Contents

Chapter 7

First-Order Logic

This chapter introduces the formal language called $\mathcal{L}^{\text{FOL}}$— pronounced, *the language of first-order logic*— which includes quantifiers like 'for all' ($\forall$) and 'there is' ($\exists$). In contrast to $\mathcal{L}^{\text{PL}}$ whose primitive symbols were limited to sentence letters and sentential operators, $\mathcal{L}^{\text{FOL}}$ includes *predicates* and *singular terms* among its primitive symbols. Whereas one-place predicates are terms for properties, two-place predicates are terms for relations, and n-place predicates are similar but relate n objects. Singular terms include *constants* which name objects and *variables* which may refer to any object. First-order logic is *first-order* because the quantifiers bind variables that are singular term (type e), and so range over a domain of objects rather than properties and relations as in *second-order logic*, or the higher-order entities of *higher-order logic* (e.g., the lambda calculus and simple type theories). For instance, $\mathcal{L}^{\text{FOL}}$ has the expressive power to quantify over people, animals, or things, but not over their various properties such as *being blue*, or relations like *being taller than*, etc.

Despite its limitations, first-order logic is extremely useful, nicely balancing the theoretical virtues of simplicity and intuitive appeal on the one hand with expressive power and logical strength on the other hand. Mastering first-order logic is also an important stepping stone to studying logics for more expressive languages. Although $\mathcal{L}^{\text{FOL}}$ will introduce a number of new expressive resources, all of the primitive symbols which we included before will be present in $\mathcal{L}^{\text{FOL}}$, and so $\mathcal{L}^{\text{FOL}}$ is said to be an *extension* of $\mathcal{L}^{\text{PL}}$. We will also be repeating the same methodology carried out before, and so much should seem familiar. Whereas this chapter develops the syntax for first-order logic by presenting $\mathcal{L}^{\text{FOL}}$ and using this language to regiment sentences and arguments in English, Chapter 8 will provide a semantics and definition of logical consequence for $\mathcal{L}^{\text{FOL}}$. Chapter 9 will repeat these two steps for a language with even more expressive power by including a primitive symbol for identity along with a stock of function symbols. We will then turn to present a proof system for first-order logic called FOL in Chapter 10 before establishing the soundness and completeness of FOL in Chapter 11 and Chapter 12, respectively. As is the case with the syntax and semantics for FOL, the proof system FOL itself will extend PL from before, and so the proofs of the soundness and completeness of FOL will similarly extend the work that we have already done.

7.1 Expressive Limitations

Consider the following argument:

A1. Every human is mortal.
A2. Socrates is human.
A3. Socrates is mortal.

In order to regiment this argument in $\mathcal{L}^{\text{PL}}$, we will need a symbolization key.

E: Every human is mortal.
H: Socrates is human.
M: Socrates is mortal.

Notice that there is no way to break down the sentences above into smaller parts. However, consider the resulting regimentation of the argument in $\mathcal{L}^{\text{PL}}$:

B1. E
B2. H
B3. M

It is easy to show that this argument is invalid. However, the argument in English is intuitively valid. Notice that although the English sentences given above include the same terms, this is not captured by the regimentation in $\mathcal{L}^{\text{PL}}$. Moreover, there is no better regimentation in $\mathcal{L}^{\text{PL}}$.

Here's another common case:

C1. All humans are mammals.
C2. All mammals are multi-celled organisms.
C3. All humans are multi-celled organisms.

This argument is intuitively valid since the conclusion follows from the premises in virtue of the logical form of the argument. However, were we to regiment this argument in $\mathcal{L}^{\text{PL}}$, the best we could do would look something like the following:

H: All humans are mammals.
M: All mammals are multi-celled organisms.
O: All humans are multi-celled organisms.

Again, there is no way to further decompose the sentences given above into smaller parts. However, the resulting argument in $\mathcal{L}^{\text{PL}}$ is invalid for the same reason:

D1. H
D2. $\underline{M\quad}$
D3. O

Since there is nothing wrong with our regimentation when restricted to $\mathcal{L}^{\text{PL}}$, we may take issue with the language $\mathcal{L}^{\text{PL}}$ itself, concluding that we have simply reached the limit of what can be captured with the expressive resources which $\mathcal{L}^{\text{PL}}$ provides. Instead of carving arguments up into *sentences*, we have to look deeper into the sentences themselves in order to capture the validity of the English arguments considered above. In particular, we need to be able to regiment predicates like 'is human', 'is mortal', 'is a mammal', etc., as well as the quantifiers 'every' and 'all', as well as names like 'Socrates'. More generally, we should like to provide a language which can regiment any such argument at this level of logical resolution power. Providing the expressive resources to do so will be the ambition of this chapter.

As we will see, $\mathcal{L}^{\text{FOL}}$ is extremely powerful. Indeed, most of mathematics is developed in a first-order language. Understanding this language and its limits will prove to be an invaluable resource, shedding light on a wide range of theoretical applications as well as extending your powers of logical thinking in day-to-day reasoning. The development of first-order logics and their applications constitute some of the most important and influential theories that human beings have developed so far. Nevertheless, modern logic is still at its very beginning.

7.2 Primitive Expressions in $\mathcal{L}^{\text{fol}}$

Our first key idea will be to include *predicates* in $\mathcal{L}^{\text{FOL}}$. An easy way to think about predicates is as terms for properties: the predicate 'is red' stands for the property of being red. We use predicates to ascribe properties to objects. On their own, predicates are neither true nor false, though they are meaningful nevertheless. For instance, English speakers will know what the predicate 'is red' means and how to use it to make claims about the world. Nevertheless, it doesn't make sense to assign a truth-value to 'is red', 'is loved by John', 'is careful and quite', 'is between Sam and Mary in birth order', or any other predicate, however complex. Rather predicates are used to ascribe properties and relations to objects where those objects may or may not have the properties and relations which are ascribed to them.

In order to refer to objects, we will need another basic type of expression for *names*. For instance, the name 'Christoph' refers to a particular object, in this case a person. Accordingly, we may construct the sentence 'Christoph is Italian', where this may be true or false depending on whether the object that 'Christoph' names has the property that the predicate 'is Italian' expresses. Although names like 'Christoph' are meaningful on their own, they do not have truth-values any more than predicates have truth-values. Rather, it is by combining predicates with singular terms that we will construct *atomic sentences* which do have truth-values. Given the same sentential operators included in $\mathcal{L}^{\text{PL}}$, we may construct complex sentences which inherit their truth-values from their parts. Despite introducing a range of new symbols to the language, $\mathcal{L}^{\text{FOL}}$ will maintain the convention that only sentences have truth-values.

Given that we can construct sentences with nothing more than names and predicates, you might be wondering where the quantifiers fit in. After all, there is no grammatical way to append 'every', 'all', or 'some' to sentences like 'Christoph is Italian'. What is missing are the *variables*. In order to bring this out, consider the following version of H from before:

H': Everything is such that if it is human, then it is a mammal.

