2 Propositional Logic

2.1 Truth-Functional Connectives

Propositional logic is the simplest and the most basic logical system there is. As its logical constants it has *connectives* and *negation;* the former link two sentences together into one new, composite sentence, and the latter operates on just one sentence. The restriction to indicative sentences mentioned in §1.2, that is, to sentences which are either true or false, suggests a class of connectives it seems natural to begin with. In order to make clear what these are, we must first introduce the concept of *truth value.* We say that the truth value of a sentence is 1 if the sentence is true and 0 if the sentence is false. Here we are dealing only with sentences which are true or false, so their truth value is either 1 or 0. The principle of compositionality (§1.2, §1.4) requires that the meaning (and thus the truth value) of a composite sentence depends only on the meanings (truth values) of the sentences of which it is composed.

By way of illustration, consider the following sentences:

(1) John has bumped his head and he is crying.

(2) John is crying because he has bumped his head.

(3) John is crying.

(4) John has bumped his head.

Let us suppose that John has in fact bumped his head and that he is indeed crying. So (1) is true. Now note that instead of (3), *John is crying* (we assume that this is what is meant by *he is crying*) we might just as well have written any other true sentence, such as, for example, *it is raining* (if this in fact happens to be the case). Then the sentence *John has bumped his head and it is raining* would also be true. It is quite different for (2): if John has in fact bumped his head and is indeed crying then (2) may well be true, but it certainly does not need to be (maybe he is crying because Mary doesn't love him); and conversely, if (2) is true then *John is crying because it is raining* is false even if it is raining.

This difference in the behavior of *and* and *because* can be put as follows. Sentence (1) is true if both (3) and (4) are true, and false if either is false. The truth value of an *and* sentence depends only on the truth values of the two parts of which it is composed. But this does not apply to sentence (2), whose

truth depends on more than just the truth of sentences (3) and (4) of which it is composed. Connectives which give rise to sentences whose truth value depends only on the truth values of the connected sentences are said to be *truth-functional.* So *and* is, and *because* is not, a truth-functional connective.

Since we here restrict the meaning of sentences to their truth values, compositionality requires that we consider only truth-functional connectives and the corresponding conjunctions from natural language. And given that we only consider such connectives, investigating the way the truth values of sentences depend on each other, and in particular investigating the validity of schemata of reasoning in which connectives figure, becomes very simple. In order to determine the truth value of a sentence A, we need only pay attention to the sentences of which A is ultimately composed. This narrowing down of the meaning of sentences may seem rather rigorous at first, but in practice the restriction has turned out to be quite productive.

2.2 Connectives and Truth Tables

As its logical constants, the vocabulary of a language for propositional logic includes *connectives*. And as logical variables there are symbols which stand for statements (that is, 'propositions'). These symbols are called *propositional letters,* or *propositional variables.* In general we shall designate them by the letters p, q, and r, where necessary with subscripts as in p_1, r_2, q_3, etc. It is usual to use different letters for different propositional symbols. The propositional letters and the composite expressions which are formed from them by means of the connectives are grouped together as *sentences* or *formulas*. We designate these by means of the letters ϕ and ψ, etc. For these metavariables, unlike the variables p, q, and r, there is no convention that different letters must designate different formulas.

Table 2.1 sums up the connectives that we shall encounter in the propositional languages in this chapter, each with an example of a sentence formed by means of it and the meaning of the sentence. The connectives $\wedge$, $\vee$, $\rightarrow$, and $\leftrightarrow$ are said to be two-place, and $\neg$ is said to be one-place; this corresponds to the number of sentences which the connective in question requires;

