

True/False Logic

“True/false” logic, also called propositional logic, studies which of the values “True” or “False” a formula built from the logical connectives AND, OR, NOT... can take.

1. True/false, connectives

1.1. True/false

We define two values T (True) and F (False), called **Boolean values** or **truth values**. We will also use the notation 0 (False) and 1 (True).

A **propositional variable** is a variable that can take either the value T (True) or the value F (False). For example, if P is “ $2 + 2 = 4$,” then P is true; in contrast, Q : “ $2^2 < 3$ ” is false.

1.2. $\neg$ (NOT), $\wedge$ (AND), $\vee$ (OR)

We build new formulas from one or more propositional variables and **connectives**. Let P and Q be two propositional variables.

- **Negation** $\neg P$. $\neg P$ is true if and only if P is false.
- **Conjunction** $P \wedge Q$. $P \wedge Q$ is true if and only if P and Q are true.
- **Disjunction** $P \vee Q$. $P \vee Q$ is true if and only if P is true or Q is true.

Let us return to P : “ $2 + 2 = 4$ ” and Q : “ $2^2 < 3$.” Then:

- $\neg P$ is false (because P is true),
- $\neg Q$ is true (because Q is false),
- $P \wedge Q$ is false (because Q is not true),
- $P \vee Q$ is true (because P is true).

Of course, there is a logical problem with our definitions: for example, we defined the connective “ $\wedge$ ” (AND) using the English word “and.” To remedy this problem, we define how each of these connectives acts through its **truth table**:

P	$\neg P$	P	Q	$P \wedge Q$	P	Q	$P \vee Q$
T	F	T	T	T	T	T	T
T	F	T	F	F	T	F	T
F	T	F	T	F	F	T	T
		F	F	F	F	F	F

Of course, we can use these connectives to build more complicated formulas:

$$\neg(P \vee Q) \quad , \quad P \wedge (\neg P) \quad , \quad P \wedge (Q \vee R) \quad , \quad (P \wedge Q) \vee R \quad , \quad \dots$$

Parentheses matter.

1.3. $\rightarrow$ (implication)

The *implication* $P \rightarrow Q$ is defined by:

$$P \rightarrow Q = (\neg P) \vee Q$$

Since the connective $\neg$ takes precedence over the others, we simply write $\neg P \vee Q$.

We read this as “if P , then Q ” or “ P implies Q .” Its truth table is:

P	Q	$P \rightarrow Q$
T	T	T
T	F	F
F	T	T
F	F	T

The traditional mathematical notation for this connective is $P \implies Q$. But over time it has been misused to mean “therefore,” which is not what it means! We will return to this in Section 3.

Thus “ $P \rightarrow Q$ ” means “if P is true, then Q is true.” Here are a few examples:

- “it is raining” $\rightarrow$ “the ground is wet” is a true implication. In this book, we will avoid examples in ordinary language to limit ambiguous wording and endless discussions such as “why would my kitchen floor be wet if it is raining outside?”
- “ $2+2=4$ ” $\rightarrow$ “ $3 \times 3 = 7$ ” is a false implication.
- Let n be an integer. “ n is even” $\rightarrow$ “ n^2 is even” is true.

Let us turn to the surprising aspects of implication.

- “ $2+2=4$ ” $\rightarrow$ “an integer is even or odd” is a true implication (because if P and Q are true, then $P \rightarrow Q$ is true), even though there is no causal link between P and Q .
- “ $3 \times 3 = 7$ ” $\rightarrow$ “ $2+2=4$ ” and “ $3 \times 3 = 7$ ” $\rightarrow$ “ $2+2=5$ ” are also true implications. Indeed, whenever P is false, $P \rightarrow Q$ is true. This is of course clear from the definition of implication using $\neg P \vee Q$. We have to accept this consequence of the definition. We will also return to it in Section 3.

1.4. $\leftrightarrow$ (equivalence)

The *equivalence* $P \leftrightarrow Q$ is defined by the two implications $P \rightarrow Q$ and $Q \rightarrow P$:

$$P \leftrightarrow Q = (P \rightarrow Q) \wedge (Q \rightarrow P)$$

We say “ P is equivalent to Q ” or “ P if and only if Q .”

In other words, $P \leftrightarrow Q$ is true when P and Q are both true or both false.

P	Q	$P \leftrightarrow Q$
T	T	T
T	F	F
F	T	F
F	F	T

Here are examples of true equivalences:

- $\neg\neg P \leftrightarrow P$: negating a negation gives us back the original proposition.
- $P \rightarrow Q \leftrightarrow (\neg P) \vee Q$: by the definition of implication.

In contrast, the following equivalence is *false*:

$$P \wedge Q \leftrightarrow P \vee Q$$

More precisely, this equivalence is not true for every choice of values assigned to P and Q . Indeed, the implication $P \wedge Q \rightarrow P \vee Q$ is always true, but the implication $P \vee Q \rightarrow P \wedge Q$ is false, for example, when P is true and Q is false.