However stilted, this claim is perfectly intelligible and express exactly what H expressed before. Three observations are in order. Whereas H includes the plural terms 'humans' and 'mammals', we now have the singular predicates 'is human' and 'is a mammal'. For instance, whereas we may combine 'is human' with a name like 'Christoph' in order to produce the perfectly intelligible declarative sentence 'Christoph is human', the same cannot be said for the plural term 'humans' since 'Christoph humans' is nonsense.

Second, we may consider the role that 'it' plays in H'. Whereas a name like 'Christoph' refers to an object, the term 'it' also appears to indicate an object, but not particular object. Rather, 'it' is playing the role of a variable. Since both constants and variables refer to objects, we will refer to both as *singular terms*. Accordingly, 'it' may be combined with a predicate to form a sentence. In particular, we may combine 'it' with the predicates above to produce 'it is human' and 'it is a mammal'. On their own, 'it is human' and 'it is a mammal' are both atomic sentences though they include the *free variable* 'it', and so are *open sentences*. Nevertheless, these sentences may be combined with a conditional operator to form 'if it is human, then it is mammal' which is also an open sentence but not atomic. Appending 'Everything is such that' to the conditional sentence 'if it is human, then it is mammal' produces H'. Instead of speaking about a particular, H' says of each object that *it* is a mammal if *it* is human. For each object, both occurrences of 'it' refer to that object, at least until we move on to the next object since we are making a general claim about everything. Put otherwise, both occurrences of 'it' are *bound* by the quantifier 'Everything is such that'.

Third, we may observe that the perfectly natural occurrence of 'All' in H has been replaced with the somewhat cumbersome 'Everything is such that'. Whereas 'All' is a *generalized quantifier*, 'Everything is such that' is an attempt to express the first-order *universal quantifier* $\forall$ with the resources of English. These quantifiers differ in a number of important respects. In particular, generalized quantifiers like 'every', 'some', 'all', 'most', 'many', etc., combine with two descriptive terms in order to produce a grammatical sentence. By contrast, we may observe that 'Everything is such that' combines with the open sentence 'if it is human, then it is mammal' to produce H'. Although 'it' is unbound in 'if it is human, then it is mammal' when considered on its own, both occurrences 'it' are bound by 'Everything is such that' in H'. Thus H' is a *closed sentence* since it does not include any unbound variables.

Although first-order quantifiers can be approximated within English, generalized quantifiers are often more natural to use in English. Despite this disparity, generalized quantifiers are much more complicated than first-order quantifiers, making them not as useful for systematic theorizing. Nevertheless, we may use first-order quantifiers to regiment claims in English which are expressed with generalized quantifiers without too much trouble. For these reasons, we will not include formal analogues of generalized quantifiers in $\mathcal{L}^{\text{FOL}}$.

Given this overview of the new expressive resources that we will include in $\mathcal{L}^{\text{FOL}}$, the following sections will provide a much more details introduction to each of the elements that we have touched on above. We will begin again with names and variables.

7.2.1 Singular Terms

In English, a proper name is a term that refers to a person, place, or thing. For instance, 'Christoph' refers to Christoph. In $\mathcal{L}^{\text{FOL}}$, we will take lower-case letters 'a', 'b', 'c', . . . , 's', 't' subscripted by a numeral to be the CONSTANTS of $\mathcal{L}^{\text{FOL}}$. For convenience, we will drop the subscript '$_0$' from constants, using subscripts only as needed. Constants are singular terms since they refer to an object and will be used to regiment names in English.

In English, the same name may refer to different objects in different contexts. Whereas 'Willard' might pick out different people in different contexts who have the same name, we will avoid this ambiguity in $\mathcal{L}^{\text{FOL}}$ by insisting that constants refer to at most one individual on any given interpretation. It is for this reason that we have included infinitely many constants (with the help of the subscripts) so that we don't run out of names.

There are other ways to refer to individuals besides using names. In particular, we often use descriptions. Whereas it is indeterminate to say 'the person standing next to the podium' if more than one person is standing next to the podium or nobody is, this expression succeeds in picking out a unique individual if there is just one person standing next to the podium. Or to take another case, we may use 'the tallest person in the room' to refer to a unique individual assuming that there are at least some people in the room and that one of them is taller than all the others. Such terms are referred to as DEFINITE DESCRIPTIONS. Although the definite descriptions that succeed in referring to unique objects are singular terms, we will not need to include primitive terms in our language to express definite descriptions. Instead, we will construct definite descriptions with the resources of $\mathcal{L}^{\text{FOL}}$.

Another important type of singular term are the VARIABLES that we mentioned above. In English, terms such as 'it', 'her', etc., often play this role. In $\mathcal{L}^{\text{FOL}}$, we will use the lower-case letters 'u', 'v', 'w', 'x', 'y', 'z', from the end of the alphabet with optional subscripts as variables. We will use variables in combination with the quantifiers in order to make claims about every object, or about some object, specifying what properties those objects have. However, in order to make claims about objects, we will need more than singular terms.

7.2.2 Predicates

A one-place predicate is used to express a property of individuals. For instance, 'is hungry' is a one-place predicate which, in combination with a singular term, forms an atomic sentence such as 'Kate is hungry' and 'She is hungry'. In claiming that a predicate is one-place, we are indicating that just one singular term is needed to construct an atomic sentence. This corresponds to the idea that the predicate expresses a property that a single object may have or fail to have. For instance, either Kate is hungry or she isn't. Accordingly, the sentence 'Kate is hungry' is true or false, but the truth-value of this sentence only depends on Kate and whether she is hungry or not, and so no other singular terms need to be provided.

A two-place predicate is used to express a relation between individuals. For instance, the two-place predicate 'is taller than' expresses a height relation between objects. Given a two-place predicate together with two singular terms, we may make a claim which is either true or false. For instance, 'Sam is taller than John' is true just in case Sam is taller than John. It is worth noting that it is possible to use a two-place predicate together with two instances of the same singular term. Just as we may say that 'Jay loves Cary', we may say, 'Jay loves Jay', or somewhat more naturally, 'Jay loves himself'. Although there is just one object at issue in such cases, the predicate 'loves' is two-place all the same.

The singular terms that a predicate requires to form a formula are referred to as the arguments of that predicate. Whereas one-place predicates require one argument and two-place predicates require two arguments, in general we may consider n-place predicates which require n arguments. Although it is easy to generalize, it is uncommon to consider more than three-place predicates in English. For instance, we may say that Sue is between Henry and Tom in birth order, where 'between' has three arguments. Even if it is uncommon to do so in English, nothing stops us from including predicates with more than three arguments in $\mathcal{L}^{\text{FOL}}$. It is convenient to refer to the number of arguments that a predicate has as the adicity (also called arity) of that predicate, where this comes from the convention of referring to one-place predicates as monadic, two-place predicates as dyadic, three-place predicates as triadic, an predicates with more than one place as polyadic. We will also include 0-place predicates which we will refer to as sentence letters and are familiar from before.

