx)\\
2 & Lab\\ \hline
3 & \exists y Lay \to \forall z Lza\\
4 & \exists y Lay\\
5 & \forall z Lza\\
6 & Lca\\
7 & \exists y Lcy \to \forall z Lzc\\
8 & \exists y Lcy\\
9 & \forall z Lzc\\
10 & Lcc\\
11 & \forall x Lxx
\end{array}$$

$$\begin{array}{r|l}
1 & \forall x(Jx \to Kx)\\
2 & \exists x \forall y Lxy\\
3 & \forall x Jx\\ \hline
4 & Ja\\
5 & Ja \to Ka\\
6 & Ka\\
7 & \quad \vert\; \underline{\forall y Lay}\\
8 & \quad \vert\; Laa\\
9 & \quad \vert\; Ka \,\&\, Laa\\
10 & \quad \vert\; \exists x(Kx \,\&\, Lxx)\\
11 & \exists x(Kx \,\&\, Lxx)
\end{array}$$

$$\begin{array}{r|l}
1 & \quad \vert\; \underline{\sim(\exists x Mx \vee \forall x{\sim}Mx)}\\
2 & \quad \vert\; \sim\exists x Mx \,\&\, \sim\forall x{\sim}Mx\\
3 & \quad \vert\; \sim\exists x Mx\\
4 & \quad \vert\; \forall x{\sim}Mx\\
5 & \quad \vert\; \sim\forall x{\sim}Mx\\
6 & \exists x Mx \vee \forall x{\sim}Mx
\end{array}$$

Part B

Without using the QN rule, prove $\sim\exists x{\sim}\mathcal{A} \vdash \forall x\mathcal{A}$

Part C Provide a proof of each claim.

1. $\vdash \forall x Fx \vee \sim\forall x Fx$
2. $\{\forall x(Mx \leftrightarrow Nx), Ma \,\&\, \exists x Rxa\} \vdash \exists x Nx$
3. $\{\forall x(\sim Mx \vee Ljx), \forall x(Bx \to Ljx), \forall x(Mx \vee Bx)\} \vdash \forall x Ljx$
4. $\forall x(Cx \,\&\, Dt) \vdash \forall x Cx \,\&\, Dt$
5. $\exists x(Cx \vee Dt) \vdash \exists x Cx \vee Dt$

Part D

In the previous chapter (p. 160), we gave the following example

> The hospital will only hire a skilled surgeon. All surgeons are greedy. Billy is a surgeon, but is not skilled. Therefore, Billy is greedy, but the hospital will not hire him.

UD: people
Gx: x is greedy.
Hx: The hospital will hire x.
Rx: x is a surgeon.
Kx: x is skilled.
b: Billy

$$\begin{array}{ll} & \forall x\big[{\sim}(Rx \,\&\, Kx) \rightarrow {\sim}Hx\big] \\ & \forall x(Rx \rightarrow Gx) \\ & Rb \,\&\, {\sim}Kb \\ \therefore & Gb \,\&\, {\sim}Hb \end{array}$$

Prove the symbolized argument.

Part E In Part C on p. 120 you were asked to translate some of the classical Aristotelian syllogisms. Now you can prove the ones that are listed below. I have omitted Datisi and Ferio because their proofs are trivial variations on Darii and Ferison.

Barbara All Bs are Cs. All As are Bs. ∴ All As are Cs.

Baroco All Cs are Bs. Some A is not B. ∴ Some A is not C.

Bocardo Some B is not C. All Bs are As. ∴ Some A is not C.

Celantes No Bs are Cs. All As are Bs. ∴ No Cs are As.

Celarent No Bs are Cs. All As are Bs. ∴ No As are Cs.

Campestres All Cs are Bs. No As are Bs. ∴ No As are Cs.

Cesare No Cs are Bs. All As are Bs. ∴ No As are Cs.

Dabitis All Bs are Cs. Some A is B. ∴ Some C is A.

