# Lecture 4: Predicates and Quantifiers

September 3, 2026

Lecturer and scribe: Karthik Gajulapalli

---

## 1 Recap: Propositions and Proofs

A **proposition** is a statement that is either true or false. A **proof** is a chain of logical deductions that begins with axioms and assumptions and ends with the proposition we want to establish. Along the way, we may use definitions and results already proved.

In Lectures 2 and 3, we saw how to combine propositions using logical operations. For example, let

$$P: \text{ Seven is odd.}$$

Then $\neg P$ says "Seven is not odd." Since every integer is exactly one of even or odd, we can also express $\neg P$ as "Seven is even."

| Notation | Meaning |
|---|---|
| $\neg P$ | NOT $P$ |
| $P \wedge Q$ | $P$ AND $Q$ |
| $P \vee Q$ | $P$ OR $Q$ |
| $P \Rightarrow Q$ | If $P$, then $Q$ |
| $P \Leftrightarrow Q$ | $P$ if and only if $Q$ |

To prove $P \Rightarrow Q$, we show that $Q$ follows when we add $P$ to our background assumptions:

$$\text{Axioms} + P \implies Q.$$

To prove $P \Leftrightarrow Q$, we must establish both $P \Rightarrow Q$ and $Q \Rightarrow P$.

### 1.1 Three Ways to Prove an Implication

1. **Direct proof:** assume $P$ and deduce $Q$.
2. **Proof by contrapositive:** assume $\neg Q$ and deduce $\neg P$. This works because $(P \Rightarrow Q) \equiv (\neg Q \Rightarrow \neg P)$.
3. **Proof by contradiction:** assume $P \wedge \neg Q$ and derive an impossibility. This shows that $P$ and $\neg Q$ cannot both hold, since $(P \Rightarrow Q) \equiv \neg(P \wedge \neg Q)$.

For a proposition without an explicit hypothesis, a contradiction proof starts by assuming that the proposition is false. The following example uses this approach.

# 2 Example: Infinitely Many Primes

A **prime** is an integer greater than 1 whose only positive divisors are 1 and itself. We use the fact that every integer greater than 1 has a prime divisor. Indeed, its smallest divisor greater than 1 must be prime; otherwise a smaller divisor greater than 1 would exist.

**Claim 1.** *There are infinitely many prime numbers.*

*Euclid's argument, by contradiction.* Suppose that there are only finitely many primes, and list all of them as

$$p_1, p_2, \ldots, p_r.$$

The list is nonempty because 2 is prime. Form the integer

$$N = p_1 p_2 \cdots p_r + 1.$$

By construction, for each $i$ we can write

$$N = a_i p_i + 1,$$

where $a_i$ is a positive integer. Thus $N$ leaves a remainder of 1 when divided by $p_i$, so none of the listed primes divides $N$. Since the list $p_1, \ldots, p_r$ contains every prime by assumption, this means that $N$ has no prime divisor. But $N > 1$, so it must have a prime divisor. This contradiction shows that there are infinitely many primes. □

The constructed number $N$ need not itself be prime. The argument only requires a *prime divisor* of $N$, which cannot belong to the proposed complete list.

# 3 Predicates: Statements with Inputs

The sentence "2 is even" is a proposition. The expression "$x$ is even" has a truth value only after we know what $x$ represents. It is an example of a **predicate**.

A predicate is a statement with one or more inputs drawn from specified **domains**. Substituting values for all its inputs produces a proposition. For a one-input predicate $R$ on a domain $D$, we can think of $R$ as a rule assigning a truth value to every element:

$$R : D \longrightarrow \{\mathrm{T}, \mathrm{F}\}.$$

For example, take

$$D = \{0, 1, 2, \ldots\}, \qquad \mathrm{Even}(x) : \ x \text{ is even}.$$

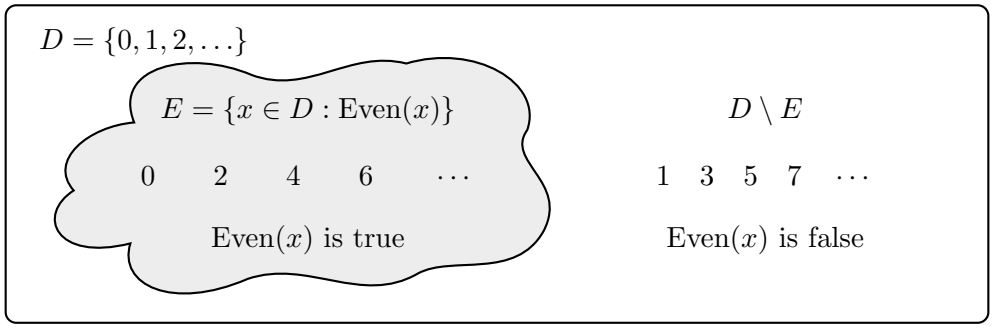
Then Even(2) is true, while Even(3) is false. The domain matters: here our inputs are nonnegative integers.