In order to regiment n-place predicates in $\mathcal{L}^{\text{FOL}}$, we will use capital letters A^n through Z^n where the superscripts indicate the arity of the predicate. As above, we will drop the superscript '0' so that $A_i, B_i, C_i, \ldots$ and the like are the same sentence letters from before. We will also drop superscripts when the arity is indicated by the number of arguments. For instance, it is clear that 'A' is a tryadic predicate in '$Axcy$' and a dyadic predicate in 'Aaz', though we would do well to use different letters if we needed to consider both at once. Symbolization keys may also rely on the number of arguments to indicate the arity and interpretation as follows:

Ax:	x is angry.	T_1xy:	x is at least as tall as y.
Hx:	x is happy.	T_2xy:	x is at least as tough as y.
Sxy:	x is shorter than y.	$Bxyz$:	y is between x and z.

It is worth emphasising that the variables above do no further work than to specify the order of the arguments in a given predicate. Since monadic predicates only have one argument, the variable merely indicates its adicity. By contrast, the order of the variables also matters for dyadic predicates like T_1. For instance, we could have introduced the predicate:

T_3xy: y is at least as tall as x.

Having reversed the order of the variables above, 'T_3' expresses the converse of the relation expressed by 'T_1'. This makes a big difference. However, it would not have mattered if we were to have replaced 'x' with some other variable such as 'z' or 'y_{13}'.

In order to put these predicates to work, we may now turn to combine the primitive symbols that we have included in $\mathcal{L}^{\text{FOL}}$ in order to produce atomic sentences.

7.2.3 Atomic Sentences

Consider the following sentences:

E1. Cordelia is angry.
E2. Cordelia is shorter than Hamlet.
E3. If Cordelia is angry, then so are Hamlet and Macbeth.
E4. Macbeth is at least as tall and tough as Hamlet.

We have already provided the predicates that we will need to regiment the sentences above, but we have not introduced any constants. Consider the following:

c:	Cordelia	Ax:	x is angry.
h:	Hamlet	Sxy:	x is shorter than y.
m:	Macbeth	T_1xy:	x is at least as tall as y.
		T_2xy:	x is at least as tough as y.

Given these resources, we may regiment sentence E1 as: Ac. Since the 'x' in the symbolization key entry for 'Ax' is just a placeholder, we have replaced the variable with the constant which names Cordelia, resulting in an atomic sentence. Whereas 'Ac' includes the monadic predicate 'A', atomic sentences may include predicates of any adicity. More generally, an ATOMIC SENTENCE of $\mathcal{L}^{\text{FOL}}$ is an n-place predicate followed by n constants where n is any natural number. Accordingly, 0-place predicates are atomic sentences of $\mathcal{L}^{\text{FOL}}$ which, as above, we will refer to as sentence letters. Or to take another example from above, the sentence E2 can be regimented with the dyadic predicate 'S' as: Sch.

By contrast with sentences E1 and E2, sentence E3 is not atomic on account of including sentential operators. Nevertheless, we may regiment this sentence by identifying its atomic parts: Ac, Ah, and Am. These atomic sentences will play a similar role to the sentence letters included in $\mathcal{L}^{\text{PL}}$. In particular, we may regiment sentence E3 by: $Ad \rightarrow (Ag \wedge Am)$.

Sentence E4 is similar though the atomic sentences are a little trickier to identify. We begin by observing that two things are said of Marybeth and Gregor: first, Marybeth is at least as tall as Gregor; and second, Marybeth is at least as tough as Gregor. Thus we can build the atomic sentences: T_1mg and T_2mg. The last step is to put these together: $T_1mg \wedge T_2mg$.

7.2.4 Open Sentences

Consider the following sentence:

F1. Either it is crimson or ruby since she wants to impress him.

In contemplating the predicates that we will need, we may introduce the following:

Cx:	x is crimson.
Rx:	x is ruby.
Wxy:	x wants to impress y.

Since there are no names in the sentence F1 above, we do need to include constants in our symbolization key. Instead, we can get by with variables alone. In particular, we may regiment sentence F1 by first identifying its atomic parts. To do so, we may observe that the object in question is said to be one of two colors, and that the person in question wants to impress someone else. Since the same object is either crimson or ruby, we will use the same variable for each: Cx, Rx. We use different variables for 'she' and 'him', where this yields: Wyz. Putting these pieces together, we get: $(Cx \vee Rx) \wedge Wyz$.

Although Cx, Rx, and Wyz are atomic, these expressions are not sentences since it is hard to assign truth-values to such expressions since there is no telling which objects the variables refer to. Matters would be different if the variables were bound by quantifiers since then we would know that we need only find some object which bears the relevant property or relation, or else that every object must do so. Since these atomic expressions include variables that are not bound by quantifiers, we will be refer to such expressions as *open sentences* and the unbound variables in them as *free variables*. Although the precise definitions will be given shortly, it is important to get a sense of the open sentences and the free variables that then include in order to appreciate the role that the quantifiers play in $\mathcal{L}^{\text{FOL}}$.

In order to get a better sense of open sentences as well as some of the subtleties of regimenting sentences in $\mathcal{L}^{\text{FOL}}$, consider the closely related sentence:

G1. Either it is crimson or ruby since she likes those colors.

Now we will need the following symbolization key:

Hxy:	x has the color y.
Lxy:	x likes y.
c:	Crimson
r:	Ruby

Instead of taking 'is crimson' and 'is ruby' to be monadic predicates, we have included constants for each together with a new dyadic predicate for 'has the color'. This is useful since now we can say that she likes each color by using its name. Thus we have the atomic open sentences: Hxc, Hxr, Lyc, and Lyr. We can then build the following regimentation: $(Hxc \lor Hxr) \land (Lyc \land Lyr)$. Despite how similar the sentences F1 and G1 are to each other, their regimentations are very different, and by no means exhaustive.

We may conclude this subsection with a final example:

H1. If it is a dolphin, then it is a mammal and he won't eat it.

Dx:	x is dolphin.
Mx:	x is mammal.
Eyx:	y will eat x.

As before, we may build the atomic open sentences Hx, Mx, and $\neg Eyx$, drawing on these to construct: $Hx \to (Mx \land \neg Eyx)$. By itself, this open sentence appears to refer to some unnamed individual, however, this hardly captures its power. Rather, it is by combining open sentences with quantifiers that we may make general claims of considerable interest. Indeed, such claims are common throughout mathematics, the sciences, and day-to-day life.

7.2.5 Quantifiers

We are now ready to introduce the first-order quantifiers. Unlike singular terms and predicates, quantifiers are a new kind of unary logical operator similar to negation. Consider the sentences:

I1. Someone is happy.	Hx:	x is happy.
I2. Nobody is as guilty as Donald.	Gxy:	x is as guilty as y.
I3. Everybody loves somebody.	Lxy:	x loves y.
	d:	Donald.