Darii All Bs are Cs. Some A is B. ∴ Some A is C.

Disamis Some B is C. All Bs are As. ∴ Some A is C.

Ferison No Bs are Cs. Some B is A. ∴ Some A is not C.

Festino No Cs are Bs. Some A is B. ∴ Some A is not C.

Frisesomorum Some B is C. No As are Bs. ∴ Some C is not A.

Part F Symbolize each of the following argument forms in QL and add the additional assumptions "There is an A" and "There is a B." Then prove that the supplemented arguments forms are valid in QL.

Darapti: All As are Bs. All As are Cs. $\therefore$ Some B is C.

Felapton: No Bs are Cs. All As are Bs. $\therefore$ Some A is not C.

Barbari: All Bs are Cs. All As are Bs. $\therefore$ Some A is C.

Camestros: All Cs are Bs. No As are Bs. $\therefore$ Some A is not C.

Celaront: No Bs are Cs. All As are Bs. $\therefore$ Some A is not C.

Fapesmo: All Bs are Cs. No As are Bs. $\therefore$ Some C is not A.

Baralipton All Bs are Cs. All As are Bs. $\therefore$ Some C is A.

Part G Provide a proof of each claim.

1. $\forall x \forall y Gxy \vdash \exists x Gxx$
2. $\forall x \forall y (Gxy \rightarrow Gyx) \vdash \forall x \forall y (Gxy \leftrightarrow Gyx)$
3. $\{\forall x(Ax \rightarrow Bx), \exists x Ax\} \vdash \exists x Bx$
4. $\{Na \rightarrow \forall x(Mx \leftrightarrow Ma), Ma, {\sim}Mb\} \vdash {\sim}Na$
5. $\vdash \forall z(Pz \vee {\sim}Pz)$
6. $\vdash \forall x Rxx \rightarrow \exists x \exists y Rxy$
7. $\vdash \forall y \exists x(Qy \rightarrow Qx)$

Part H Show that each pair of sentences is provably equivalent.

1. $\forall x(Ax \rightarrow {\sim}Bx)$, ${\sim}\exists x(Ax \,\&\, Bx)$
2. $\forall x({\sim}Ax \rightarrow Bd)$, $\forall x Ax \vee Bd$
3. $\exists x Px \rightarrow Qc$, $\forall x(Px \rightarrow Qc)$

Part I Show that each of the following is provably inconsistent.

1. $\{Sa \rightarrow Tm,\ Tm \rightarrow Sa,\ Tm \,\&\, {\sim}Sa\}$
2. $\{{\sim}\exists x \exists y Lxy,\ Laa\}$
3. $\{\forall x(Px \rightarrow Qx),\ \forall z(Pz \rightarrow Rz),\ \forall y Py,\ {\sim}Qa \,\&\, {\sim}Rb\}$

Part J Write a symbolization key for the following argument, translate it, and prove it:

> There is someone who likes everyone who likes everyone that he likes. Therefore, there is someone who likes himself.

7.2 Rules for Identity

The identity predicate is not part of QL, but we add it when we need to symbolize certain sentences. For proofs involving identity, we add two rules of proof.

Suppose you know that many things that are true of a are also true of b. For example: $Aa \,\&\, Ab$, $Ba \,\&\, Bb$, $\sim Ca \,\&\, \sim Cb$, $Da \,\&\, Db$, $\sim Ea \,\&\, \sim Eb$, and so on. This would not be enough to justify the conclusion $a = b$. (See p. 144.) In general, there are no sentences that do not already contain the identity predicate that could justify the conclusion $a = b$. This means that the identity introduction rule will not justify $a = b$ or any other identity claim containing two different constants.