### 3.1 A Predicate Picks Out a Subset

The inputs for which $\text{Even}(x)$ is true form the subset

$$E = \{x \in D : \text{Even}(x)\} = \{0, 2, 4, 6, \ldots\}.$$

We can draw this subset inside the domain:




Every input lies either in $E$, where the predicate is true, or in $D \setminus E$ (the elements of $D$ that are not in $E$), where it is false. These two regions are disjoint and together cover $D$. In this sense, a predicate divides its domain into a true region and a false region. For some predicates, one of the regions may be empty.

## 4 Quantifiers: "For All" and "There Exists"

Predicates let us talk about an unspecified input. We also want to make claims about *all* inputs, or assert that *at least one* input has a property. **Quantifiers** express these two ideas and turn a predicate with one free variable into a proposition.

A variable is **bound** when it is governed by a quantifier; otherwise it is **free**. For example, in $\forall x\, P(x, y)$, $x$ is bound but $y$ remains free.

### 4.1 The Universal Quantifier

The statement

$$\forall x \in D,\ R(x)$$

means "for every $x$ in $D$, $R(x)$ is true." The symbol $\forall$ is the **universal quantifier**.

To prove a universal statement, choose an *arbitrary* element of the domain and show that the property holds without making any special assumption about that element. To disprove it, one **counterexample** is enough.

**Example: every integer has a nonnegative square.**

$$\forall x \in \mathbb{Z},\ x^2 \geq 0.$$

This proposition is **true**. Let $x$ be an arbitrary integer. If $x \geq 0$, then $x^2 = x \cdot x \geq 0$. If $x < 0$, then $-x > 0$, so $x^2 = (-x)^2 > 0$. Thus $x^2 \geq 0$ in either case, and the claim holds for every integer.

For $D = \{0, 1, 2, \ldots\}$, the proposition

$$\forall x \in D,\ \text{Even}(x)$$

is **false**: $x = 3$ is a counterexample because 3 is not even. Checking several even inputs cannot establish the universal claim.

## 4.2 The Existential Quantifier

The statement

$$\exists x \in D,\ R(x)$$

means "there is at least one $x$ in $D$ for which $R(x)$ is true." The symbol $\exists$ is the **existential quantifier**. It asserts existence, not uniqueness.

For example,

$$\exists x \in D,\ \text{Even}(x)$$

is **true**: choose $x = 2$, and observe that $2 = 2 \cdot 1$ is even. An input that makes the predicate true is called a **witness**. Finding a witness and verifying its property is a **constructive existence proof**.

When the domain is clear, we abbreviate $\forall x \in D$ to $\forall x$ and $\exists x \in D$ to $\exists x$. Parentheses indicate the scope of each quantifier: in $\forall x\,[\text{Even}(x) \Leftrightarrow \neg\,\text{Even}(x+1)]$, the quantifier applies to the whole equivalence.

## 4.3 A Note on Quantifier Order

Read quantifiers from left to right. In $\forall x\,\exists y\,R(x, y)$, the witness $y$ may depend on $x$. In $\exists y\,\forall x\,R(x, y)$, one fixed $y$ must work for every $x$.

For example,

$$\forall x \in \mathbb{Z},\ \exists y \in \mathbb{Z},\ y > x$$

is **true**: after choosing $x$, take $y = x + 1$. However,

$$\exists y \in \mathbb{Z},\ \forall x \in \mathbb{Z},\ y > x$$

is **false**: no integer is greater than every integer. For any proposed $y$, choosing $x = y$ makes $y > x$ false. Thus interchanging $\forall$ and $\exists$ can change the meaning and truth value of a statement.

# 5 Proving Quantified Statements

**Claim 2.** *For $D = \{0, 1, 2, \ldots\}$,*

$$\forall x \in D,\ \big[\text{Even}(x) \Leftrightarrow \neg\,\text{Even}(x+1)\big].$$

*Proof.* Let $x \in D$ be arbitrary. As in Lecture 3, we use the arithmetic fact that each integer is exactly one of even or odd. We prove both directions.