It might be tempting to regiment sentence I1 as: Hx. After all, if 'It is happy' is true, then surely someone— whoever 'It' refers to— must be happy. Tempting as this may be, these are very different claims. In order to see how these claims differ, we may consider who these claims *concern* or *are about*. Although it may not be clear who exactly, 'It is happy' is about whoever 'It' refers to where this is some particular individual. By contrast, claiming that someone is happy is not about any particular individual at all. For instance, suppose there are many happy people but 'It' happens to be used to refer to a broken table sitting in the corner. Although sentence I2 may be true, 'It is happy' is false.

Instead of asserting anything about a particular, quantifiers make claims about a *domain* of individuals which we will introduce in the following chapter. Depending on the context of assertion, the domain may differ. Nevertheless, quantified claims may be understood to make claims about the domain which includes all of the individuals that there are. By contrast, open sentences make claims about particular individuals. These are important differences, and we will have a lot more to say about them when we introduce the semantics for $\mathcal{L}^{\text{FOL}}$.

Although the sentence 'It is happy' does not say the same thing as sentence I1, their regimentations start off the same. Given the symbolization key above, 'It is happy' is just: Hx. What is needed to regiment I1 is to add that *there is something such that* it is happy. We will accomplish this by appending an EXISTENTIAL QUANTIFIER $\exists x$, where this yields: $\exists xHx$. As defined below, the quantifier *binds* the variable x in Hx. Since the result does not contain any free variables, $\exists xHx$ is not an open sentence, but rather a sentence of $\mathcal{L}^{\text{FOL}}$.

In order to regiment sentence I2, we may ask what this sentence says. One way to read this sentence is that it is not the case that someone is as guilty as Donald. Unpacking this claim even further: it is not the case that something is such that *it is as guilty as Donald*. However cumbersome, this reading makes the italicized open sentence easy to regiment: Gxd. Although Gxd contains the constant d, it also contains the free variable x which we may bind with an existential quantifier as before: $\exists xGxd$. We may then negate the result: $\neg\exists xGxd$.

Although this provides a fine regimentation of sentence I2, there is another natural reading. Instead of beginning with negation, we may take sentence I2 to express that everyone is not as guilty as Donald. Put otherwise, everything is such that it is not the case that *it is as guilty as Donald*. Here we begin as before with Gxd but then immediately add negation: $\neg Gxd$. Note that this sentence is open since it includes the free variable x. Finally, we add the UNIVERSAL QUANTIFIER $\forall x$, where this yields: $\forall x\neg Gxd$. It turns out that this reading is logically equivalent to the first regimentation of sentence I2 considered above.

In general, $\forall x\varphi$ is logically equivalent to $\neg\exists x\neg\varphi$, and similarly, $\exists x\varphi$ is logically equivalent to $\neg\forall x\neg\varphi$. This means that any sentence which can be regimented with a universal quantifier can be regimented with an existential quantifier together with an appropriate number of negation signs, and *vice versa*. Even though there is no logical difference in regimenting a sentence with one quantifier rather than the other, sometimes one regimentation might seem more natural than the other. Other times, the difference between regimenting with an existential as opposed to a universal quantifier will make no difference at all.

It remains to regiment sentence I3. This says of everything that it loves something. Focusing on 'it loves something', we may take this to mean that there is something such that *it loves that thing.* Thus we have: everything is such that there is something such that it loves that thing. Notice that without the help of variables, it is easy to lose track of which quantifier is binding which variable. It is for this reason that the method that we have employed above of unpacking the sentence in stilted English is of limited utility. Nevertheless, we do need to make sure we know what the claim is saying, and sometimes it helps to write it out a few different ways in English, proceeding in stages as we have above.

Instead of using English to successively unpack I3, we may employ a similar process in $\mathcal{L}^{\text{FOL}}$ itself. For instance, we may begin by taking sentence I3 to say that something is such that it loves somebody which we may partially regiment as $\forall x LSx$ by temporarily introducing 'LSx' for 'x loves somebody'. Next we may take 'LSx' to mean 'there is some y where x loves y' which we may regiment: $\exists y Lxy$. Replacing 'LSx' in our first pass regimentation with '$\exists y Lxy$' yields the final product: $\forall x \exists y Lxy$. In general, sentences with mixed quantifiers are tricky and require a lot of care to get right. Even if some cases you can immediately intuit a correct regimentation, it is important to learn how to systematically decompose complex sentences into simpler parts before putting the pieces back together. This will take some practice.

Having introduced the new primitives symbols included in $\mathcal{L}^{\text{FOL}}$, we may now proceed to define the sentences of $\mathcal{L}^{\text{FOL}}$ a little more rigorously. Nevertheless, the informal remarks that we have made so far should help to provide an intuitive target for our definition to strike.

7.3 The Well-Formed Formulas

Whereas the well-formed sentences (wfss) of $\mathcal{L}^{\text{PL}}$ were defined recursively in terms of the primitive symbols included in $\mathcal{L}^{\text{PL}}$, this approach will not do $\mathcal{L}^{\text{FOL}}$. In order to see why, it is important to appreciate that the wfss of any language are intended to include all and only those expressions which may ultimately be assigned a truth-value by an interpretation. If the wfss of $\mathcal{L}^{\text{FOL}}$ are to be built up in similar manner to the way they were in $\mathcal{L}^{\text{PL}}$, then there will invariably be stages of construction which include the unbound variables that are then bound by quantifiers introduced in later stages of construction. For instance, if we are to construct $\forall x \exists y Lxy$ as above, we must start with Lxy, building up to $\exists y Lxy$, and only then add the final quantifier. However, as mentioned above, there is no easy way to interpret open sentences such as Lxy and $\exists y Lxy$ which includes a free variable.

Rather that admitting sentences which cannot be assigned truth-values by an interpretation of our language, we will arrive at the well-formed sentences of $\mathcal{L}^{\text{FOL}}$ by first defining the more modest notion of a *well-formed formula* of $\mathcal{L}^{\text{FOL}}$, or wff for short. This definition will proceed much as before. Only then will we move to identify those wffs of $\mathcal{L}^{\text{FOL}}$ which do not have free variables and so may be said to be well-formed sentences of $\mathcal{L}^{\text{FOL}}$ whose truth-values will turn out not depend on anything besides the interpretation of $\mathcal{L}^{\text{FOL}}$. As we will see in the following chapter, not all wffs of $\mathcal{L}^{\text{FOL}}$ may claim this virtue.

