Contents

Part II

Quantificational Logic: Introduction

Chapter 1

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.

1.1 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. Therefore, 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 $\neg N$. Why not? $\neg N$ 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:

1. $L \rightarrow (N \vee E)$
2. $E \rightarrow B$
3. L

$\therefore\ \neg B \rightarrow N$

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. Therefore, 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:

1. L
2. 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.

1.2 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, "∃" 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 $a, b, c, \dots w, a_1, f_{32}, j_{390}$, 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 $Ad \rightarrow (Ag \wedge Am)$.

Sentence 3 can be translated as $T_1mg \wedge T_2mg$.

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 tall or taller than Gregor." We can translate it as $\neg T_1dg$.

Sentence 5 requires that we pay careful attention to the order of terms in the key. It becomes $Bdgm$.

1.3 Quantifiers

We are now ready to introduce quantifiers. Consider the symbolization key on page 6 with Donald, Gregor and Marybeth and the sample sentences that came with it. Let's add these sentences to that list:

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 $Hd \wedge Hg \wedge Hm$. 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 $\forall x T_2 xd$.

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 $\forall x_5 Hx_5$.

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: $\neg\exists x Ax$. 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: $\forall x \neg Ax$. 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 $\neg\exists x \neg\mathcal{A}$. 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 difference 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 $\exists x \neg Hx$. Equivalently, we could write $\neg\forall x Hx$.

Sentence 11 is most naturally translated as $\neg\forall x Hx$. This is logically equivalent to sentence 10 and so could also be translated as $\exists x \neg Hx$.

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 "$\neg\forall x\neg$." 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, $\forall xHx$ 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 $\forall xPx$ 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 $\forall xPx$ 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 $\neg Ac$.

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 $\neg Ac$ 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. 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.

1.4 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 $\forall x(Px \wedge Qx)$ 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 $\exists x(Tx \wedge Dx)$.

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 $\neg\mathcal{A} \vee \mathcal{B}$, and this will also hold in QL. So $\exists x(Tx \rightarrow Dx)$ is true if there is some a such that $(\neg Ta \vee Da)$; 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 $\neg\forall x(Tx \rightarrow Dx)$. 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 $\exists x(Tx \wedge \neg Dx)$. Although it is probably not obvious, these two translations are logically equivalent. (This is due to the logical equivalence between $\neg\forall x\mathcal{A}$ and $\exists x\neg\mathcal{A}$, along with the equivalence between $\neg(\mathcal{A} \rightarrow \mathcal{B})$ and $\mathcal{A} \wedge \neg\mathcal{B}$.)

Sentence 17 can be paraphrased as, "It is not the case that there is some dime in my pocket." This can be translated as $\neg\exists x(Px \wedge Dx)$. It might also be paraphrased as, "Everything in my pocket is a non-dime," and then could be translated as $\forall x(Px \rightarrow \neg Dx)$. Again the two translations are logically equivalent. Both are correct translations of sentence 17.

We can now translate the argument from p. 3, 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:

1. Lw
2. $\forall x(Lx \rightarrow Fx)$

$\therefore$ 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 $\forall x(Mx \to Sx)$.

Sentence 19 becomes $\exists x(Mx \wedge Sx)$.

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 $\forall x(Mx \to Sx)$ to be true even though the sentence $\exists x(Mx \wedge Sx)$ 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 $\neg\exists\neg\mathcal{A}$. 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 $\forall x(Mx \to Sx)$ would be true and $\exists x(Mx \wedge Sx)$ 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.

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 $\forall x(Rx \to Mx)$ 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 $\forall x(Rx \to Mx)$ 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 $\forall x(Rx \to Mx)$ 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.

- ▷ 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.

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: $\exists x(Rx \wedge Hex)$.

It is tempting to translate sentence 22 as $\forall xCxe$. 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."

1.5 Recursive Syntax for QL

Up to this point, we have been working informally while translating things in to QL. Now we will introduce a more rigorous syntax for the language we are using. As in section 4 of chapter 2, we will be using recursive definitions. This time, however, we will need to define two kinds of grammatical units. In addition to defining a sentence in QL, we will need to define something called a *well-formed formula* or WFF. A well-formed formula will be like a sentence, but it will have some elements missing, and it won't have a truth value until those elements are provided.

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	$\neg, \wedge, \vee, \rightarrow, \leftrightarrow$
parentheses	(,)
quantifiers	$\forall, \exists$

We define an EXPRESSION OF QL as any string of symbols of QL. 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.

At this point in Chapter **??**, we gave a recursive definition for a sentence in SL (p.**??**). We said that the atomic sentences were sentences of SL, and then gave rules for building new sentences in SL. We are going to do something similar here, except this time we will just be defining a well-formed formula (wff) in QL. Every atomic formula—every formula like Fx or Gab—is a wff. And you can build new wffs by applying the sentential connectives, just as before

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 $\neg\mathcal{A}$ is a wff.
3. If $\mathcal{A}$ and $\mathcal{B}$ are wffs, then $(\mathcal{A} \wedge \mathcal{B})$, 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 $\forall x Ax$ is a wff, but so are $\forall y Ay$, $\forall z Az$, $\forall x_4 Ax_4$, and $\forall z_9 Az_9$.

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 expression that satisfied the recursive syntax was a sentence. You could tell, because we could decide whether expression was true or false. This will not be the case for wffs in QL. Consider the following symbolization key:

UD: people.
Lxy: x loves y.
b: Boris.

Consider the expression Lzz. It is an atomic formula: 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, $\exists z Lzz$ would mean that someone loves themselves.

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 themselves, then you need to write the quantifier: $\forall z Lzz$

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 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 occurrence of a variable that is not bound.

For example, consider the wff $\forall x(Ex \vee Dy) \to \exists z(Ex \to Lzx)$. 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 $(Ex \to Lzx)$, so the occurrence of z is 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. 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: $Aa \to Ba$, $Af \to Bf$, and $Ak \to Bk$ are all substitution instances of $\forall x(Ax \to Bx)$; 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.

1.6 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:

$$\forall x\big[\neg(Rx \wedge Kx) \rightarrow \neg Hx\big]$$
$$\forall x(Rx \rightarrow Gx)$$
$$Rb \wedge \neg Kb$$
$$\therefore\ Gb \wedge \neg Hb$$

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:

$(Rc \wedge Kc) \wedge Tc$
$\therefore\ Tc \wedge Kc$

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:

$(Rc \wedge K_1c) \wedge Tc$
$\therefore\ Tc \wedge K_2c$

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 $\exists x(Dx \wedge Ogx)$.

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 $\exists y \exists x(Dx \wedge Oyx)$.

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 $\forall x\big[Fxg \rightarrow \exists z(Dz \wedge Oxz)\big]$.

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

$$\forall x\big[x \text{ is a dog owner} \rightarrow \exists y(y \text{ is a dog owner} \wedge Fxy\big].$$

Completing the translation, sentence 28 becomes

$$\forall x\big[\exists z(Dz \wedge Oxz) \rightarrow \exists y(\exists z(Dz \wedge Oyz) \wedge Fxy\big].$$

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: $\exists x Gx \rightarrow Gl$.

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 $\exists x Gx \rightarrow Gl$, 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, $\exists x Gx \rightarrow Gl$ 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, $\exists x Gx \wedge Gl$ means the same thing as $\exists x(Gx \wedge Gl)$, and $Gl \rightarrow \exists x Gx$ means the same things as $\exists x(Gl \rightarrow Gx)$.