1.5. Summary

There are two Boolean values T/F (True/False, or 1/0). We have defined the five basic logical connectives.

Symbol	Name	Definition
$\neg$	NOT	$\neg P$ is true iff P is false
$\wedge$	AND	$P \wedge Q$ is true iff P and Q are true
$\vee$	OR	$P \vee Q$ is true iff P or Q is true
$\rightarrow$	implication	$\neg P \vee Q$
$\leftrightarrow$	equivalence	$(P \rightarrow Q) \wedge (Q \rightarrow P)$

The connective $\neg$ is a **unary** connective, because it applies to a single parameter (like a function of one variable), whereas the others are **binary** connectives (for example, we could write $P \wedge Q$ as $\wedge(P, Q)$, like a function of two variables).

2. Formulas

2.1. Goals

We have **propositional variables** $P, Q, R, \dots$, each of which can take the value True or False. Using the connectives, we build a *formula* $\mathcal{P}$, for example:

$$\mathcal{P}: P \wedge (Q \vee (\neg R))$$

Here are the questions we ask. If we replace $P, Q, R, \dots$ with values T/F, is the formula $\mathcal{P}$ true or false? We can be even more precise:

- Is there a choice of values T/F that makes $\mathcal{P}$ evaluate to True?
- Does $\mathcal{P}$ evaluate to True for every possible choice of values T/F?

For example, for the proposition $\mathcal{P}: P \wedge (Q \vee (\neg R))$, if we assign the value T to P , F to Q , and F to R , then $\mathcal{P}$ evaluates to $T \wedge (F \vee (\neg F))$, which equals T. In contrast, if we replace (P, Q, R) with (T, F, T), then $\mathcal{P}$ evaluates to F.

Of course, we could ask the same questions about when $\mathcal{P}$ evaluates to False, but that amounts to deciding when the formula $\neg \mathcal{P}$ evaluates to True.

Given two formulas $\mathcal{P}$ and $\mathcal{Q}$, we can build more complex formulas $\mathcal{P} \wedge \mathcal{Q}$, $\mathcal{P} \vee \mathcal{Q}$, etc.

Given two formulas $\mathcal{P}$ and $\mathcal{Q}$, our most important goal will be to understand the relationship between $\mathcal{P}$ and $\mathcal{Q}$. For example, is $\mathcal{P} \rightarrow \mathcal{Q}$ true? Is $\mathcal{P} \leftrightarrow \mathcal{Q}$ true?

2.2. Definition of a formula

Consider an **alphabet** consisting of:

- a set $\text{Var} = \{P, Q, R, \dots\}$ of propositional variables,
- a set $\text{Conn} = \{\neg, \wedge, \vee, \rightarrow, \leftrightarrow\}$ of logical connectives,
- the parenthesis symbols (and).

A **formula** is a finite word made up of characters from this alphabet; it must be correctly parenthesized and use the connectives consistently.

Definition.

More precisely, the set Form of all **formulas** is the smallest set of words containing the propositional variables such that:

- if $\mathcal{P}$ is a formula, then so is $\neg\mathcal{P}$,
- if $\mathcal{P}$ and $\mathcal{Q}$ are formulas, then $(\mathcal{P} \wedge \mathcal{Q})$, $(\mathcal{P} \vee \mathcal{Q})$, $(\mathcal{P} \rightarrow \mathcal{Q})$, $(\mathcal{P} \leftrightarrow \mathcal{Q})$ are also formulas.

Remark: we will use $P, Q, R, \dots$ (plain letters) for propositional variables and $\mathcal{P}, \mathcal{Q}, \mathcal{R}, \dots$ (calligraphic letters) for formulas.

Here are some examples of formulas:

$$P \quad , \quad \neg P \quad , \quad (Q \rightarrow P) \quad , \quad ((Q \rightarrow P) \leftrightarrow Q) \quad , \quad (\neg R \rightarrow (P \vee Q))$$

With some abuse of notation:

- We will allow ourselves to omit the outermost parentheses, for example writing $Q \rightarrow P$ instead of $(Q \rightarrow P)$, or to omit parentheses when there is no ambiguity, as in $P \vee Q \vee R$.
- Conversely, we will sometimes add unnecessary parentheses. For example, we may write $(\neg P) \wedge Q$ instead of $\neg P \wedge Q$ to avoid confusion with $\neg(P \wedge Q)$.

2.3. Truth tables, evaluation

For a formula $\mathcal{P}$ that depends on two propositional variables P and Q , we can determine when the formula takes the value True or False depending on the values assigned to P and Q . We summarize this in a **truth table** for the formula $\mathcal{P}$:

P	Q	$\mathcal{P}$
T	T	T/F ?
T	F	T/F ?
F	T	T/F ?
F	F	T/F ?