**Forward direction.** Suppose $x$ is even. Then $x = 2k$ for some nonnegative integer $k$, and

$$x + 1 = 2k + 1.$$

Thus $x + 1$ is odd, so $\neg\,\text{Even}(x + 1)$ holds.

**Reverse direction.** Suppose $\neg\,\text{Even}(x + 1)$ holds. Then $x + 1$ is odd, so $x + 1 = 2k + 1$ for some nonnegative integer $k$. Subtracting 1 gives

$$x = 2k,$$

so $\text{Even}(x)$ holds.

We have proved the equivalence for an arbitrary $x \in D$, so it holds for every $x \in D$. □

**Claim 3.** *There exists a nonnegative integer that is both even and a perfect square:*

$$\exists x \in D,\ \big[\text{Even}(x) \wedge (\exists y \in D,\ x = y^2)\big].$$

*Proof.* Choose $x = 4$. It is even because $4 = 2 \cdot 2$, and it is a perfect square because $4 = 2^2$. Thus $x = 4$, with $y = 2$, supplies the required witnesses. □

Unlike a universal proof, this argument may choose a particular input. We need one successful example to prove existence.

# 6 Quantifiers as a Big AND and a Big OR

Suppose first that $D = \{d_1, d_2, \ldots, d_n\}$ is some finite domain. Then

$$\forall x \in D,\ R(x) \equiv R(d_1) \wedge R(d_2) \wedge \cdots \wedge R(d_n),$$
$$\exists x \in D,\ R(x) \equiv R(d_1) \vee R(d_2) \vee \cdots \vee R(d_n).$$

Universal quantification acts like a **big AND**: every instance must be true. Existential quantification acts like a **big OR**: at least one instance must be true. We write this interpretation as

$$\forall x \in D,\ R(x) \ \equiv\ \bigwedge_{x \in D} R(x), \qquad \exists x \in D,\ R(x) \ \equiv\ \bigvee_{x \in D} R(x).$$

For $D = \{0, 1, 2, \ldots\}$, think of

$$\begin{aligned} \forall x \in D,\ \text{Even}(x) \quad &\text{as} \quad \text{Even}(0) \wedge \text{Even}(1) \wedge \text{Even}(2) \wedge \cdots, \\ \exists x \in D,\ \text{Even}(x) \quad &\text{as} \quad \text{Even}(0) \vee \text{Even}(1) \vee \text{Even}(2) \vee \cdots. \end{aligned}$$

For an infinite domain, these are interpretations of what the quantifiers mean, rather than finite formulas we can finish writing out.

## 6.1 De Morgan's Laws for Quantifiers

Recall De Morgan's laws from Boolean algebra:

$$\neg(A \wedge B) \equiv \neg A \vee \neg B, \qquad \neg(A \vee B) \equiv \neg A \wedge \neg B.$$

Applying the same idea to a big AND or a big OR gives

$$\boxed{\neg\big(\forall x \in D,\ R(x)\big) \equiv \exists x \in D,\ \neg R(x)}$$

and

$$\boxed{\neg\big(\exists x \in D,\ R(x)\big) \equiv \forall x \in D,\ \neg R(x).}$$

In words, "not all inputs satisfy $R$" means that some input fails $R$, while "no input satisfies $R$" means that every input fails $R$. These laws hold for both finite and infinite domains.

For example, over the nonnegative integers,

$$\begin{aligned} \neg\big(\forall x\ \text{Even}(x)\big) &\equiv \exists x\, \neg\, \text{Even}(x), \\ \neg\big(\exists x\ \text{Even}(x)\big) &\equiv \forall x\, \neg\, \text{Even}(x). \end{aligned}$$

Both sides of the first equivalence are true, as shown by $x = 3$. Both sides of the second equivalence are false because $x = 2$ is even; the equivalence itself is still valid. "Not every number is even" is different from "No number is even."

## 6.2 Negating Several Quantifiers

Move the negation inward one quantifier at a time, switching $\forall$ and $\exists$ each time:

$$\neg\big(\forall x\, \exists y\, R(x, y)\big) \equiv \exists x\, \neg\big(\exists y\, R(x, y)\big) \equiv \exists x\, \forall y\, \neg R(x, y).$$

The order of the variables stays the same; each quantifier changes type.

