0 Predicates and Quantifiers

Generally, a language[1] is a systematic[2] way of encoding ideas into sequences of symbols. At a minimum, this requires a language to specify a *syntax* and a *semantics* in order for us to be able to understand it and use it for communication. The ***syntax*** of a language refers to its physical and mechanical structure: the glyphs used to write it, the conventions used for determining spelling, and the grammatical rules that govern which arrangements of symbols are acceptable or not.[3] The smallest functional unit of syntax is usually called a *grapheme.* The meaning of a particular string of symbols is given by the language's ***semantics***.[4] The semantics of a language is what determines how syntax and ideas relate to each other.

syntax

semantics

In establishing a new language for mathematics, we therefore have the burden of defining the syntax of this new language along with a semantic interpretation of that syntax that corresponds with our historical and intuitive use of logical and mathematical language.[5] We proceed then by abstracting from natural language a core *logical* framework we can use to express the kind of reasoned arguments common in mathematical speech.[6] Once we've got this language—which we will call the ***first-order logic***—we will be able develop *theories*[7] that describe mathematical objects.

[1] For our purposes, we will only be considering *written* languages, and any speech derived from them

[2] Many natural languages (*e.g.*, English, Gaeilge, Greek) have no official *"definition"* or regulating body, and even languages that do (*e.g.*, Spanish, French, Chinese) often have their rules violated in everyday speech. Despite that, there *is some* systematicity to natural language, as evidenced by the fact that only one of the two sentences below parses as intelligible English:

- *"My favorite season is autumn."*
- *"Is your autumn favorite season."*

[3] The fact that *"syntax"* is a noun with that spelling—as opposed to an adjective with the spelling *"sintacks"*—is a feature of English syntax.

[4] The use of the same word *"football"* to mean different things by Americans and Britons is an example of a semantic distinction between American and British English.

[5] This is a somewhat controversial claim. The language we will be developing is the *classical* first-order logic, and its purpose is to formally encode (a fragment of) the kind of logical reasoning that has been studied and engaged in since the time of the ancient Greeks. The careful reader must nonetheless note that *intuitionistic logic*—one of the many other forms of logic out there that we could have relied on instead—is *not* synonymous with *"intuitive logic"* and provides a significantly different semantics for the first-order logic.

[6] We will eventually call (arguments that can be reduced to) these sorts of arguments *"proofs."*

[7] *E.g.*, the theory of geometry according to Euclid, the theory of arithmetic according to Peano, the theory of sets according to Zermelo and Fraenkel.

0.1 Truth and Falsity

The most basic kind of mathematical statement is a *declarative* sentence, claiming to describe a property about an object. Here are a few of examples of such sentences.

- *"One plus one is equal to two."*
- *"Two plus two is equal to three."*
- *"The sum of the squares of the lengths of any two perpendicular sides of a triangle is equal to the square of the length of that triangle's remaining side."*
- *"The singular (co)homologies of homotopy-equivalent spaces are isomorphic."*

Whether or not we understand all (or any) of the words in these sentences, they are all declarative. Additionally, they all share a property: if someone were to clarify the meaning of these sentences by explaining precisely what all of the terms in each one mean, we would agree that they are all either *true* or *false* but *not both.* In other words, these declarative sentences carry exactly one consistent ***truth value***—either ***true*** or ***false***, denoted by the respective symbols $\top$ and $\bot$, depending on whether the sentence accurately describes its object. Of course, we can't actually know which of the two truth values corresponds to any of the above sentences unless we understand what the words in those sentences mean. To make the point more clear, consider the following expression.

$\top$ $\bot$

- *"x plus one is equal to two."*

You might think you have an idea of what the words *"plus"* and *"equal"* mean, and who the terms *"one"* and *"two"* refer to, but what about "x"? We should hope for "x" to refer to an object of a compatible type[8] given the surrounding context, otherwise the sentence would fail to make any sense at all.[9] But, if x refers to a number, which one? Different numbers, when substituted for x in that expression, may produce different truth values! *Until we know who x is, we can't understand that sentence.*

[8] This is a deep and interesting topic that we will not be developing further.