However, it is always true that $a = a$. In general, no premises are required in order to conclude that something is identical to itself. So this will be the identity introduction rule, abbreviated =I:

$$\begin{array}{l|ll} & \mathscr{c} = \mathscr{c} & \quad =\text{I} \end{array}$$

Notice that the =I rule does not require referring to any prior lines of the proof. For any constant $\mathscr{c}$, you can write $\mathscr{c} = \mathscr{c}$ on any point with only the =I rule as justification.

If you have shown that $a = b$, then anything that is true of a must also be true of b. For any sentence with a in it, you can replace some or all of the occurrences of a with b and produce an equivalent sentence. For example, if you already know Raa, then you are justified in concluding Rab, Rba, Rbb. Recall that $\mathscr{A}[\mathscr{a}||\mathscr{b}]$ is the sentence produced by replacing $\mathscr{a}$ in $\mathscr{A}$ with $\mathscr{b}$. This is not the same as a substitution instance, because $\mathscr{b}$ may replace some or all occurrences of $\mathscr{a}$. The identity elimination rule (=E) justifies replacing terms with other terms that are identical to it:

$$\begin{array}{l|ll} m & \mathscr{a} = \mathscr{b} & \\ n & \mathscr{A} & \\ & \mathscr{A}[\mathscr{a}||\mathscr{b}] & \quad =\text{E } m,\, n \\ & \mathscr{A}[\mathscr{b}||\mathscr{a}] & \quad =\text{E } m,\, n \end{array}$$

To see the rules in action, consider this proof:

1	$\forall x \forall y\ x = y$		
2	$\exists x Bx$		
3	$\forall x(Bx \rightarrow {\sim}Cx)$		want ${\sim}\exists x Cx$
4		Be	
5		$\forall y\ e = y$	$\forall$E 1
6		$e = f$	$\forall$E 5
7		Bf	=E 6, 4
8		$Bf \rightarrow {\sim}Cf$	$\forall$E 3
9		${\sim}Cf$	$\rightarrow$E 8, 7
10	${\sim}Cf$		$\exists$E 2, 4–9
11	$\forall x {\sim}Cx$		$\forall$I 10
12	${\sim}\exists x Cx$		QN 11

Practice Exercises

Part A Provide a proof of each claim.

1. $\{Pa \vee Qb, Qb \rightarrow b = c, {\sim}Pa\} \vdash Qc$
2. $\{m = n \vee n = o, An\} \vdash Am \vee Ao$
3. $\{\forall x x = m, Rma\} \vdash \exists x Rxx$
4. ${\sim}\exists x x \neq m \vdash \forall x \forall y(Px \rightarrow Py)$
5. $\forall x \forall y(Rxy \rightarrow x = y) \vdash Rab \rightarrow Rba$
6. $\{\exists x Jx, \exists x {\sim}Jx\} \vdash \exists x \exists y x \neq y$
7. $\{\exists x Dx, \forall x(x = p \leftrightarrow Dx)\} \vdash Dp$
8. $\{\exists x[(Kx \,\&\, Bx) \,\&\, \forall y(Ky \rightarrow x = y)], Kd\} \vdash Bd$
9. $\vdash Pa \rightarrow \forall x(Px \vee x \neq a)$

Appendix A

Other Symbolic Notation

In the history of formal logic, different symbols have been used at different times and by different authors. Often, authors were forced to use notation that their printers could typeset.

In one sense, the symbols used for various logical constants is arbitrary. There is nothing written in heaven that says that '$\sim$' must be the symbol for truth-functional negation. We might have specified a different symbol to play that part. Once we have given definitions for well-formed formulae (wff) and for truth in our logic languages, however, using '$\sim$' is no longer arbitrary. That is the symbol for negation in this textbook, and so it is the symbol for negation when writing sentences in our languages SL or QL.