Collecting the elements that we have introduced so far, the primitive symbols of $\mathcal{L}^{\text{FOL}}$ include:

n-place predicates for $n \geq 0$ with subscripts, as needed	$A^n, B^n, C^n, \ldots, Z^n$ $A_1^n, B_1^n, Z_1^n, A_2^n, A_{25}^n, J_{375}^n, \ldots$
constants with subscripts, as needed	$a, b, c, \ldots, t$ $a_1, w_4, h_7, m_{32}, \ldots$
variables with subscripts, as needed	w, x, y, z $x_1, y_1, z_1, x_2, \ldots$
sentential operators	$\neg$, $\wedge$, $\vee$, $\rightarrow$, $\leftrightarrow$
quantifiers	$\forall, \exists$
parentheses	(,)

In a similar manner to the definition of the wfss of $\mathcal{L}^{\text{PL}}$, we will recursively define the WELL-FORMED FORMULAS (wffs) of $\mathcal{L}^{\text{FOL}}$ as follows:

1. $\mathcal{F}^n\alpha_1, \ldots, \alpha_n$ is a wff of $\mathcal{L}^{\text{FOL}}$ if $\mathcal{F}^n$ is an n-place predicate of $\mathcal{L}^{\text{FOL}}$ and $\alpha_1, \ldots, \alpha_n$ are singular terms (i.e., variables or constants) of $\mathcal{L}^{\text{FOL}}$.
2. For any wffs φ and ψ of $\mathcal{L}^{\text{FOL}}$ and variable α of $\mathcal{L}^{\text{FOL}}$:
   (a) $\exists\alpha\varphi$ is a wff of $\mathcal{L}^{\text{FOL}}$;
   (b) $\forall\alpha\varphi$ is a wff of $\mathcal{L}^{\text{FOL}}$;
   (c) $\neg\varphi$ is a wff of $\mathcal{L}^{\text{FOL}}$;
   (d) $(\varphi \wedge \psi)$ is a wff of $\mathcal{L}^{\text{FOL}}$;
   (e) $(\varphi \vee \psi)$ is a wff of $\mathcal{L}^{\text{FOL}}$;
   (f) $(\varphi \rightarrow \psi)$ is a wff of $\mathcal{L}^{\text{FOL}}$; and
   (g) $(\varphi \leftrightarrow \psi)$ is a wff of $\mathcal{L}^{\text{FOL}}$.
3. Nothing else is a wff of $\mathcal{L}^{\text{FOL}}$.

As brought out in Chapter 1, the definitions above are strictly speaking nonsense. This is because the clauses above *use* rather than *mention* the meta-variables φ and ψ for wffs, as well as new meta-variables for n-place predicates $\mathcal{F}^n$ and singular terms $\alpha_1, \ldots, \alpha_n$. Moreover, it would not help to simply add quotes since the meta-variables that we have used above do not belong to $\mathcal{L}^{\text{FOL}}$. This is what motivated the introduction of corner quotes before, and we may reproduce a similar tactic here. However, instead of doing so, we will follow the common convention of relying on the reader to know where the corner quotes are supposed to go. If you do not remember how to do this, look back to §1.9.3 to review before proceeding.

As before, we will refer to the clauses included in the definition of the wffs of $\mathcal{L}^{\text{FOL}}$ as the COMPOSITION RULES for the wffs of $\mathcal{L}^{\text{FOL}}$ since they tell us how to *compose* wffs from the primitive symbols included in $\mathcal{L}^{\text{FOL}}$. In particular, the first clauses allows us to construct ATOMIC WFFS where the other clauses allow us to construct COMPLEX WFFS. Since we have already seen a number of paradigm cases of wffs in $\mathcal{L}^{\text{FOL}}$, it will help to consider some examples which do not follow the composition rules above. For the sake of readability, we will drop subscripts and superscripts below and throughout much of what follows:

J1. $aFbx$.
J2. $\forall aFbx$.
J3. $\forall yFbx$.
J4. $\forall xFbx$.
J5. $\exists y\forall xFbx$.
J6. $GFbx$.
J7. $G \wedge Fbx$.

The expression in J1 is complete nonsense, or to use the definition given above, J1 is *not* a wff of $\mathcal{L}^{\text{FOL}}$. The reason is that a constant a occurs before the predicate F, but there is no way to achieve this given the composition rules provided above. Were we to remove this misplaced constant, the result would be Fbx which is a wff of $\mathcal{L}^{\text{FOL}}$.

The expression in J2 is not a wff of $\mathcal{L}^{\text{FOL}}$ since the constant a follows the quantifier $\forall$. There is no way to achieve this given our composition rules, and for good reason. For instance, consider the English analogue 'for every Sally, Jim loves it'. Although we might be able to contrive a way to make sense of this in English— something natural languages are very good at— our initial point remains: there is no way to construct J2 given the composition rules for $\mathcal{L}^{\text{FOL}}$, and so J2 is not a wff of $\mathcal{L}^{\text{FOL}}$.

The expression in J3 is a wff of $\mathcal{L}^{\text{FOL}}$. Perhaps this comes as a surprise. After all, the binding variable y which occurs after the quantifier differs from what we might expect to bind the variable x. Accordingly, the quantifier does not make any substantive contribution to the sentence. Be this as it may, we may construct *J3* all the same. We may simulate this effect in English by saying: everything is such that Jim loves Raha. Here, the quantifier 'everything is such that' does no substantive work, where something similar may be said for $\forall y$ in J3.

The expression in J4 is also a wff of $\mathcal{L}^{\text{FOL}}$, but this time the quantifier succeeds in binding the variable x. For instance, we might use this wff to regiment the claim 'Everything is such that Jim loves it', or more naturally, 'Jim loves everything'.

The expression in J5 is a wff of $\mathcal{L}^{\text{FOL}}$, but includes an extra quantifier that does no work. This is because the binding variable y does not bind any variables. Perhaps surprisingly, the same may be said were we to replace y with x. We will have more to say about such cases in the following subsection where we will introduce the notion of *scope*.

The expression in J6 is not a wff of $\mathcal{L}^{\text{FOL}}$ since it includes two concatenated predicates F and G. There is no more latitude for this in $\mathcal{L}^{\text{FOL}}$ than there is for concatenating two sentence letters in $\mathcal{L}^{\text{PL}}$. After all, this would be like saying in English: is red Jim loves it.

The expression in J7 is not a wff of $\mathcal{L}^{\text{FOL}}$, but only for the pedantic reason that we have not included parentheses. Leaving such pedantry to the side, we will go on dropping outermost parentheses when no ambiguity results. For instance, were to wish to bind the variable x with a quantifier, parentheses would be required. Although both $(\forall x G \wedge Fbx)$ and $\forall x(G \wedge Fbx)$ are wffs of $\mathcal{L}^{\text{FOL}}$, the quantifier only succeeds in binding the variable x in the latter. This provides some indication of the role that parentheses play in defining the scope of a quantifier.

7.4 Quantifier Scope

Given a quantified wff of $\mathcal{L}^{\text{FOL}}$ of the form $\exists\alpha\varphi$ or $\forall\alpha\varphi$, we may define φ to be the SCOPE of each of these quantifiers. In considering the composition rules above, the scope of a quantifier is the wff to which that quantifier is applied. Although it is easy to identify the scope of a quantifier when the quantifier is the main connective, this is not always the case.

In the sentence $\exists x Gx \to Gl$, the scope of the existential quantifier is the expression Gx. Would it make a difference if the scope of the quantifier were the whole sentence as in $\exists x(Gx \to Gl)$? In order to answer this question, consider the following symbolization key:

Gx:	x is a guitarist.
l:	Lenny.