[9] What if x refered to *"Mazda Miata"*?

It should be noted that not every declarative sentence necessarily carries exactly one consistent truth value. Some sentences may have too many admissible truth values, while some may not carry any at all. Consider the following examples.

› *"This sentence is* true."
› *"This sentence is* false."
› *"If this sentence is* true, *then two is an odd number."*
› *"If two is an even number, then this sentence is* false."

As such, we need to distinguish between the objects of our discussion and the descriptions we make about them. Objects can not carry truth values, but when we form a grammatically correct sentence about a valid object in our new language, we want to be sure that sentence definitely has exactly one consistent truth value.

0.2 *Terms and Variables*

term
predicate

Any symbol in an expression that denotes an object is typically called a ***term***, which come in two flavors: constants and variables.[10][11] A ***predicate*** is is descriptive claim about one or more terms such that, once the terms are all understood, we obtain exactly one consistent truth value. For example, the expression below defines a predicate π that describes one term x.[12]

$:=$

$$\pi(x) := \text{“\textit{The number} } x \text{ \textit{is prime.}”}$$

If we now utter "$\pi(2)$," we can intuitively understand this as a *true* declarative sentence, whereas "$\pi(4)$" would be a *false* declarative sentence.[13] These two expressions would translate to the following English language sentences respectively.

› *"The number* 2 *is prime."*
› *"The number* 4 *is prime."*

constant

The terms "2" and "4" in the above expressions are examples of ***constants*** since they refer to particular objects. By letting $n := 4$, the symbol "n" in the expression "$\pi(n)$" is also a constant, so $\pi(n)$ also has one truth value. This is translated to English below.

› *"The number n is prime."*

In fact, despite these sentences being distinct from the previous one, they communicate the same message as long as we know $n = 4$ (which we do thanks to the definition we made).[14] However, the symbol x in the following expression is not a constant, since we don't understand x to be referring to any object in particular.

› *"The number x is prime."*

variable

As a result, we have no idea what the above sentence really means, and we won't be able to resolve this dilemma until we have more information about x. We call a term like this one a ***variable*** since it has not been fixed to any particular (constant) value. Now, although the presence of a variable in the above sentence prevents us from understanding what it means, it can be remedied by defining what x is, converting x into a constant term. On the other hand, we could leave the particular identity of x vague—intentionally keeping x a variable—and disambiguate the meaning of the variable by indicating *how it's supposed to vary* more precisely. For example, think about the expression below.

[10] We will still often call them *"objects."*

[11] *N.B.* A consequence of our desire to use language with precision and consistency is that a symbol can only refer to one particular thing at a time within a given context.

[12] In general, when we need to *define* a symbol to refer to an object, we will introduce that definition using the symbol $:=$ to make explicit that this is simply the assignment of a label to an object. The label (reference) goes on the same side as the colon, and the object (referent) goes on the side without the colon. For example: the expression *"let $k := 3$"* defines a new symbol k, and whenever we use k from this point onwards within this context, we can think of that symbol as standing in for the number 3. We could then write $k + 2 \geqslant 0$ and understand that it means precisely $3 + 2 \geqslant 0$.

[13] The astute reader will recognize that this looks like the syntax for function application that might be found in expressions like $f(x) = x^2 + 1$. This is no coincidence: one way to think about predicates is as *functions* that take one or more *terms as inputs* and produce a *truth value as output.*

[14] *N.B.* The use of $=$ here instead of $:=$ is intentional. A sentence like $m = 5$ should carry a truth value because it asserts that one object is the same as another one. A sentence like $m := 5$ does not carry a truth value because it is not describing a property of an object—this is more like a command (an *imperative*) that the label m be attached to the object referred to by 5. There are many consequences of this, but one to keep in mind right now is that definitions $:=$ do not require justification, but statements of equality $=$ do require proof, and can therefore be "known" to be *true* or *false.* We will expand on this later in more detail.

