

9 Quantifiers

Logical statements usually include *variables*, which range over sets of possible instances, often referred to as *universes*. We use quantifiers to specify that something holds for all possible instances or for some but possibly not all instances.

Universal and existential quantifiers. We introduce the concept by taking an in-depth look at a result we have discussed in Chapter II.

EUCLID'S DIVISION THEOREM. Letting n be a positive integer, for every integer m there are unique integers q and r , with $0 \leq r < n$, such that $m = nq + r$.

In this statement, we have n, m, q, r as variables. They are integers, so \mathbb{Z} is the universe, except that some of the variables are constrained further, that is, $n \geq 1$ and $0 \leq r < n$. The claim is “for all” m “there exist” q and r . These are quantifiers expressed in English language. The first is called the *universal quantifier*:

$\forall x [p(x)]$: for all instantiations of the variable x , the statement $p(x)$ is true.

For example, if x varies over the integers then this is equivalent to

$$\dots \wedge p(-1) \wedge p(0) \wedge p(1) \wedge p(2) \wedge \dots$$

The second is the *existential quantifier*:

$\exists x [q(x)]$: there exists an instantiation of the variable x such that the statement $q(x)$ is true.

For the integers, this is equivalent to

$$\dots \vee q(-1) \vee q(0) \vee q(1) \vee q(2) \vee \dots$$

With these quantifiers, we can restate Euclid's Division Theorem more formally:

$$\forall n \geq 1 \forall m \exists q \exists 0 \leq r < n [m = nq + r].$$

Negating quantified statements. Recall de Morgan's Law for negating a conjunction or a disjunction:

$$\begin{aligned} \neg(p \wedge q) &\Leftrightarrow \neg p \vee \neg q; \\ \neg(p \vee q) &\Leftrightarrow \neg p \wedge \neg q. \end{aligned}$$

The corresponding rules for quantified statements are

$$\begin{aligned} \neg(\forall x [p(x)]) &\Leftrightarrow \exists x [\neg p(x)]; \\ \neg(\exists x [q(x)]) &\Leftrightarrow \forall x [\neg q(x)]. \end{aligned}$$

We get the first line by applying de Morgan's first Law to the conjunction that corresponds to the expression on the left hand side. Similarly, we get the second line by applying de Morgan's second Law. Alternatively, we can derive the second line from the first. Since both sides of the first line are equivalent, so are its negations. Now, all we need to do it to substitute $\neg q(x)$ for $p(x)$ and exchange the two sides, which we can because \Leftrightarrow is commutative.

Big-Oh notation. We practice the manipulation of quantified statements by discussing the big-Oh notation for functions. It is commonly used in statements about the convergence of an iteration or the running time of an algorithm. We write \mathbb{R}^+ for the set of positive real numbers.

DEFINITION. Let f and g be functions from \mathbb{R}^+ to \mathbb{R}^+ . Then $f = O(g)$ if there are positive constants c and n_0 such that $f(x) \leq cg(x)$ whenever $x > n_0$.

This notation is useful in comparing the asymptotic behavior of the functions f and g , that is, beyond a constant n_0 . If $f = O(g)$ then f can grow at most a constant times as fast as g . For example, we do not have $f = O(g)$ if $f(x) = x^2$ and $g(x) = x$. Indeed, $f(x) = xg(x)$ so there is no constant c such that $f(x) \leq cg(x)$ because we can always choose x larger than c and n_0 and get a contradiction. We rewrite the definition in more formal notation. The statement $f = O(g)$ is equivalent to

$$\exists c > 0 \exists n_0 > 0 \forall x \in \mathbb{R} [x > n_0 \Rightarrow f(x) \leq cg(x)].$$

We can simplify by absorbing the constraint of x being larger than the constant n_0 into the last quantifying statement:

$$\exists c > 0 \exists n_0 > 0 \forall x > n_0 [f(x) \leq cg(x)].$$

We have seen above that negating a quantified statement reverses all quantifiers and pulls the negation into the unquantified, inner statement. Recall that $\neg(p \Rightarrow q)$ is equivalent to $p \wedge \neg q$. Hence, the statement $f \neq O(g)$ is equivalent to

$$\forall c > 0 \forall n_0 > 0 \exists x \in \mathbb{R} [x > n_0 \wedge f(x) > cg(x)].$$

We can again simplify by absorbing the constraint on x into the quantifying statement:

$$\forall c > 0 \forall n_0 > 0 \exists x > n_0 [f(x) > cg(x)].$$

Big-Theta notation. Recall that the big-Oh notation is used to express that one function grows asymptotically at most as fast as another, allowing for a constant factor of difference. The big-Theta notation is stronger and expresses that two functions grow asymptotically at the same speed, again allowing for a constant difference.