This appendix presents some common symbols, so that you can recognize them if you encounter them in an article or in another book.

summary of symbols	
negation	$\neg$, $\sim$
conjunction	&, $\wedge$, $\bullet$
disjunction	$\vee$
conditional	$\rightarrow$, $\supset$
biconditional	$\leftrightarrow$, $\equiv$

Negation Two commonly used symbols are the *hoe*, '$\neg$', and the *swung dash*, '$\sim$.' In some more advanced formal systems it is necessary to distinguish between two kinds of negation; the distinction is sometimes represented by using both '$\neg$' and '$\sim$.'

Disjunction The symbol '$\vee$' is typically used to symbolize inclusive disjunction.

Conjunction Conjunction is often symbolized with the *ampersand*, '&.' The ampersand is actually a decorative form of the Latin word 'et' which means 'and'; it is commonly used in English writing. As a symbol in a formal system, the ampersand is not the word 'and'; its meaning is given by the formal semantics for the language. Perhaps to avoid this confusion, some systems use a different symbol for conjunction. For example, '$\wedge$' is a counterpart to the symbol used for disjunction. Sometimes a single dot, '$\bullet$', is used. In some older texts, there is no symbol for conjunction at all; 'A and B' is simply written 'AB.'

Material Conditional There are two common symbols for the material conditional: the *arrow*, '$\rightarrow$', and the *hook*, '$\supset$.'

Material Biconditional The *double-headed arrow*, '$\leftrightarrow$', is used in systems that use the arrow to represent the material conditional. Systems that use the hook for the conditional typically use the *triple bar*, '$\equiv$', for the biconditional.

Quantifiers The universal quantifier is typically symbolized as an upside-down A, '$\forall$', and the existential quantifier as a backwards E, '$\exists$.' In some texts, there is no separate symbol for the universal quantifier. Instead, the variable is just written in parentheses in front of the formula that it binds. For example, 'all x are P' is written $(x)Px$.

In some systems, the quantifiers are symbolized with larger versions of the symbols used for conjunction and disjunction. Although quantified expressions cannot be translated into expressions without quantifiers, there is a conceptual connection between the universal quantifier and conjunction and between the existential quantifier and disjunction. Consider the sentence $\exists x Px$, for example. It means that *either* the first member of the UD is a P, *or* the second one is, *or* the third one is, ... Such a system uses the symbol '$\bigvee$' instead of '$\exists$.'

Polish notation

This section briefly discusses sentential logic in Polish notation, a system of notation introduced in the late 1920s by the Polish logician Jan Łukasiewicz.

Lower case letters are used as sentence letters. The capital letter N is used for negation. A is used for disjunction, K for conjunction, C for the conditional, E for the biconditional. ('A' is for alternation, another name for logical disjunction. 'E' is for equivalence.)

In Polish notation, a binary connective is written *before* the two sentences that it connects. For example, the sentence $A \& B$ of SL would be written Kab in Polish notation.

notation of SL	Polish notation
$\sim$	N
$\&$	K
$\vee$	A
$\rightarrow$	C
$\leftrightarrow$	E

The sentences $\sim A \rightarrow B$ and $\sim(A \rightarrow B)$ are very different; the main logical operator of the first is the conditional, but the main connective of the second is negation. In SL, we show this by putting parentheses around the conditional in the second sentence. In Polish notation, parentheses are never required. The left-most connective is always the main connective. The first sentence would simply be written $CNab$ and the second $NCab$.

This feature of Polish notation means that it is possible to evaluate sentences simply by working through the symbols from right to left. If you were constructing a truth table for $NKab$, for example, you would first consider the truth-values assigned to b and a, then consider their conjunction, and then negate the result. The general rule for what to evaluate next in SL is not nearly so simple. In SL, the truth table for $\sim(A \,\&\, B)$ requires looking at A and B, then looking in the middle of the sentence at the conjunction, and then at the beginning of the sentence at the negation. Because the order of operations can be specified more mechanically in Polish notation, variants of Polish notation are used as the internal structure for many computer programming languages.