› *"Every number is prime."*

There's no particular object we could substitute in for x in $\pi(x)$ so that it would mean the same thing as the sentence above. Instead, we will indicate that x is meant to stand in for *any particular object* by ***universally quantifying*** the variable x, which we use the symbol $\forall$ to denote. The syntax will look something like this.

$\forall$

$$\forall x(\ldots)$$

"For all x, …"

So, for example, the expression below means precisely that *"Every number is prime."*

$$\forall x\big(\pi(x)\big)$$

"For all x, $\pi(x)$."

In the expression $\forall x(\pi(x))$, the symbol x is still a variable, but it is ***bound*** to the scope inside the parentheses by the universal quantifier. That means, in an expression like $\forall x(\ldots)$, that inside of the $(\ldots)$ brackets we are meant to understand x as a term standing in for any possible object under consideration.[15] Therefore, the expression $\forall x(\pi(x))$ asserts that $\pi(x)$ is *true* about *every* object that we could possibly substitute in for x in the expression. In contrast, think of the following expression.

[15] More on this later.

› *"A prime number exists."*

If we try to communicate this by picking a particular object, such as 2, and asserting $\pi(2)$, we are again not really communicating what the sentence above says; in fact, we are saying *more.* The sentence above says that there is some number out there—possibly one we could even find for ourselves—that has the property of being prime, but it doesn't specify *which* number that is. The expression $\pi(2)$ not only asserts the existence of a prime number, it also says that 2 specifically is one such example.[16] To resolve this, we introduce the ***existential quantifier***, which we denote with the $\exists$ symbol, to bind a variable and indicate that *there exists some object* out there that could be substituted for it, but that we don't necessarily know who exactly it could be.

$\exists$

[16] If this seems like a stupid distinction, think about the difference between the following two sentences:
› Someone won the lottery.
› ***YOU*** (the reader) won the lottery.

$$\exists x(\ldots)$$

"There exists an x such that…"

So, for example, the expression below says *"There exists a prime number."*

$$\exists x\big(\pi(x)\big)$$

"There exists an x such that $\pi(x)$."

These are the two types of quantifiers that we will introduce for *binding* variables, allowing us to distinguish between two kinds of variables. Within a particular scope, a ***bound variable*** is one which is bound by a quantifier sitting outside that scope. A ***free variable***, in contrast, is not bound by any quantifier that applies to that scope. An expression that contains free variables *does not mean anything* until those free variables are either resolved as constants or bound by quantifiers!

0.3 Universes of Discourse

In order to make sense of the quantifiers we just introduced, we should clarify what it is we consider an *"object."* Suppose we define a binary[17] predicate as below.

$$\geqslant(x, y) \coloneqq \text{“}x \textit{ is greater than } y.\text{”}$$

[17] The ***arity*** of a predicate indicates the number of arguments that must be provided in order for the prediate to return a sensible truth value. The examples we had seen before this section were *unary* predicates because they only took one argument, and the one introduced here is *binary* because it requires two arguments.

The usual convention with binary predicates—especially those like $\geqslant$ that have *"special notation"* introduced for them—is that we write the predicate using *infix* as opposed to *prefix* notation.[18] Now let's consider the following expression.

$$\forall x(x \geqslant 3)$$

In English, this claims *"Everything is greater than or equal to* 3." This seems to be a grammatically well-formed sentence, and it's certainly describing some sort of object, so we should be able to infer a truth value from it—but there are several immediate impediments. For example, if we try to reason about this sentence by evaluating "$x \geqslant 3$" when x represents an object like a human, you might be left wondering what it means to say *"Jannik Sinner is greater than or equal to* 3." That doesn't seem to make sense, but okay—we usually think of $\geqslant$ as being a predicate that applies to *numbers,* and while that wasn't specified anywhere, evaluating expressions like "$x \geqslant 3$" when "x" represents a number like "5" seems to make more sense, as in "$5 \geqslant 3$." It seems like we can find values of x for which the expression is *false* (*e.g.,* 1), suggesting that $\forall x(x \geqslant 3)$ is *false.* Okay… but what about an expression like the one below?