DEFINITION. Let f and g be functions from \mathbb{R}^+ to \mathbb{R}^+ . Then $f = \Theta(g)$ if $f = O(g)$ and $g = O(f)$.

Note that in big-Oh notation, we can always increase the constants c and n_0 without changing the truth value of the statement. We can therefore rewrite the big-Theta statement using the larger of the two constants c and the larger of the two constants n_0 . Hence, $f = \Theta(g)$ is equivalent to

$$\exists c > 0 \exists n_0 > 0 \forall x > n_0 [f(x) \leq cg(x) \wedge g(x) \leq cf(x)].$$

Here we can further simplify by rewriting the two inequalities by a single one: $\frac{1}{c}g(x) \leq f(x) \leq cg(x)$. Just for practice, we also write the negation in formal notation. The statement $f \neq \Theta(f)$ is equivalent to

$$\forall c > 0 \forall n_0 > 0 \exists x > n_0 [cg(x) < f(x) \vee cf(x) < g(x)].$$

Because the two inequalities are connected by a logical or, we cannot simply combine them. We could by negating it first, $\neg(\frac{1}{c}g(x) \leq f(x) \leq cg(x))$, but this is hardly easier to read.

Big-Omega notation. Complementary to the big-Oh notation, we have

DEFINITION. Let f and g be functions from \mathbb{R}^+ to \mathbb{R}^+ . Then $f = \Omega(g)$ if $g = O(f)$.

In formal notation, $f = \Omega(g)$ is equivalent to

$$\exists c > 0 \exists n_0 > 0 \forall x > n_0 [f(x) \geq cg(x)].$$

We may think of big-Oh like a less-than-or-equal-to for functions, and big-Omega as the complementary greater-than-or-equal-to. Just as we have $x = y$ iff $x \leq y$ and $x \geq y$, we have $f = \Theta(g)$ iff $f = O(g)$ and $f = \Omega(g)$.

Little-oh and little-omega notation. For completeness, we add notation that corresponds to the strict less-than and greater-than relations.

DEFINITION. Let f and g be functions from \mathbb{R}^+ to \mathbb{R}^+ . Then $f = o(g)$ if for all constants $c > 0$ there exists a constant $n_0 > 0$ such that $f(x) < cg(x)$ whenever $x > n_0$. Furthermore, $f = \omega(g)$ if $g = o(f)$.

This is not equivalent to $f = O(g)$ and $f \neq \Omega(g)$. The reason for this is the existence of functions that cannot be compared at all. Consider for example $f(x) = x^2(\cos x + 1)$. For $x = 2k\pi$, k a non-negative integer, we have $f(x) = 2x^2$, while for $x = (2k + 1)\pi$, we have $f(x) = 0$. Let $g(x) = x$. For even multiples of π , f grows much faster than g , while for odd multiples of π it grows much slower than g , namely not at all. We rewrite the little-Oh notation in formal notation. Specifically, $f = o(g)$ is equivalent to

$$\forall c > 0 \exists n_0 > 0 \forall x > n_0 [f(x) < cg(x)].$$

Similarly, $f = \omega(g)$ is equivalent to

$$\forall c > 0 \exists n_0 > 0 \forall x > n_0 [f(x) > \frac{1}{c}g(x)].$$

In words, no matter how small our positive constant c is, there always exists a constant n_0 such that beyond that constant, $f(x)$ is larger than $g(x)$ over c . Equivalently, no matter how big our constant c is, there always exists a constant n_0 such that beyond that constant, $f(x)$ is larger than c times $g(x)$. We can thus simplify the formal statement by substituting $[f(x) > cg(x)]$ for the inequality.