More generally, let $\mathcal{P}$ be a formula in the variables $P_1, P_2, \dots, P_n$. An **assignment of truth values** is a choice of T/F for the variable P_1 , a choice of T/F for the variable $P_2, \dots$. This gives an **evaluation** of the formula $\mathcal{P}$, with value T/F.

In practice, we construct the truth table of the formula $\mathcal{P}$.

Example.

Let:

$$\mathcal{P} : \quad (P \wedge Q) \vee (\neg P)$$

We construct the truth table of $\mathcal{P}$ using the intermediate tables for $P \wedge Q$ and $\neg P$.

P	Q	$P \wedge Q$	$\neg P$	$\mathcal{P}$
T	T	T	F	T
T	F	F	F	F
F	T	F	T	T
F	F	F	T	T

In the chapters “Which Connectives Are Useful?” and “Compactness Theorem,” we will present an assignment of truth values as a function $v : \{P_1, \dots, P_n\} \rightarrow \{T, F\}$ that extends to the set of formulas to give an evaluation $v(\mathcal{P})$.

2.4. Logical equivalence

Definition.

Two formulas $\mathcal{P}$ and $\mathcal{Q}$ are **logically equivalent** if they have the same truth table. We write:

$$\mathcal{P} \equiv \mathcal{Q}$$

This means that the formulas $\mathcal{P}$ and $\mathcal{Q}$ evaluate to True at the same time and to False at the same time, whatever the choice of truth values.

There is a direct link with the equivalence connective $\leftrightarrow$. Indeed, $\mathcal{P} \equiv \mathcal{Q}$ means that $(\mathcal{P} \leftrightarrow \mathcal{Q})$ is a tautology (that is, it always evaluates to True); see Section 4.

Example.

1. $\neg\neg P \equiv P$ (double negation)
2. $\neg(P \vee Q) \equiv (\neg P) \wedge (\neg Q)$ (De Morgan’s law)
3. $(P \wedge Q) \rightarrow R \equiv (\neg(P \wedge Q)) \vee R \equiv ((\neg P) \vee (\neg Q)) \vee R$

2.5. Common logical equivalences

idempotence	$\mathcal{P} \wedge \mathcal{P} \equiv \mathcal{P}$	$\mathcal{P} \vee \mathcal{P} \equiv \mathcal{P}$
commutativity	$\mathcal{P} \wedge \mathcal{Q} \equiv \mathcal{Q} \wedge \mathcal{P}$	$\mathcal{P} \vee \mathcal{Q} \equiv \mathcal{Q} \vee \mathcal{P}$
associativity	$(\mathcal{P} \wedge \mathcal{Q}) \wedge \mathcal{R} \equiv \mathcal{P} \wedge (\mathcal{Q} \wedge \mathcal{R})$	$(\mathcal{P} \vee \mathcal{Q}) \vee \mathcal{R} \equiv \mathcal{P} \vee (\mathcal{Q} \vee \mathcal{R})$
distributivity	$\mathcal{P} \wedge (\mathcal{Q} \vee \mathcal{R}) \equiv (\mathcal{P} \wedge \mathcal{Q}) \vee (\mathcal{P} \wedge \mathcal{R})$	$\mathcal{P} \vee (\mathcal{Q} \wedge \mathcal{R}) \equiv (\mathcal{P} \vee \mathcal{Q}) \wedge (\mathcal{P} \vee \mathcal{R})$
absorption	$\mathcal{P} \wedge (\mathcal{P} \vee \mathcal{Q}) \equiv \mathcal{P}$	$\mathcal{P} \vee (\mathcal{P} \wedge \mathcal{Q}) \equiv \mathcal{P}$
De Morgan’s laws	$\neg(\mathcal{P} \wedge \mathcal{Q}) \equiv (\neg \mathcal{P}) \vee (\neg \mathcal{Q})$	$\neg(\mathcal{P} \vee \mathcal{Q}) \equiv (\neg \mathcal{P}) \wedge (\neg \mathcal{Q})$
contrapositive	$\mathcal{P} \rightarrow \mathcal{Q} \equiv (\neg \mathcal{Q}) \rightarrow (\neg \mathcal{P})$	

These are the most familiar ones. There are many others!

3. Connectives (continued)

We offer some clarifications and comments on our favorite connectives.

3.1. True/False, negation

There are exactly two Boolean values $\{T, F\}$ (or $\{0, 1\}$), no more and no fewer! So if a variable P or a formula $\mathcal{P}$ does not evaluate to False, it must be True. And conversely. There is no “maybe” answer. This has the following consequences:

- *Law of excluded middle:*

$$\mathcal{P} \vee \neg \mathcal{P} \quad \text{is always True}$$

The expression “excluded middle” means that there is no choice other than T/F.