Given this key above, $\exists x Gx \to Gl$ reads: if there is some guitarist, then Lenny is a guitarist. By contrast, $\exists x(Gx \to Gl)$ reads: something is such that if it is a guitarist, then Lenny is a guitarist. Recall that the material conditional is true any time the antecedent is false. Let the constant j denote Jack who we may assume is not a guitarist. It follows that the sentence $Gj \to Gl$ is true because Gj is false. Since Jack is such that if he is a guitarist then Lenny is a guitarist, it follows that $\exists x(Gx \to Gl)$ is true. The sentence is true because there is a non-guitarist, regardless of whether Lenny is a guitarist. This may strike you as strange.

It turns out that $\exists x Gx \to Gl$ is logically equivalent to $\forall x(Gx \to Gl)$, and $\exists x(Gx \to Gl)$ is logically equivalent to $\forall x Gx \to Gl$. This oddity does not arise with other connectives, nor does it arise if the variable only occurs in the consequent. For example, $\exists x Gx \wedge Gl$ is logically equivalent to $\exists x(Gx \wedge Gl)$, and $Gl \to \exists x Gx$ is logically equivalent to $\exists x(Gl \to Gx)$. What this brings out is yet another unusual features of the material conditional. Nevertheless, these examples indicate just how important it is to be clear about the scope of a quantifier.

Another type of case to watch out for is when quantifiers occur within each other's scope. For instance, suppose we have an open sentence $\forall x(Lxy \to Px)$ which includes the free variable y. Now suppose we conjoin this sentence with Kxy, wrapping the result in two further quantifiers to produce the wfs $\forall y \exists x(Kxy \wedge \forall x(Lxy \to Px))$. Since $\forall x$ binds the x variables in the innermost parentheses, those variables are not also bound by the $\exists x$ quantifier, though the x in Kxy is bound by this latter existential quantifier. Nevertheless, the existential quantifier has all of $Kxy \wedge \forall x(Lxy \to Px)$ as its scope.

7.4.1 First-Order Sentences

Recall that a declarative sentence (*sentence* for short) is an expression that can either be true or false on an interpretation. In $\mathcal{L}^{\text{PL}}$, every wfs could be assigned a truth-value given an interpretation, thereby justifying the name 'wfs of $\mathcal{L}^{\text{PL}}$'. By contrast, not all wffs of $\mathcal{L}^{\text{FOL}}$ have truth-values given an interpretation alone. For instance, consider the following:

Lxy:	x loves y
b:	Boris

The expression Lzz is an atomic wff since a two-place predicate is followed by two singular terms, in this case both of which are instances of the variable z. We may then ask what it would mean for Lzz to be true. For instance, perhaps it means that z is self loving in the same way that Lbb means that Boris loves himself. However, since z is a variable, it does not name something the way that a constant like b does. Whereas there is a clear way to interpret constants, there is no equally determinate way to interpret variables.

In order to be able to say which wffs of $\mathcal{L}^{\text{FOL}}$ can be assigned truth-values and which cannot, it will help to provide precise definitions of some of the notions that we made intuitive use of above. To begin with, we may provide the following recursive definition of free variables:

1. α is free in $\mathcal{F}^n\alpha_1, \ldots, \alpha_n$ if $\alpha = \alpha_i$ for some $1 \leq i \leq n$ where α is a variable, $\mathcal{F}^n$ is an n-place predicate, and $\alpha_1, \ldots, \alpha_n$ are singular terms.
2. If φ and ψ are wffs of $\mathcal{L}^{\text{FOL}}$ and α and β are variables, then:
   (a) α is free in $\exists\beta\varphi$ if α is free in φ and $\alpha \neq \beta$;
   (b) α is free in $\forall\beta\varphi$ if α is free in φ and $\alpha \neq \beta$;
   (c) α is free in $\neg\varphi$ if α is free in φ;
   (d) α is free in $(\varphi \land \psi)$ if α is free in φ or α is free in ψ;
   (e) α is free in $(\varphi \lor \psi)$ if α is free in φ or α is free in ψ;
   (f) α is free in $(\varphi \rightarrow \psi)$ if α is free in φ or α is free in ψ;
   (g) α is free in $(\varphi \leftrightarrow \psi)$ if α is free in φ or α is free in ψ;
3. Nothing else is a free variable.

Observe the manner in which the definition above follows the same recursive structure as the composition rules for $\mathcal{L}^{\text{FOL}}$. Given any wffs of the form $\exists\alpha\varphi$ or $\forall\alpha\varphi$, we may refer to α as the binding variable of these quantifiers where every free occurrence of α in φ is bound by the quantifiers $\exists\alpha$ and $\forall\alpha$ in $\exists\alpha\varphi$ and $\forall\alpha\varphi$, respectively. Accordingly, quantifiers bind all free occurrences of their binding variable which occur within their scope.

A wff of $\mathcal{L}^{\text{FOL}}$ is a WELL-FORMED SENTENCE (wfs) of $\mathcal{L}^{\text{FOL}}$ if it does not include any free variables, and an OPEN SENTENCE of $\mathcal{L}^{\text{FOL}}$ otherwise. Since there are wffs of $\mathcal{L}^{\text{FOL}}$ that include free variables, not all wffs are wffs of $\mathcal{L}^{\text{FOL}}$. Consider the examples:

K1. $\forall x \forall x(Ex \vee Dy) \rightarrow \exists z(Rzx \rightarrow Lzx)$.
K2. $\forall x(\forall x(Ex \vee Dy) \rightarrow \exists z(Rzx \rightarrow Lzx))$.
K3. $\exists y \forall x(\forall x(Ex \vee Dy) \rightarrow \exists z(Rzx \rightarrow Lzx))$.

The scope of the first universal quantifier on the left in K1 is $\forall x(Ex \vee Dy)$. Although x occurs in $\forall x(Ex \vee Dy)$, it is a bound occurrence. Rather, y is the only free variable in $\forall x(Ex \vee Dy)$. Since $y \neq x$, we may observe that y remains free in $\forall x \forall x(Ex \vee Dy)$. Moving to the consequent of K1, both occurrences of z are bound, and neither occurrence of x are bound. Accordingly, K1 is a wff of $\mathcal{L}^{\text{FOL}}$, but not a sentence of $\mathcal{L}^{\text{FOL}}$.

We may change the scope of the universal quantifier on the far left in K1 by adding an additional pair of parentheses as given in K2. Now the scope of the left most universal quantifier is $\forall x(Ex \vee Dy) \rightarrow \exists z(Rzx \rightarrow Lzx)$. Whereas x is bound by the universal quantifier in the antecedent, x is free in the consequent, and so only these latter occurrences of x are bound by the outermost universal quantifier in K2. Nevertheless, y remains free throughout, and so K2 is not a sentence of $\mathcal{L}^{\text{FOL}}$ though it is a wff of $\mathcal{L}^{\text{FOL}}$.