# 7 Practice: Translating English into Logic

Two useful translation patterns, for predicates $A$ and $B$ on the same domain, are

$$\begin{aligned} \text{“Every } A \text{ is } B\text{”} \quad &\longrightarrow \quad \forall x\,[A(x) \Rightarrow B(x)], \\ \text{“Some } A \text{ is } B\text{”} \quad &\longrightarrow \quad \exists x\,[A(x) \wedge B(x)]. \end{aligned}$$

The first restricts the claim to objects satisfying $A$, while the second requires a witness satisfying both properties.

For the exercises below, we follow the method and notation of Task 3.1 in van Melkebeek's notes [1]. All variables range over $D$, the set of people:

$$\begin{aligned} F(x):&\ x \text{ is female}, \\ M(x):&\ x \text{ is male}, \\ P(x,y):&\ x \text{ is a parent of } y, \\ W(x,y):&\ x \text{ is married to } y. \end{aligned}$$

## 7.1 Statement 1: Everyone Has a Father and a Mother

For each person $x$, separate the two requirements:

$$\begin{aligned} A(x):&\ x \text{ has a father} \quad \equiv (\exists y)[M(y) \wedge P(y,x)], \\ B(x):&\ x \text{ has a mother} \quad \equiv (\exists z)[F(z) \wedge P(z,x)]. \end{aligned}$$

Combine them as $(\forall x)[A(x) \wedge B(x)]$:

$$\boxed{(\forall x)\Big[\big((\exists y)[M(y) \wedge P(y,x)]\big) \wedge \big((\exists z)[F(z) \wedge P(z,x)]\big)\Big].}$$

With the quantifiers at the front, this becomes

$$(\forall x)(\exists y)(\exists z)[M(y) \wedge P(y,x) \wedge F(z) \wedge P(z,x)].$$

The variable names $y$ and $z$ must remain distinct in this form.

## 7.2 Statement 2: Some Children's Parents Are Not Married

Use $x$ for the child and $y, z$ for the parents. Require

$$\begin{aligned} &P(y,x) && y \text{ is a parent of } x, \\ &P(z,x) && z \text{ is a parent of } x, \\ &\neg W(y,z) && \text{the parents are not married to each other.} \end{aligned}$$

Also require $y \neq z$ to prevent selecting the same parent twice:

$$\boxed{(\exists x)(\exists y)(\exists z)[P(y,x) \wedge P(z,x) \wedge \neg W(y,z) \wedge y \neq z].}$$

### 7.3 Statement 3: Not Every Married Couple Has Children

Choose a married pair $x, y$ and require that neither has children:

$$\begin{array}{ll} W(x,y) & x \text{ and } y \text{ are married,} \\ (\forall z)\neg P(x,z) & x \text{ has no children,} \\ (\forall z)\neg P(y,z) & y \text{ has no children.} \end{array}$$

Joining these conditions gives

$$\boxed{(\exists x)(\exists y)\Big[W(x,y) \wedge [(\forall z)\neg P(x,z)] \wedge [(\forall z)\neg P(y,z)]\Big].}$$

### 7.4 Statement 4: No One Is Married to Their Uncle

We can restate this as follows: for every $x$ and $u$, if $x$ and $u$ are married, then $u$ is not an uncle of $x$.

Use the contrapositive: if $u$ is an uncle of $x$, then $\neg W(x,u)$.

We introduce an intermediate **sibling predicate** $S(p,u)$ that is true exactly when $p$ and $u$ are distinct people with a shared parent $y$:

$$S(p,u) \equiv (\exists y)[P(y,p) \wedge P(y,u) \wedge p \neq u].$$

An uncle must be male and a sibling of a parent, so we join these conditions with AND:

$$(\forall x)(\forall u)\Big[\big(M(u) \wedge (\exists p)[P(p,x) \wedge S(p,u)]\big) \Rightarrow \neg W(x,u)\Big].$$

Expanding $S$ gives

$$\boxed{\begin{array}{l}(\forall x)(\forall u)\Big[\big(M(u) \wedge (\exists p)\big[P(p,x) \wedge \\ \qquad\qquad (\exists y)[P(y,p) \wedge P(y,u) \wedge p \neq u]\big]\big) \Rightarrow \neg W(x,u)\Big].\end{array}}$$

The condition $M(u)$ requires $u$ to be male. The shared parent $y$ is a parent of $p$ and $u$.

## References

[1] Dieter van Melkebeek, instructor; Dalibor Zelený, scribe. *Lecture 3: Predicates and Sets*, CS/Math 240: Introduction to Discrete Mathematics, University of Wisconsin–Madison, draft notes, Task 3.1, pp. 6–7. Lecture notes (PDF).