- Principle of noncontradiction:

$$\mathcal{P} \wedge \neg \mathcal{P} \quad \text{is always False}$$

- Law of double negation:

$$\neg \neg \mathcal{P} \equiv \mathcal{P}$$

A little later, in Section 4, we will see a new way of writing what is true/false:

$\top$: a formula that always evaluates to True

This symbol denotes a formula that always evaluates to T. It is read as “always True,” “tautology,” or *top*. Its counterpart is the upside-down symbol:

$\perp$: a formula that always evaluates to False

We can therefore write:

$$\mathcal{P} \vee \neg \mathcal{P} \equiv \top \quad \mathcal{P} \wedge \neg \mathcal{P} \equiv \perp$$

What is the difference between $\top$ and T ? T is a value, whereas $\top$ is a formula that always evaluates to T . If we instead identify the Boolean values with $0/1$, then 1 is a constant, whereas $\top$ is the constant function equal to 1 .

3.2. $\wedge$ (AND), $\vee$ (OR)

In our list of logical equivalences, we had:

$$\mathcal{P} \wedge \mathcal{Q} \equiv \mathcal{Q} \wedge \mathcal{P} \quad \mathcal{P} \vee \mathcal{Q} \equiv \mathcal{Q} \vee \mathcal{P} \quad \text{commutativity}$$

and also:

$$(\mathcal{P} \wedge \mathcal{Q}) \wedge \mathcal{R} \equiv \mathcal{P} \wedge (\mathcal{Q} \wedge \mathcal{R}) \quad (\mathcal{P} \vee \mathcal{Q}) \vee \mathcal{R} \equiv \mathcal{P} \vee (\mathcal{Q} \vee \mathcal{R}) \quad \text{associativity}$$

We therefore allow ourselves to write $\mathcal{P} \wedge \mathcal{Q} \wedge \mathcal{R}$ and $\mathcal{P} \vee \mathcal{Q} \vee \mathcal{R}$ without parentheses. Finally:

$$\mathcal{P} \wedge (\mathcal{Q} \vee \mathcal{R}) \equiv (\mathcal{P} \wedge \mathcal{Q}) \vee (\mathcal{P} \wedge \mathcal{R}) \quad \mathcal{P} \vee (\mathcal{Q} \wedge \mathcal{R}) \equiv (\mathcal{P} \vee \mathcal{Q}) \wedge (\mathcal{P} \vee \mathcal{R}) \quad \text{distributivity}$$

The connective $\wedge$ (AND) is analogous to multiplication $\times$, while $\vee$ (OR) is analogous to addition $+$: $a \times b = b \times a$ (commutativity), $(a \times b) \times c = a \times (b \times c)$ (associativity), $a \times (b + c) = a \times b + a \times c$ (distributivity).

De Morgan’s laws can be summed up by the incantations “the negation of an AND is an OR” and “the negation of an OR is an AND”:

$$\neg(\mathcal{P} \wedge \mathcal{Q}) \equiv \neg \mathcal{P} \vee \neg \mathcal{Q} \quad \neg(\mathcal{P} \vee \mathcal{Q}) \equiv \neg \mathcal{P} \wedge \neg \mathcal{Q}$$

3.3. $\rightarrow$ (implication)

By definition:

$$P \rightarrow Q \equiv \neg P \vee Q$$

and its truth table is:

P	Q	$P \rightarrow Q$
T	T	T
T	F	F
F	T	T
F	F	T

There are several points to keep in mind about implications.

No causality.

It is essential to understand that $P \rightarrow Q$ (here with propositional variables) does not indicate a causal link. Instead, it makes an observation: if P is true, is Q also true?

Here is an example that should make all your math teachers scream.

Example.

Let n be an integer:

$$n \text{ is odd} \rightarrow n \text{ is prime}$$

All your teachers will tell you that this implication is false, but that is not true! The confusion comes from our habit of understanding this sentence to mean “every odd number is also a prime number,” which is of course false. But here n is fixed and its value has not been specified, so the implication may be true or false depending on the value of n .

- if $n = 3$, then “ n is odd $\rightarrow n$ is prime” is true, because P : “3 is odd” and Q : “3 is prime” are true.
- if $n = 15$, then “ n is odd $\rightarrow n$ is prime” is false, because P : “15 is odd” is true whereas Q : “15 is prime” is false.
- if $n = 4$, then “ n is odd $\rightarrow n$ is prime” is true! Indeed, P : “4 is odd” is false, so $P \rightarrow Q$ is true.

When P is false, $P \rightarrow Q$ is true!

This is of course clear from the definition $(\neg P) \vee Q$. Let us try to explain why this is natural. Let us ask ourselves: “When do we want P implies Q to be false?” The answer is: “When P is true and Q is false” (it is like finding a counterexample). In every other case, $P \rightarrow Q$ is true, including when P is false.

We also find this in everyday language:

“If you’re right, then I’m the pope.”