Table 2.1 Connectives and Their Meanings

Connective	*Composite sentence with this connective*	*Meaning*
$\neg$ *(negation symbol)*	$\neg p$ *(negation of p)*	*it is not the case that p*
$\wedge$ (conjunction symbol)	$(p \wedge q)$ (conjunction of p and q)	p and q
$\vee$ (disjunction symbol)	$(p \vee q)$ (disjunction of p and q)	p and/or q
$\rightarrow$ (implication symbol)	$(p \rightarrow q)$ ((material) implication of p and q)	if p, then q
$\leftrightarrow$ (equivalence symbol)	$(p \leftrightarrow q)$ ((material) equivalence of p and q)	p if and only if q

p is said to be the first member (or *conjunct*) of the conjunction $(p \wedge q)$, and q is the second. The same applies to implications, disjunctions, and equivalences, though the first and second members of an implication are sometimes referred to as its *antecedent* and its *consequent*, respectively, while the two members of a disjunction are referred to as its *disjuncts*.

Our choice of connectives is in a certain sense arbitrary. Some are obviously important. We shall discuss all five separately and consider the extent to which the corresponding expressions from natural language can be regarded as truth-functional. We shall also discuss a few other possible connectives which have not been included in this list.

The syntactic rules of propositional languages allow us to link up, by means of connectives, not only propositional letters (which are also referred to as *atomic* formulas), but also composite formulas. The terminology is just the same: if ϕ and ψ are formulas, then $\neg\phi$ is said to be the negation of ϕ, $(\phi \wedge \psi)$ is the conjunction of ϕ and ψ, etc.; $\neg\phi$ refers naturally enough to the string of symbols obtained by prefixing the string ϕ with a $\neg$; $(\phi \wedge \psi)$ refers to the string of symbols consisting of a left bracket, followed by the string ϕ, followed by the connective $\wedge$, followed by the string ψ, and closing with a right bracket.

The brackets serve to remove any ambiguities. Otherwise a sentence like $p \vee q \wedge r$ could have either of two different meanings. It could be (a) the disjunction of p, on the one hand, and the conjunction of q and r on the other; or (b) the conjunction of the disjunction of p and q, on the one hand, and r on the other. That these have distinct meanings can easily be seen from examples like (5) and (6):

(5) McX has been elected, or Wyman has been elected and a new era has begun.

(6) McX has been elected or Wyman has been elected, and a new era has begun.

Example (5) corresponds to $(p \vee (q \wedge r))$ and as a whole is a disjunction, while (6) corresponds to $((p \vee q) \wedge r)$ and as a whole is a conjunction. We shall return to these more complex formulas in §2.3. Now we shall expand upon the different meanings of the various connectives.

What concerns us is the way in which the truth value of a composite sentence formed from one or two simpler sentences depends on the truth values of the constituent sentences and the connective used. For each connective, this is prescribed in a *truth table*. The discussion of sentence (1) shows that the truth table for the conjunction is as in (7):

(7)

ϕ	ψ	$(\phi \wedge \psi)$
1	1	1
1	0	0
0	1	0

Beside each possible combination of the truth values of ϕ and ψ we see the resulting truth value of $(\phi \wedge \psi)$. On the face of it, it might seem that the logical behavior of $\wedge$ is wholly in accordance with that of *and* in natural language. The agreement, however, is not perfect. If someone truthfully states (8), then according to the truth table sentence (9) is also true.

(8) Annie took off her socks and climbed into bed.

(9) Annie climbed into bed and took off her socks.

But it is very likely that the person in question would not be inclined to accept this, since placing one sentence after the other suggests that this was also the order in which the described events happened. Similar complications arise with all the other connectives. In chapter 6 we shall discuss whether this kind of phenomenon can be explained in terms of *conditions for usage*.

The left column of (10) is a list of sentences which have the same truth conditions as the conjunction of the two sentences to their right.

(10)

Zandvoort and Haarlem lie west of Amsterdam.	Zandvoort lies west of Amsterdam.	Haarlem lies west of Amsterdam.
John and Peter are married to Anne, and Betty, respectively.	John is married to Anne.	Peter is married to Betty.
Both the Liberals and the Socialists favored the motion.	The Liberals favored the motion.	The Socialists favored the motion.
John is at home but he is asleep.	John is at home.	John is asleep.
John is at home but Peter is not.	John is at home.	Peter is not at home.
Although it was extremely cold, John did not stay indoors.	It was extremely cold.	John did not stay indoors.
Even though it was beautiful out of doors, John stayed indoors.	It was beautiful out of doors.	John stayed indoors.