$$\exists x\Big(x^2 + 1 = 0\Big)$$

We can express this using a predicate $\iota(x) :=$ "*x squared plus* 1 *is equal to* 0" and forming a sentence like $\exists x(\iota(x))$.[19] Is this *true* or *false?* Well, it becomes much more obvious now that the answer is: ***it depends!*** If x represents something like a natural, integral, rational, or real number, then $\exists x(\iota(x))$ is clearly *false,* but if we are allowed to reason about *complex numbers* like $i := \sqrt{-1}$, then we can clearly see that $\iota(i)$ is *true* because $\left(\sqrt{-1}\right)^2 + 1 = -1 + 1 = 0$, so that $\exists x(\iota(x))$ is *true.* In fact, the earlier question about the truth value of $\forall x(x \geqslant 3)$ is resolved in the exact same way: ***it depends*** on whether or not the objects under consideration are comparable to "3," and, if they are, then ***it depends*** on how that comparison hashes out for each object.

The important missing piece required to understand these sentences—and thus resolve their truth values—is a ***universe of discourse*** that specifies what counts as an *"object"* for the purposes of our discussion. The universal $\forall$ quantifier asserts a condition about *every* object in the universe, while the existential $\exists$ quantifier only asserts a condition about *at least one* object in the universe. This intuitively means that changing our universe of discourse changes the collection of objects that $\forall$ and $\exists$ have access to, which changes the *meaning* of our sentences. If we take a universe of discourse consisting of every natural number, then $\forall x(x \geqslant 3)$ is *false;* but if our universe of discourse only contains the natural numbers greater than or equal to 3, then $\forall x(x \geqslant 3)$ is *true.*

It is therefore immensely important to specify a universe of discourse *before* we begin speaking in our new language if we want to be sure that our expressions actually mean anything concrete.[20] We are necessarily incapable of doing this *formally,* so it must be done *informally.*[21] As such, speaking *informally,* we say that a universe of discourse is *"the collection of objects under consideration"* for the purposes of our communication. The import of this discussion is, no matter which particular universe we specify, that…

We will always consider our universe of discourse to be nonempty!

[18] Informally: suppose that φ is something that behaves like a binary function (*e.g.,* a real-valued function of two variables, a binary predicate, a binary relation). When we apply φ on two inputs x and y, we obtain some result that is usually denoted $\varphi(x, y)$. This is especially common when φ maps *objects to objects.* As a concrete example, consider the real-valued function $f(x, y) := x^2 + y$ of two variables. We would then say that $f(2, 1)$ is the output of f when applied to the inputs 2 and 1, which is $2^2 + 1$, which is 5. This is called ***prefix notation*** because the function name f is written *before* its arguments x and y. However, if φ maps *objects to truth values* (like a predicate or a relation) or *truth values to truth values* (like a logical connective, which we will talk about in the next set of notes), then it is common to instead write $x \varphi y$ to denote the output of φ on the inputs x and y. As an example, consider $\geqslant(x, y) :=$ *"x is greater than or equal to y."* You would never expect to see $\geqslant(3, 1)$, but it is very common to see $3 \geqslant 1$. Since the name of the function $\geqslant$ is written *between* its two inputs, we refer to this as ***infix notation.*** There is an uncommon third alternative—***postfix notation***—which we leave to the reader.

[19] Once we have introduced all of the machinery of our new language and we begin speaking about actual mathematical objects, we will be able to simply write $\exists x(x^2 + 1 = 0)$, as each of the symbols will be formally understood as a constant (*i.e.,* 0, 1), a function (*i.e.,* $+$, $◌^2$), a predicate (*i.e.,* $=$), or a logical symbol (*i.e.,* $\exists$, x).

[20] The point of this sentence is that, while we can say $\forall x(x \geqslant 3)$ is a grammatically correct sentence (assuming that $\geqslant$ is a binary predicate), we won't know what it actually means without context about what our universe of discourse is.

[21] Despite this, we can be surprisingly precise if we put the appropriate effort in, and there are some interesting tricks that can be done here.