Meaning, “You’re wrong.” In the same vein:

“I’ll do it when pigs fly.”

Meaning, “I won’t do it.” The implication “pigs flying $\rightarrow$ action” is always true, with or without the action.

About proofs.

From $P \rightarrow Q$, we cannot deduce that Q is true. What we will see next is called *modus ponens*, and it is the basis of our reasoning:

If P is true and $P \rightarrow Q$ is true, then Q is true.

This is the mistake we make out of habit when we write $P \implies Q$: we implicitly assume that P is true and deduce that Q is true.

Finally, $\mathcal{P} \rightarrow \mathcal{Q} \rightarrow \mathcal{R}$ does not denote a chain of consequences; by definition, it is the formula:

$$\mathcal{P} \rightarrow (\mathcal{Q} \rightarrow \mathcal{R})$$

3.4. $\leftrightarrow$ (equivalence)

For an equivalence to be true, $\mathcal{P}$ and $\mathcal{Q}$ must be true at the same time and false at the same time. By definition:

$$\mathcal{P} \leftrightarrow \mathcal{Q} \equiv (\mathcal{P} \rightarrow \mathcal{Q}) \wedge (\mathcal{Q} \rightarrow \mathcal{P})$$

We can give the same warnings as for implication: writing $\mathcal{P} \leftrightarrow \mathcal{Q}$ does not mean that $\mathcal{P}$ and $\mathcal{Q}$ always evaluate to True, only that they evaluate to True at the same time.

3.5. Other connectives

An *n -ary connective* is a function $f : \{T, F\}^n \rightarrow \{T, F\}$, or, if you prefer, a function $f : \{0, 1\}^n \rightarrow \{0, 1\}$.

- $n = 1$. $\neg$ is a unary connective (a function of one variable). We may also need the identity id , which maps T to T and F to F.
- $n = 2$. $\wedge$ is a binary connective (like a function of two variables). Except that we write $P \wedge Q$ instead of $\wedge(P, Q)$.
- $n = 2$. **Exclusive OR** $\oplus/\text{XOR}$ is another binary connective:
 $P \oplus Q$ is true if exactly one of the variables is true.
- $n = 2$. The connective $\uparrow/\text{NAND}$, also called the Sheffer stroke, is defined as the negation of AND:

$$P \uparrow Q = \neg(P \wedge Q)$$

We will list all binary connectives in the chapter “Which Connectives Are Useful?”

- For any n , we can define an n -ary connective by giving its truth table in terms of the values assigned to the variables $P_1, P_2, \dots, P_n$. This table has 2^n rows. We will see why we do not really need these connectives in that same chapter, “Which Connectives Are Useful?”

4. Formulas (continued)

4.1. Satisfiable formulas

Let $P_1, P_2, \dots, P_n$ be propositional variables. An assignment of truth values v associates with each variable P_i a truth value $v(P_i)$, which is T or F (or a value 0/1). We can then associate with any formula $\mathcal{P}$ a value T/F, denoted by $v(\mathcal{P})$.

Example.

Consider three variables P, Q, R and the formula:

$$\mathcal{P} : (P \wedge \neg Q) \vee (Q \wedge R)$$

Consider the assignment of truth values (T, F, F) (that is, $v(P) = T$, $v(Q) = F$, $v(R) = F$). Substituting these values into $\mathcal{P}$ gives:

$$v(\mathcal{P}) = (T \wedge \neg F) \vee (F \wedge F)$$

so $v(\mathcal{P}) = T$. In contrast, for the assignment (T, T, F), $\mathcal{P}$ evaluates to F.

Definition.

- A formula $\mathcal{P}$ is **satisfied** by an assignment of truth values v if $\mathcal{P}$ evaluates to True for that assignment.
- A formula $\mathcal{P}$ is **satisfiable** if there is an assignment of truth values for which $\mathcal{P}$ evaluates to True.

The formula $\mathcal{P} : (P \wedge \neg Q) \vee (Q \wedge R)$ in the previous example is satisfiable, because we found an assignment of truth values that makes $\mathcal{P}$ true.

4.2. Tautology**Definition.**

- A **tautology** is a formula $\mathcal{P}$ that always evaluates to True, regardless of the assignment of truth values.
- A **contradiction** is a formula $\mathcal{P}$ that always evaluates to False, regardless of the assignment of truth values.

Notation:

tautology : $\top$	contradiction : $\perp$
--------------------	-------------------------

A few remarks:

- Thus a tautology is a formula that is always true (more precisely, one that always evaluates to the value True). The term comes from the Greek *tautos* (the same) and *logos* (speech): it is always the same value (True).
- A contradiction is a formula that is always false.
- However, most formulas are neither tautologies nor contradictions: they evaluate to true or false values depending on the assignment of truth values.
- A formula $\mathcal{P}$ is a contradiction if and only if $\neg \mathcal{P}$ is a tautology. This is why we will focus mainly on tautologies.