So the sentences in the left column all express a logical conjunction, although from a strictly linguistic point of view we are not dealing with two sentences linked by placing an *and* between them. Apparently the connotations which *but, although,* and *though* have do not alter the truth conditions of the sentences in which the words occur. Note also that not every sentence in which the word *and* figures is a conjunction. Here is an example that is not:

(11) John and Peter are friends.

It seems rather unnatural to regard this as a conjunction, say, of *John is friends with Peter* and *Peter is friends with John*. And sentence (12) does not mean the same as the conjunction (13):

(12) Cheech and Chong are fun at parties.

(13) Cheech is fun at parties and Chong is fun at parties.

Perhaps they are only fun when they're together.

Negation is also a relatively simple matter. The truth table consists of just two rows; see (14):

(14)

ϕ	$\neg\phi$
1	0
0	1

There are more ways to express the negation of a sentence than by means of *not* or *it is not the case that*. See, for example, (15):

(15)

Porcupines are unfriendly.	Porcupines are friendly.
John is neither at home nor at school.	John is either at home or at school.
No one is at home.	Someone is at home.
John is never at home.	John is sometimes at home.
John is not home yet.	John is home already.
John has never yet been at home.	John has on occasion been at home.

For the disjunction we give the truth table in (16):

(16)

ϕ	ψ	$\phi \vee \psi$
1	1	1
1	0	1
0	1	1
0	0	0

This is the obvious truth table for *and/or*, and in natural language *or* generally means *and/or*. This usage is said to be *inclusive*. If *or* is used in such a way as to exclude the possibility that both disjuncts are true—this is also expressed by an *either . . . or . . .* construction—then the *or* is said to be *exclusive*. Sometimes a separate connective ∞ is introduced for the exclusive disjunction, the truth table for which is given in figure (17):

(17)

ϕ	ψ	$\phi \infty \psi$
1	1	0
1	0	1
0	1	1
0	0	0

In this book, *or* is understood to be inclusive unless stated otherwise, as is also usual in mathematics.

Actually it is not very easy to find a natural example of an exclusive *or*. A sentence like (18) will not do:

(18) It is raining or it isn't raining.

In sentence (18) there would be no difference in the truth value whether the *or* were inclusive or not, since it cannot both rain and not rain. What we need is an example of the form *A or B* in which there is a real possibility that both *A* and *B* hold; this eventuality is excluded by the exclusive disjunction. In natural language, this is usually expressed by placing extra emphasis on the *or*, or by means of *either . . . or*. For example:

(19) Either we are going to see a film tonight, or we are going to the beach this afternoon.

Another construction which can be used to express an exclusive disjunction is that with *unless*. Sentence (19) has the same truth conditions as (20):

(20) We are going to see a film tonight, unless we are going to the beach this afternoon.

The truth table for (material) implication is given in figure (21):

(21)

ϕ	ψ	$\phi \rightarrow \psi$
1	1	1
1	0	0
0	1	1
0	0	1

In everyday language, *if* (. . . , *then*) can usually not be considered truth-functional. First, a sentence like

(22) If John bumps his head, he cries.

usually means that at any given moment it is the case that John cries if he has just bumped his head. If (22) is interpreted as

(23) If John has just bumped his head then he is now crying.

then it is clearly true if John has just bumped his head and is in fact crying, and it is false if he has just bumped his head and is at the moment not crying. But what if John has not just bumped his head? One certainly would not wish to say that the sentence must always be false in that case, but it also doesn't seem very attractive to say that it must always be true. Since we have agreed that indicative sentences are either true or false, let us choose the least unattractive alternative and say that material conditionals are true if their antecedent is untrue. What we then get is just (21).