In order to bind the free occurrence of y in K2, we may add a quantifier whose binding variable is y as given by K3. Since K3 does not include any free variables, K3 is a sentence of $\mathcal{L}^{\text{FOL}}$ and so may be assigned a truth-value given an appropriate interpretation of $\mathcal{L}^{\text{FOL}}$.

7.5 First-Order Regimentation

We now have the basic elements of $\mathcal{L}^{\text{FOL}}$ in place. Translating more complicated sentences will only be a matter of knowing the right way to combine predicates, constants, quantifiers, variables, and sentential connectives. Consider these sentences and symbolization key:

L1. Every coin in my pocket is a loonie.
L2. Some coin on the table is a dime.
L3. Not all the coins on the table are loonies.
L4. None of the coins in my pocket are dimes.

Cx:	x is a coin.	Lx:	x is a loonie.
Px:	x is in my pocket.	Dx:	x is a dime.
Tx:	x is on the table.		

Sentence L1 is most naturally translated with a universal quantifier. The universal quantifier says something about everything, not just about coins, or the coins in my pocket. Accordingly, we may take L1 to say that, for anything, *if* it is a coin and in my pocket, *then* it is a loonie. So we can translate it as $\forall x((Cx \land Px) \to Lx)$.

Since sentence L1 is about coins that are both in my pocket *and* that are loonies, it might be tempting to translate it using a conjunction. However, the sentence $\forall x((Cx \land Px) \land Lx)$ would mean that everything is a coin in my pocket and a loonie. It would also be wrong to regiment sentence L1 as $\forall x(Cx \to (Px \land Lx))$ since this say that everything that is a coin is in my pocket and a loonie. This is a very strong claim, and is unlikely to be true since there are a lot of coins out there which are neither in my pocket or loonies.

These examples bring out the idea of *restricting* a universal quantifier. Since saying something about everything in an unrestricted way is only very rarely something that we intend to do, universal claims almost always take the following form:

M1. $\forall \alpha(\varphi(\alpha) \to \psi)$.

Here $\varphi(\alpha)$ is a wff of $\mathcal{L}^{\text{FOL}}$ in which α occurs as a free variable, and so α as it occurs in $\varphi(\alpha)$ is bound by the quantifier $\forall \alpha$. Accordingly, the sentence M1 says that ψ holds of everything that satisfies the condition φ. Although we have not required ψ to also include α as a free variable, it is typical that ψ would include free occurrences of α. For instance, in the case above, we compared taking $Cx \land Px$ to restrict the quantifier with taking just Cx to restrict the quantifier. Whereas the former allows us to make universal claims about just the coins in my pocket, the latter allows us to make claims about all coins.

Sentence L2 is most naturally translated with an existential quantifier. It says that there is some coin which is both on the table and which is a dime. So we can translate it as $\exists x(Cx \land (Tx \land Dx))$. Notice that we did not use a conditional to restrict the existential quantifier in the same way as we did with the universal quantifier. Instead, we used conjunction to say that there is at least one thing which is a coin and moreover it is on the table and is a dime. Using conjunction with an existential quantifier is a common pattern.

What would it mean to write $\exists x(Cx \to (Tx \land Dx))$? This says that there is something which is a dime on the table *if it is a coin*. Suppose that there are no coins that are dimes on the table, but there is at least one thing which is not a coin, say the planet Jupiter which we will regiment by the constant j. Since the planet Jupiter is not a coin, it follows that $Cj \to (Tj \land Dj)$ is true since the antecedent Cj is false. As a result, $\exists x(Cx \to (Tx \land Dx))$ is true even though we have assumed that there are no coins that are dimes on the table. More generally, whenever there is something that is not a coin, $\exists x(Cx \to (Tx \land Dx))$ will be true, making this an extremely weak claim. Although the conditional is often used to restrict a universal quantifier, a conditional within the scope of an existential quantifier results in extremely week claims which we almost never intend to assert. Thus it's a good rule of thumb that you should avoid putting conditionals in the scope of existential quantifiers.

Sentence L3 can be paraphrased as, 'It is not the case that every coin on the table is a loonie'. So we can translate it as $\neg\forall x((Cx \wedge Tx) \rightarrow Lx)$. Alternatively, we paraphrase sentence L3 as, 'Some coin on the table is not a loonie'. We would then translate sentence L3 as $\exists x((Cx \wedge Tx) \wedge \neg Lx)$. Despite their superficial differences, these two regimentations are logically equivalent. More generally, $\neg\forall x\varphi$ and $\exists x\neg\varphi$ are logically equivalent, as are $\neg(\varphi \rightarrow \psi)$ and $\varphi \wedge \neg\psi$. This is something that we will prove to in the following chapter.

Sentence L4 can be paraphrased as, 'It is not the case that there is a coin in my pocket that is a dime'. This can be regimented by $\neg\exists x(Cx \wedge (Px \wedge Dx))$. It might also be paraphrased as 'Every coin in my pocket is not a dime' which we may regiment by $\forall x((Cx \wedge Px) \rightarrow \neg Dx)$. These two translations are logically equivalent, and so equally regiment sentence L4.

7.6 Paraphrasing Pronouns

When regimenting English sentences in $\mathcal{L}^{\text{FOL}}$, it is often helpful to paraphrase the sentence in English in a manner that exposes the logical features of that sentence. We have already provided a number of examples of this above. When paraphrasing, it is important that you do not accidentally make changes to the logical structure of the sentence since mistakes at this stage will end up resulting in the wrong regimentation.

Paraphrasing often requires departing from the superficial structure of the target sentence. For instance, consider the following examples and symbolization key:

N1. If MIT is in session, then it is full of activity.
N2. If some institute is in session, then it is full of activity.

Sx: x is in session.
Ax: x is full of activity.
m: MIT

Sentence N1 and sentence N2 have the same words in the consequent, but they cannot be regimented in the same way. This is because the 'it' in sentence N1 is not bound by a quantifier but rather indicates the same subject while avoiding repetition. By contrast, both occurrences of 'it' in sentence N2 are bound by the outermost quantifier. Whereas there is nothing to change about sentence N2, we may paraphrase sentence N1 as follows:

O1. If MIT is in session, then MIT is full of activity.

Compare the following regimentations of the sentences N1:

P1. $Sm \to Ax$.
P2. $Sm \to Am$.

Although it might be tempting to try to regiment the 'it' in the original sentence N1 by a variable, this would result in the open sentence P1 in place of the wfs P2. Since the original sentence says something entirely about MIT and nothing about some yet to be bound variable the way 'It is red' does, sentence P2 provides a better regimentation than P1 does.

7.7 Ambiguous Predicates

Suppose we want to regiment this sentence:

Q1. Jenny is a skilled surgeon.

Consider the following symbolization key:

Kx: x is skilled
Rx: x is a surgeon
j: Jenny

This yields the following:

R1. $Kj \wedge Rj$.