Example.

1. $P \vee \neg P$ is a tautology (if P is True, $P \vee \neg P$ is True; if P is False, $P \vee \neg P$ is also True).
2. $P \wedge \neg P$ is a contradiction (it is a formula that is always false, whatever the value assigned to P).
3. $(P \vee Q \vee R) \vee (\neg P \wedge \neg Q \wedge \neg R)$ is a tautology.
4. $(P \wedge Q) \rightarrow Q$ is a tautology.
5. $\neg(P \wedge Q) \leftrightarrow (\neg P) \vee (\neg Q)$ is a tautology.
6. $((P \rightarrow Q) \wedge (Q \rightarrow R)) \rightarrow (P \rightarrow R)$ is a tautology.

Here are examples of how the symbols $\top/\perp$ are used:

$$\mathcal{P} \wedge \top \equiv \mathcal{P} \quad \mathcal{P} \vee \top \equiv \top \quad \mathcal{P} \wedge \perp \equiv \perp \quad \mathcal{P} \vee \perp \equiv \mathcal{P}$$

We can now make the link between logical equivalence ($\equiv$) and equivalence ($\leftrightarrow$) precise:

$$\mathcal{P} \equiv \mathcal{Q} \quad \text{iff} \quad \mathcal{P} \leftrightarrow \mathcal{Q} \text{ is a tautology}$$

If this is the case, that is, if $P \equiv Q$ (or if $P \leftrightarrow Q$ is always true), then P and Q evaluate to True at the same time and to False at the same time, whatever the evaluation.

In the following sections, we will discover three other ways of viewing formulas and their evaluations.

4.3. Logic gates and circuits

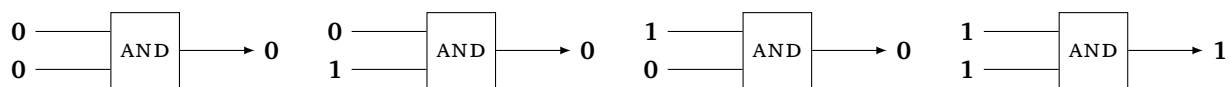
To get closer to what happens inside a computer, we can think of connectives as logic gates. Here the truth values are $\{0, 1\}$.

NOT gate.

It has a single input and changes 0 to 1 and 1 to 0.

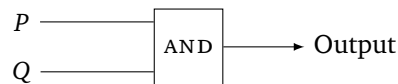


AND gate.

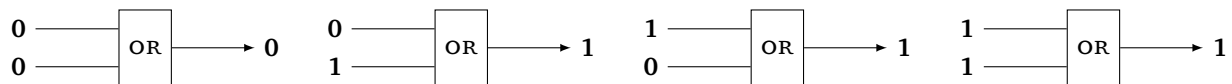


The following table summarizes how the AND gate acts:

P	Q	Output
0	0	0
0	1	0
1	0	0
1	1	1



OR gate.



NAND gate.

It can be implemented as the gate $\text{NOT}(\text{AND})$, that is, an AND gate followed by a NOT gate.

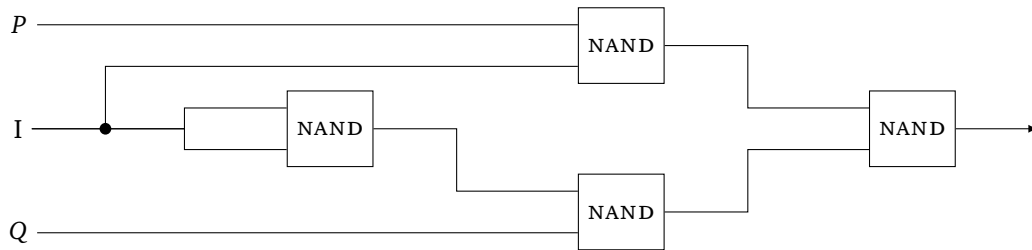


But we prefer to see it as a gate in its own right.



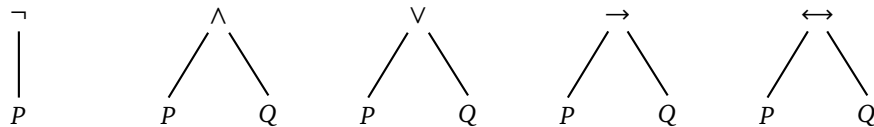
Example.

A *multiplexer* is a circuit that uses a selector I to select a signal from one of two inputs P and Q . If I is 1, the circuit's output is the value of P . If I is 0, the output is the value of Q . We can also view this as a rudimentary test: "If $I = 1$, do this; otherwise, do that."