That the implications which one encounters in mathematics are material can be illustrated as follows. Sentence (24) is taken to be true and can for the sake of clarity also be rendered as (25):

(24) If a number is larger than 5, then it is larger than 3.

(25) For all numbers x, if $x > 5$, then $x > 3$.

The truth of a universal statement such as (25) implies the truth of each of its instantiations, in this case, for example, (26):

(26) If $6 > 5$, then $6 > 3$.
If $4 > 5$, then $4 > 3$.
If $2 > 5$, then $2 > 3$.

Now these three combinations correspond precisely to the three different combinations of truth values such that $\phi \rightarrow \psi$ is true: $6 > 5$ and $6 > 3$ are both true, $4 > 5$ is false, and $4 > 3$ is true, while $2 > 5$ and $2 > 3$ are both untrue. Assuming we want a truth table for material implication, the one we have chosen is apparently the only real choice we had. Similar points can be made with regard to sentence (22). If (22) is taken to mean that John always cries if he has just bumped his head, then one must, assuming one accepts that (22) is true, accept that at any given point t in time there are just three possibilities:

(i) At time t John has (just) bumped his head and he is crying.
(ii) At time t John has not (just) bumped his head and he is crying.
(iii) At time t John has not (just) bumped his head and he is not crying.

The eventuality that John has (just) bumped his head and is not crying is ruled out by the truth of (22). From this it should be clear that material implication has at least some role to play in the analysis of implications occurring in natural language. Various other forms of implication have been investigated in logic, for example, in intensional logic (see vol. 2). A number of sentences which can be regarded as implications are to be found in (27):

(27)

$\rightarrow$ *(implication)*	*p (antecedent)*	*q (consequent)*
John cries if he has bumped his head.	John has bumped his head.	John cries.
John is in a bad mood only if he has just gotten up.	John is in a bad mood.	John has just gotten up.
In order for the party to function better, it is necessary that more contact be made with the electorate.	The party functions better.	More contact is made with the electorate.
In order for the party to function better, it is sufficient that Smith be ousted.	Smith is ousted.	The party functions better.

The truth table for material equivalence is given in figure (28):

(28)

ϕ	ψ	$\phi \leftrightarrow \psi$
1	1	1
1	0	0
0	1	0
0	0	1

It can be seen that $\phi \leftrightarrow \psi$ is true if ϕ and ψ both have the same truth values, and false if their truth values differ. Another way of saying this is that $\phi \leftrightarrow \psi$ is true just in case ϕ materially implies ψ while ψ also materially implies ϕ. *If and only if* is a very rare conjunction in natural language. A much more common one, which arguably has the same truth table, is *provided:*

(29) We are going to see a film tonight, provided the dishes have been done.

In mathematical contexts, $\Leftrightarrow$ and *iff* are commonly written for *if and only if.*

2.3 Formulas

Having come this far, we can now capture the concepts we introduced above in precise definitions.

A *language L for propositional logic* has its own reservoir of propositional letters. We shall not specify these; we shall just agree to refer to them by means of the metavariables p, q, and r, if necessary with subscripts appended. Then there are the brackets and connectives ($\neg$, $\wedge$, $\vee$, $\rightarrow$, $\leftrightarrow$) which are common to all languages for propositional logic. Together these form the vocabulary of L. In the syntax we define what is meant by the *well-formed expressions* (*formulas, sentences*) in L. The definition is the same for all propositional languages.

Definition 1

(i) Propositional letters in the vocabulary of L are formulas in L.
(ii) If ψ is a formula in L, then $\neg\psi$ is too.
(iii) If ϕ and ψ are formulas in L, then $(\phi \wedge \psi)$, $(\phi \vee \psi)$, $(\phi \rightarrow \psi)$, and $(\phi \leftrightarrow \psi)$ are too.
(iv) Only that which can be generated by the clauses (i)–(iii) in a finite number of steps is a formula in L.

The first three clauses of the definition give a recipe for preparing formulas; (iv) adds that only that which has been prepared according to the recipe is a formula.