In English, this reads: Jenny is skilled and Jenny is a surgeon. Here one may object that being skilled and also being a surgeon is not the same thing as being a skilled surgeon. For instance, perhaps it is possible to be both skilled and a surgeon without being a skilled surgeon. Accordingly, we could have specified the following predicate instead:

Sx: x is a skilled surgeon
j: Jenny

We may then provide the following regimentation:

S1. Sa.

Considering sentence Q1 on its own, it may be unclear whether to go with regimentation R1 or S1. However, in the context of an argument, there may be good reason to go one way rather than the other. For instance, suppose that we want to regiment this argument:

T1. The hospital will only hire a skilled surgeon.
T2. Adina is skilled but not a surgeon.
T3. Billy is a surgeon, but not skilled.
T4. $\underline{\text{Jenny is a skilled surgeon.}}$
T5. Therefore, the hospital will hire Jenny and not Billy or Adina.

Here we need to distinguish being a *skilled surgeon* from merely being a *surgeon*. Consider this symbolization key:

Hx:	The hospital hires x.	a:	Adina.
Kx:	x is skilled.	b:	Billy.
Rx:	x is a surgeon.	j:	Jenny.

Now the argument can be regimented this way:

U1. $\forall x(Hx \to (Kx \land Rx))$
U2. $Ka \land \neg Ka$
U3. $Rb \land \neg Kb$
U4. $\underline{Kj \land Rj}$
U5. $Hj \land \neg(Hb \lor Ha)$

Although we have not yet provided a semantic nor theory of logical consequence for $\mathcal{L}^{\text{FOL}}$, it is worth considering whether the formal argument in $\mathcal{L}^{\text{FOL}}$ is valid. Whereas the argument in English might have seemed somewhat compelling— at least it is the kind of argument that it is common to hear— its formalization may be used to reveal its invalidity. The next chapter will provide resources to prove that this is so by providing a counterexample.

Compare the previous argument to the following:

V1. $\underline{\text{Carol is a skilled surgeon and a tennis player.}}$
V2. Therefore, Carol is a skilled tennis player.

We may regiment this argument as follows:

W1.	$\underline{(Rc \land Kc) \land Tc}$	Tx:	x is a tennis player
W2.	$Tc \land Kc$	c:	Carol

This argument is valid in $\mathcal{L}^{\text{FOL}}$, but the original argument in English is not. Something has gone wrong with our regimentation. The problem is that there is a difference between being *skilled as a surgeon* and *skilled as a tennis player*. Regimenting this argument correctly requires two separate predicates, one for each type of skill. Thus we may add the following:

K_1x:	x is skilled as a tennis player
K_2x:	x is skilled as a surgeon

We may now regiment the argument in this way:

X1. $\underline{(K_2c \land Rc) \land Tc}$
X2. $K_1c \land Tc$

Like the English argument it regiments, this $\mathcal{L}^{\text{FOL}}$ argument is invalid.

Similar problems can arise with predicates like 'is good', 'is big', 'is tall', etc. Just as skilled surgeons and skilled tennis players have different skills, it's obvious that big dogs, big mice, and big problems are all big in different ways. Must we always distinguish between different ways of being skilled, good, big, or tall? Not necessarily. If you are translating an argument that is just about dogs, it is fine to use the predicate 'x is big'. However, if the argument is also about mice, it might be important to let use the predicate 'x is big for a dog' instead. In general, we try to introduce as few predicates as possible while nevertheless capturing the intended meaning of the sentence or argument in question, as well as the intuitive validity or invalidity of the arguments in English that we might consider.

7.8 Multiple Quantifiers

Consider the following sentences and symbolization key:

Dt:	x is a dog.	Y1.	Fifi is a dog.
Fxy:	x is a friend of y.	Y2.	Gerald is a dog owner.
Oxy:	x owns y.	Y3.	Someone is a dog owner.
f:	Fifi	Y4.	All of Gerald's friends are dog owners.
g:	Gerald	Y5.	Dog owners are friends with dog owners.

Sentence Y1 is easy: Df.

Sentence Y2 can be paraphrased as, 'There is a dog that Gerald owns' which can be translated as $\exists x(Dx \wedge Ogx)$. Also no so difficult.

Sentence Y3 can be paraphrased as, 'There is something such that it is a dog owner'. The subsentence 'it is a dog owner' is just like sentence Y2, except that it is about *it* rather than being about Gerald. We can then regiment the sentence Y3 as $\exists y \exists x(Dx \wedge Oyx)$, replacing 'is a dog owner' with our previous regimentation.

Sentence Y4 can be paraphrased as, 'Every friend of Gerald is a dog owner'. We can expand this to: 'Everything is such that, if it is a friend of Gerald, then it is a dog owner'. Translating part of this sentence, we get $\forall x(Fxg \to x \text{ is a dog owner})$. Again, it is important to recognize that 'x is a dog owner' is structurally just like sentence Y2. Since we already have a quantifier binding x, we will need a different variable for the existential quantifier. Any other variable will do. Using z, sentence Y4 can be translated as $\forall x(Fxg \to \exists z(Dz \wedge Oxz))$.

Sentence Y5 can be paraphrased as 'For any x that is a dog owner, there is a dog owner who is a friend of x'. Partially translated, this becomes:

$$\forall x\Big[x \text{ is a dog owner} \to \exists y(y \text{ is a dog owner} \wedge Fxy)\Big].$$

Completing the translation, sentence Y5 becomes:

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

Regimenting English sentences in $\mathcal{L}^{\text{FOL}}$ will take some practice, and often there will be more than one way to go. When you come up with a regimentation, it is often worth considering if there are any other regimentations that you could have provided. Often there are, and they are not always logically equivalent. Sometimes what this means is that the original claim is ambiguous. Other times, some regimentations will be more natural than others.

Consider the following sentences and symbolization key:

Z1. Imre likes everyone that Karl likes.
Z2. There is someone who likes everyone who likes everyone that he likes.

Z3. x likes y.
Z4. Imre.
Z5. Karl.

Sentence Z1 can be partially regimented as: $\forall x(\text{Karl likes } x \to \text{Imre likes } x)$ which becomes $\forall x(Lkx \to Lix)$. But we might have been able to skip to this final regimentation.

Sentence Z2 is much more complex. There is little hope of writing down the whole regimentation at once, so we will proceed in stages. An initial regimentation might look like:

$$\exists x \text{ everyone who likes everyone that } x \text{ likes is liked by } x$$

The part that remains in English is a universal sentence, so we translate further:

$$\exists x \forall y(y \text{ likes everyone that } x \text{ likes} \to x \text{ likes } y).$$

The antecedent of the conditional is structurally just like sentence Z1, with y and x in place of Imre and Karl. So sentence Z2 can be completely regimented as follows:

$$\exists x \forall y \Big[\forall z(Lxz \to Lyz) \to Lxy\Big]$$

When symbolizing sentences with multiple quantifiers, it is best to proceed by small steps. Paraphrase the English sentence so that the logical structure is readily regimented in $\mathcal{L}^{\text{FOL}}$. Then regiment piecemeal, replacing the daunting task of regimenting a long or dense sentence with many simple tasks of regimenting shorter sentences.