4.4. Trees

We can also represent connectives as *trees*:

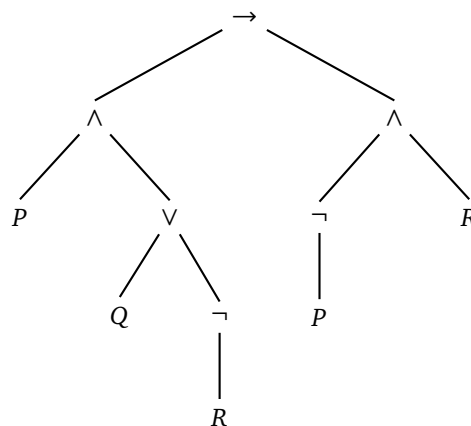


Example.

Consider the formula:

$$(P \wedge (Q \vee \neg R)) \rightarrow (\neg P \wedge R)$$

The corresponding tree is:



The advantage is that a tree structure is well suited to algorithmic processing. A preorder depth-first traversal visits the root, then explores the left subtree, then the right subtree.

We can also adopt *Polish notation*, that is, write $\wedge PQ$ (like a function of two variables $\wedge(P, Q)$) instead of $P \wedge Q$. In Polish notation, the formula in the previous example is written:

$$\rightarrow \wedge P \vee Q \neg R \wedge \neg P R$$

This expression is particularly well suited to computer processing, because it requires no parentheses. For a human, it is easier to read when written as follows:

$$\rightarrow (\wedge (P)(\vee (Q)(\neg R)))(\wedge (\neg P)(R))$$

4.5. Algebraic formulas

We denote the Boolean values by $\mathbb{B} = \{0, 1\}$. In this section, we calculate modulo 2, that is, we write $1 + 1 = 0$ (strictly speaking, we should write $1 + 1 \equiv 0 \pmod{2}$). Note that modulo 2, we have $+1 = -1$. Let $x, y \in \{0, 1\}$. Here are the algebraic expressions that reproduce the action of the connectives:

$$\begin{aligned}\neg x &= 1 + x \\ x \wedge y &= x \times y \\ x \vee y &= x + y - xy \quad \text{inclusive OR} \\ x \oplus y &= x + y \quad \text{exclusive OR}\end{aligned}$$

This is easy to justify. For example, if $x = 0$, then $1 + x = 1$, and if $x = 1$, then $1 + x = 1 + 1 = 0$. So $1 + x$ does indeed correspond to $\neg x$.

Similarly, $x \times y$ can equal 1 only if $x = 1$ and $y = 1$.

Example.

These algebraic expressions can provide proofs. For example, double negation is justified by:

$$\neg\neg x = \neg(1 + x) = 1 + (1 + x) = 1 + 1 + x = x$$

Here is a proof of De Morgan's law for the negation of an AND, $\neg(x \wedge y) = (\neg x) \vee (\neg y)$:

$$(\neg x) \vee (\neg y) = (1 + x) \vee (1 + y) = (1 + x) + (1 + y) - (1 + x)(1 + y) = x + y - 1 - x - y - xy = 1 + xy = \neg(xy)$$

We used $-1 = +1$, so $-xy = xy$.

Exercises

True/false, connectives

Exercise 1. Let P be " $2 \times 2 = 5$ " and Q be " $4 + 4 > 7$." State whether the following propositions are true or false: $\neg P$, $\neg Q$, $P \wedge Q$, $P \vee Q$, $P \rightarrow Q$, $Q \rightarrow P$, $P \leftrightarrow Q$.

Exercise 2. Assume that P is true. For which values of Q are the following propositions true: $P \wedge Q$, $P \vee Q$, $P \rightarrow Q$, $Q \rightarrow P$, $P \leftrightarrow Q$? Answer the same question assuming this time that P is false.

Formulas

Exercise 3. Which of the following words are formulas (in the strict sense of the definition)?

$$\neg\neg P \quad (\neg P \vee Q \wedge R) \quad (P \rightarrow (Q \wedge (R \wedge S))) \quad (P, Q \rightarrow R) \quad ((P \wedge \neg Q \rightarrow)(Q \vee R))$$

Exercise 4. Construct the truth table for $\neg(P \vee Q)$ and the one for $(\neg P) \wedge (\neg Q)$. Deduce De Morgan's law for $\vee$.

Exercise 5. Prove the following associativity and distributivity formulas:

$$P \vee (Q \vee R) \equiv (P \vee Q) \vee R \quad P \wedge (Q \vee R) \equiv (P \wedge Q) \vee (P \wedge R)$$

Exercise 6. Show the following logical equivalences:

$$P \rightarrow Q \wedge R \equiv (P \rightarrow Q) \wedge (P \rightarrow R) \quad P \vee Q \rightarrow R \equiv (P \rightarrow R) \wedge (Q \rightarrow R) \quad P \leftrightarrow Q \equiv (P \vee Q) \rightarrow (P \wedge Q)$$

Express these equivalences in a sentence in English.