Quick Reference

Characteristic Truth Tables

$\mathcal{A}$	$\sim\mathcal{A}$
T	F
F	T

$\mathcal{A}$	$\mathcal{B}$	$\mathcal{A}\,\&\,\mathcal{B}$	$\mathcal{A}\vee\mathcal{B}$	$\mathcal{A}\rightarrow\mathcal{B}$	$\mathcal{A}\leftrightarrow\mathcal{B}$
T	T	T	T	T	T
T	F	F	T	F	F
F	T	F	T	T	F
F	F	F	F	T	T

Symbolization

Sentential Connectives (chapter 2)

It is not the case that P.	$\sim P$
Either P, or Q.	$(P \vee Q)$
Neither P, nor Q.	$\sim(P \vee Q)$ or $(\sim P \,\&\, \sim Q)$
Both P, and Q.	$(P \,\&\, Q)$
If P, then Q.	$(P \rightarrow Q)$
P only if Q.	$(P \rightarrow Q)$
P if and only if Q.	$(P \leftrightarrow Q)$
Unless P, Q. P unless Q.	$(P \vee Q)$

Predicates (chapter 5)

All Fs are Gs.	$\forall x(Fx \rightarrow Gx)$
Some Fs are Gs.	$\exists x(Fx \,\&\, Gx)$
Not all Fs are Gs.	$\sim\forall x(Fx \rightarrow Gx)$ or $\exists x(Fx \,\&\, \sim Gx)$
No Fs are Gs.	$\forall x(Fx \rightarrow \sim Gx)$ or $\sim\exists x(Fx \,\&\, Gx)$

Identity (section 5.7)

Only j is G.	$\forall x(Gx \leftrightarrow x = j)$
Everything besides j is G.	$\forall x(x \neq j \rightarrow Gx)$
The F is G.	$\exists x(Fx \,\&\, \forall y(Fy \rightarrow x = y) \,\&\, Gx)$

'The F is not G' can be translated two ways:

It is not the case that the F is G. (wide)	$\sim\exists x(Fx \,\&\, \forall y(Fy \rightarrow x = y) \,\&\, Gx)$
The F is non-G. (narrow)	$\exists x(Fx \,\&\, \forall y(Fy \rightarrow x = y) \,\&\, \sim Gx)$

Using identity to symbolize quantities

There are at least _____ Fs.

one $\exists xFx$
two $\exists x_1\exists x_2(Fx_1 \,\&\, Fx_2 \,\&\, x_1 \neq x_2)$
three $\exists x_1\exists x_2\exists x_3(Fx_1 \,\&\, Fx_2 \,\&\, Fx_3 \,\&\, x_1 \neq x_2 \,\&\, x_1 \neq x_3 \,\&\, x_2 \neq x_3)$
four $\exists x_1\exists x_2\exists x_3\exists x_4(Fx_1 \,\&\, Fx_2 \,\&\, Fx_3 \,\&\, Fx_4 \,\&\, x_1 \neq x_2 \,\&\, x_1 \neq x_3 \,\&\, x_1 \neq x_4 \,\&\, x_2 \neq x_3 \,\&\, x_2 \neq x_4 \,\&\, x_3 \neq x_4)$
n $\exists x_1 \cdots \exists x_n(Fx_1 \,\&\, \cdots \,\&\, Fx_n \,\&\, x_1 \neq x_2 \,\&\, \cdots \,\&\, x_{n-1} \neq x_n)$

There are at most _____ Fs.