Connectives (continued)

Exercise 7. Show that the following formulas are tautologies (they are always true, whatever the values of P and Q):

$$P \vee (P \rightarrow Q) \quad (P \wedge Q) \rightarrow P \quad (P \rightarrow Q) \vee (P \rightarrow \neg Q) \quad (P \rightarrow Q) \vee (\neg P \rightarrow Q)$$

Exercise 8. For each of the following formulas, find a logically equivalent formula whose expression is as simple as possible:

$$Q \vee (P \wedge Q) \quad P \wedge (P \vee Q) \quad P \wedge (\neg P \vee Q) \quad (P \vee Q) \rightarrow (P \wedge Q)$$

Exercise 9. Let $x \in \mathbb{R}$, let P be “ $x \geq 2$,” and let Q be “ $x^2 \geq 4$.” Find all values $x \in \mathbb{R}$ for which $P \rightarrow Q$ is true. Answer the same question for $Q \rightarrow P$, then for $P \leftrightarrow Q$.

Exercise 10.

1. Determine the truth table of $P \rightarrow (Q \rightarrow R)$, then that of $(P \rightarrow Q) \rightarrow R$. What do you conclude?
2. Show that $P \leftrightarrow (Q \leftrightarrow R) \equiv (P \leftrightarrow Q) \leftrightarrow R$.
3. Show that $P \rightarrow (Q \rightarrow R) \equiv (P \wedge Q) \rightarrow R \equiv Q \rightarrow (P \rightarrow R)$.

Exercise 11. Recall that $P \oplus Q$ denotes “exclusive OR” (it is true when P is true or Q is true, but not both).

1. Calculate the truth table of $P \oplus Q$.
2. Show that $P \oplus Q$ amounts to “ $P \neq Q$,” that is, $P \oplus Q \equiv (P \wedge \neg Q) \vee (\neg P \wedge Q)$. What is the relationship between $\oplus$ and $\leftrightarrow$?
3. Show that $\oplus$ is commutative and associative.
4. Express $P \wedge (Q \oplus R)$ using only the connectives $\wedge$, $\vee$, and $\neg$. Do the same for $P \oplus (Q \wedge R)$.

Exercise 12. Consider $P_1, P_2, \dots, P_n$, with $n \geq 1$. When is $P_1 \oplus P_2 \oplus \dots \oplus P_n$ true?

Formulas (continued)

Exercise 13. Are the following formulas satisfiable? Are they tautologies?

$$P \wedge Q \rightarrow P \quad (P \vee Q) \wedge (\neg P \vee \neg Q) \quad P \vee (Q \wedge \neg Q) \leftrightarrow P \quad (P \rightarrow Q) \wedge (Q \rightarrow \neg P) \\ ((P \rightarrow Q) \wedge (R \rightarrow S)) \rightarrow ((P \vee R) \rightarrow (Q \vee S))$$

Exercise 14. Show that the following formulas are tautologies:

1. $P \wedge (P \rightarrow Q) \rightarrow Q$ (*modus ponens*)
2. $(P \rightarrow Q) \wedge \neg Q \rightarrow \neg P$ (*modus tollens*)
3. $(P \rightarrow Q) \leftrightarrow (\neg Q \rightarrow \neg P)$ (*contraposition*)
4. $(P \rightarrow Q) \leftrightarrow ((P \wedge \neg Q) \rightarrow \perp)$ (*proof by contradiction*)
5. $((P \rightarrow Q) \wedge (Q \rightarrow R)) \rightarrow (P \rightarrow R)$ (*sylogism/transitivity of implication*).
6. $(P \wedge \top) \leftrightarrow P$

Exercise 15.

1. Construct the circuit for $P \wedge (Q \vee R)$ and the one for $(P \wedge Q) \vee (P \wedge R)$. Check that they produce the same outputs.
2. Show that using only NAND gates, we can build circuits whose outputs are identical to the action of $\neg$, $\wedge$, and $\vee$.

Exercise 16. For each of the following formulas, construct the corresponding tree and give its expression in Polish notation:

$$\neg P \wedge (Q \vee (P \wedge R)) \qquad ((P \vee Q) \wedge \neg R) \rightarrow (P \wedge (\neg Q \wedge R))$$

Exercise 17.

1. Justify the algebraic formula $x \vee y = x + y - xy$.
2. Using the algebraic formulas, show:

$$x \wedge (y \wedge z) = (x \wedge y) \wedge z \qquad x \vee (y \vee z) = (x \vee y) \vee z \qquad x \wedge (y \vee z) = (x \wedge y) \vee (x \wedge z)$$

3. Calculate $x \wedge (y \oplus z)$ and $(x \wedge y) \oplus (x \wedge z)$. What do you conclude?
4. Find an algebraic formula for $x \rightarrow y$. Do the same for $x \leftrightarrow y$.
5. Using the algebraic formulas, show that $x \rightarrow y = \neg y \rightarrow \neg x$.