One way to say 'at most n things are F' is to put a negation sign in front of one of the symbolizations above and say $\sim$'at least $n+1$ things are F.' Equivalently:

one $\forall x_1\forall x_2\big[(Fx_1 \,\&\, Fx_2) \rightarrow x_1 = x_2\big]$
two $\forall x_1\forall x_2\forall x_3\big[(Fx_1 \,\&\, Fx_2 \,\&\, Fx_3) \rightarrow (x_1 = x_2 \vee x_1 = x_3 \vee x_2 = x_3)\big]$
three $\forall x_1\forall x_2\forall x_3\forall x_4\big[(Fx_1 \,\&\, Fx_2 \,\&\, Fx_3 \,\&\, Fx_4) \rightarrow (x_1 = x_2 \vee x_1 = x_3 \vee x_1 = x_4 \vee x_2 = x_3 \vee x_2 = x_4 \vee x_3 = x_4)\big]$
n $\forall x_1 \cdots \forall x_{n+1}\big[(Fx_1 \,\&\, \cdots \,\&\, Fx_{n+1}) \rightarrow (x_1 = x_2 \vee \cdots \vee x_n = x_{n+1})\big]$

There are exactly _____ Fs.

One way to say 'exactly n things are F' is to conjoin two of the symbolizations above and say 'at least n things are F' & 'at most n things are F.' The following equivalent formulae are shorter:

zero $\forall x{\sim}Fx$
one $\exists x\big[Fx \,\&\, {\sim}\exists y(Fy \,\&\, x \neq y)\big]$
two $\exists x_1\exists x_2\big[Fx_1 \,\&\, Fx_2 \,\&\, x_1 \neq x_2 \,\&\, {\sim}\exists y\big(Fy \,\&\, y \neq x_1 \,\&\, y \neq x_2\big)\big]$
three $\exists x_1\exists x_2\exists x_3\big[Fx_1 \,\&\, Fx_2 \,\&\, Fx_3 \,\&\, x_1 \neq x_2 \,\&\, x_1 \neq x_3 \,\&\, x_2 \neq x_3 \,\&\, {\sim}\exists y(Fy \,\&\, y \neq x_1 \,\&\, y \neq x_2 \,\&\, y \neq x_3)\big]$
n $\exists x_1 \cdots \exists x_n\big[Fx_1 \,\&\, \cdots \,\&\, Fx_n \,\&\, x_1 \neq x_2 \,\&\, \cdots \,\&\, x_{n-1} \neq x_n \,\&\, {\sim}\exists y(Fy \,\&\, y \neq x_1 \,\&\, \cdots \,\&\, y \neq x_n)\big]$

Specifying the size of the UD

Removing F from the symbolizations above produces sentences that talk about the size of the UD. For instance, 'there are at least 2 things (in the UD)' may be symbolized as $\exists x\exists y(x \neq y)$.

Basic Rules of Proof

Reiteration

$$\begin{array}{r|ll} m & \mathcal{A} & \\ & \mathcal{A} & \text{R } m \end{array}$$

Conjunction Introduction

$$\begin{array}{r|ll} m & \mathcal{A} & \\ n & \mathcal{B} & \\ & \mathcal{A} \,\&\, \mathcal{B} & \&\text{I } m, n \end{array}$$

$$\begin{array}{r|ll} m & \mathcal{A} & \\ n & \mathcal{B} & \\ & \mathcal{B} \,\&\, \mathcal{A} & \&\text{I } m, n \end{array}$$

Conjunction Elimination

$$\begin{array}{r|ll} m & \mathcal{A} \,\&\, \mathcal{B} & \\ & \mathcal{A} & \&\text{E } m \end{array}$$

$$\begin{array}{r|ll} m & \mathcal{A} \,\&\, \mathcal{B} & \\ & \mathcal{B} & \&\text{E } m \end{array}$$

Disjunction Introduction

$$\begin{array}{r|ll} m & \mathcal{A} & \\ & \mathcal{A} \vee \mathcal{B} & \vee\text{I } m \end{array}$$

$$\begin{array}{r|ll} m & \mathcal{A} & \\ & \mathcal{B} \vee \mathcal{A} & \vee\text{I } m \end{array}$$

Disjunction Elimination

$$\begin{array}{r|ll} m & \mathcal{A} \vee \mathcal{B} & \\ n & \sim\mathcal{B} & \\ & \mathcal{A} & \vee\text{E } m, n \end{array}$$

$$\begin{array}{r|ll} m & \mathcal{A} \vee \mathcal{B} & \\ n & \sim\mathcal{A} & \\ & \mathcal{B} & \vee\text{E } m, n \end{array}$$

Conditional Introduction

$$\begin{array}{r|l|ll} m & & \mathcal{A} & \text{want } \mathcal{B} \\ \cline{3-3} n & & \mathcal{B} & \\ & \multicolumn{2}{l}{\mathcal{A} \to \mathcal{B}} & \to\text{I } m\text{–}n \end{array}$$

Conditional Elimination

$$\begin{array}{r|ll} m & \mathcal{A} \to \mathcal{B} & \\ n & \mathcal{A} & \\ & \mathcal{B} & \to\text{E } m, n \end{array}$$

Biconditional Introduction

$$\begin{array}{r|l|ll} m & & \mathcal{A} & \text{want } \mathcal{B} \\ \cline{3-3} n & & \mathcal{B} & \\ p & & \mathcal{B} & \text{want } \mathcal{A} \\ \cline{3-3} q & & \mathcal{A} & \\ & \multicolumn{2}{l}{\mathcal{A} \leftrightarrow \mathcal{B}} & \leftrightarrow\text{I } m\text{–}n,\ p\text{–}q \end{array}$$

Biconditional Elimination

$$\begin{array}{r|ll} m & \mathcal{A} \leftrightarrow \mathcal{B} & \\ n & \mathcal{B} & \\ & \mathcal{A} & \leftrightarrow\text{E } m, n \end{array}$$

$$\begin{array}{r|ll} m & \mathcal{A} \leftrightarrow \mathcal{B} & \\ n & \mathcal{A} & \\ & \mathcal{B} & \leftrightarrow\text{E } m, n \end{array}$$

Negation Introduction

$$\begin{array}{r|l|ll} m & & \mathcal{A} & \text{for reductio} \\ \cline{3-3} n-1 & & \mathcal{B} & \\ n & & \sim\mathcal{B} & \\ & \multicolumn{2}{l}{\sim\mathcal{A}} & \sim\text{I } m\text{–}n \end{array}$$

Negation Elimination

$$\begin{array}{r|l|ll} m & & \sim\mathcal{A} & \text{for reductio} \\ \cline{3-3} n-1 & & \mathcal{B} & \\ n & & \sim\mathcal{B} & \\ & \multicolumn{2}{l}{\mathcal{A}} & \sim\text{E } m\text{–}n \end{array}$$

Quantifier Rules

Existential Introduction

$$\begin{array}{r|ll} m & \mathcal{A} & \\ & \exists\mathscr{x}\mathcal{A}[\mathscr{x}||\mathscr{c}] & \exists\text{I } m \end{array}$$

$\mathscr{x}$ may replace some or all occurrences of c in $\mathcal{A}$.

Existential Elimination

$$\begin{array}{r|ll} m & \exists\mathscr{x}\mathcal{A} & \\ n & \quad\mid\underline{\mathcal{A}[\mathscr{c}|\mathscr{x}]} & \\ p & \quad\mid\mathcal{B} & \\ & \mathcal{B} & \exists\text{E } m,\ n\text{–}p \end{array}$$

The constant $\mathscr{c}$ must not appear in $\exists\mathscr{x}\mathcal{A}$, in $\mathcal{B}$, or in any undischarged assumption.

Universal Introduction

$$\begin{array}{r|ll} m & \mathcal{A} & \\ & \forall\mathscr{x}\mathcal{A}[\mathscr{x}|\mathscr{c}] & \forall\text{I } m \end{array}$$

$\mathscr{c}$ must not occur in any undischarged assumptions.

Universal Elimination

$$\begin{array}{r|ll} m & \forall\mathscr{x}\mathcal{A} & \\ & \mathcal{A}[\mathscr{c}|\mathscr{x}] & \forall\text{E } m \end{array}$$

Identity Rules

$$\begin{array}{r|ll} & \mathscr{c} = \mathscr{c} & =\text{I} \end{array}$$

$$\begin{array}{r|ll} m & \mathscr{c} = \mathscr{d} & \\ n & \mathcal{A} & \\ & \mathcal{A}[\mathscr{c}||\mathscr{d}] & =\text{E } m,\ n \\ & \mathcal{A}[\mathscr{d}||\mathscr{c}] & =\text{E } m,\ n \end{array}$$

One constant may replace some or all occurrences of the other.

Derived Rules

Dilemma

$$\begin{array}{r|ll} m & \mathcal{A} \vee \mathcal{B} & \\ n & \mathcal{A} \to \mathcal{C} & \\ p & \mathcal{B} \to \mathcal{C} & \\ & \mathcal{C} & \vee{*}\ m,\ n,\ p \end{array}$$

Modus Tollens

$$\begin{array}{r|ll} m & \mathcal{A} \to \mathcal{B} & \\ n & {\sim}\mathcal{B} & \\ & {\sim}\mathcal{A} & \text{MT } m,\ n \end{array}$$

Hypothetical Syllogism

$$\begin{array}{r|ll} m & \mathcal{A} \to \mathcal{B} & \\ n & \mathcal{B} \to \mathcal{C} & \\ & \mathcal{A} \to \mathcal{C} & \text{HS } m,\ n \end{array}$$

Replacement Rules

Commutivity (Comm)
$(\mathcal{A} \,\&\, \mathcal{B}) \iff (\mathcal{B} \,\&\, \mathcal{A})$
$(\mathcal{A} \vee \mathcal{B}) \iff (\mathcal{B} \vee \mathcal{A})$
$(\mathcal{A} \leftrightarrow \mathcal{B}) \iff (\mathcal{B} \leftrightarrow \mathcal{A})$

DeMorgan (DeM)
${\sim}(\mathcal{A} \vee \mathcal{B}) \iff ({\sim}\mathcal{A} \,\&\, {\sim}\mathcal{B})$
${\sim}(\mathcal{A} \,\&\, \mathcal{B}) \iff ({\sim}\mathcal{A} \vee {\sim}\mathcal{B})$

Double Negation (DN)
${\sim}{\sim}\mathcal{A} \iff \mathcal{A}$

Material Conditional (MC)
$(\mathcal{A} \to \mathcal{B}) \iff ({\sim}\mathcal{A} \vee \mathcal{B})$
$(\mathcal{A} \vee \mathcal{B}) \iff ({\sim}\mathcal{A} \to \mathcal{B})$

Biconditional Exchange ($\leftrightarrow$ex)
$[(\mathcal{A} \to \mathcal{B}) \,\&\, (\mathcal{B} \to \mathcal{A})] \iff (\mathcal{A} \leftrightarrow \mathcal{B})$

Quantifier Negation (QN)
${\sim}\forall\mathscr{x}\mathcal{A} \iff \exists\mathscr{x}{\sim}\mathcal{A}$
${\sim}\exists\mathscr{x}\mathcal{A} \iff \forall\mathscr{x}{\sim}\mathcal{A}$

In the Introduction to his volume *Symbolic Logic*, Charles Lutwidge Dodson advised: "When you come to any passage you don't understand, *read it again*: if you *still* don't understand it, *read it again*: if you fail, even after *three* readings, very likely your brain is getting a little tired. In that case, put the book away, and take to other occupations, and next day, when you come to it fresh, you will very likely find that it is *quite* easy."

The same might be said for this volume, although the reader is forgiven if they take a break for snacks after *two* readings.

about the author:

P.D. Magnus is an assistant professor of philosophy in Albany, New York. His primary research is in the philosophy of science, concerned especially with the underdetermination of theory by data.