We illustrate the definition by examining a few examples of strings of symbols which this definition declares are well-formed, and a few examples of strings which cannot be considered well-formed. According to definition 1, p,

$\neg\neg\neg\neg p$, $((\neg p \wedge q) \wedge r)$, and $((\neg(p \vee q) \rightarrow \neg\neg\neg q) \leftrightarrow r)$ are examples of formulas, while pq, $\neg(\neg\neg p)$, $\wedge\, p \neg q$, and $\neg((p \rightarrow q \vee r))$ are not.

That p is a formula follows from clause (i), which states that all propositional letters of L are formulas of L. And $\neg\neg\neg\neg p$ is a formula on the basis of (i) and (ii): according to (i), p is a formula, and (ii) allows us to form a new formula from an existing one by prefixing the negation symbol, an operation which has been applied here four times in a row. In $((\neg p \wedge q) \wedge r)$, clause (iii) has been applied twice: it forms a new formula from two existing ones by first introducing an opening, or left, bracket, then the first formula, followed by the conjunction sign and the second formula, and ending with a closing, or right, bracket. In forming $((\neg p \wedge q) \wedge r)$, the operation has been applied first to $\neg p$ and q, which results in $(\neg p \wedge q)$, and then to this result and r. Forming disjunctions, implications, and equivalences also involves the introduction of brackets. This is evident from the fourth example, $((\neg(p \vee q) \rightarrow \neg\neg\neg q) \leftrightarrow r)$, in which the outermost brackets are the result of forming the equivalence of $(\neg(p \vee q) \rightarrow \neg\neg\neg q)$ and r; the innermost are introduced by the construction of the disjunction of p and q; the middle ones result from the introduction of the implication sign. Note that forming the negation does not involve the introduction of brackets. It is not necessary, since no confusion can arise as to what part of a formula a negation sign applies to: either it is prefixed to a propositional letter, or to a formula that begins with a negation sign, or it stands in front of a formula, in whose construction clause (iii) was the last to be applied. In that case the brackets introduced by (iii) make it unambiguously clear what the negation sign applies to.

That pq, i.e., the proposition letter p immediately followed by the proposition letter q, is not a formula is clear: the only way to have two propositional letters together make up a formula is by forming their conjunction, disjunction, implication, or equivalence. The string $\neg(\neg\neg p)$ does not qualify, because brackets occur in it, but no conjunction, disjunction, implication, or equivalence sign, and these are the only ones that introduce brackets. Of course, $\neg\neg\neg p$ *is* well-formed. In $\wedge\, p \neg q$, the conjunction sign appears before the conjuncts, and not, as clause (iii) prescribes, between them. Also, the brackets are missing. In $\neg((p \rightarrow q \vee r))$, finally, the brackets are misplaced, the result being ambiguous between $\neg(p \rightarrow (q \vee r))$ and $\neg((p \rightarrow q) \vee r)$.

Exercise 1

For each of the following expressions, determine whether it is a formula of propositional logic.

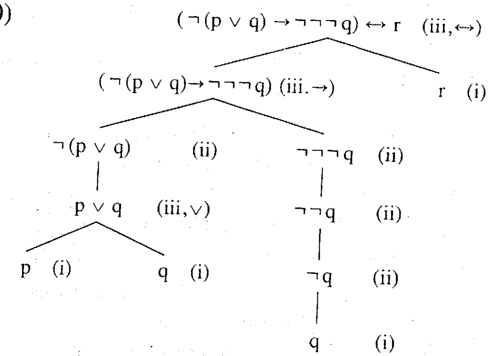
(i) $\neg(\neg p \vee q)$
(ii) $p \vee (q)$
(iii) $\neg(q)$
(iv) $(p_2 \rightarrow (p_2 \rightarrow (p_2 \rightarrow p_2)))$
(v) $(p \rightarrow ((p \rightarrow q)))$
(vi) $((p \rightarrow p) \rightarrow (q \rightarrow q))$

(vii) $((p_{28} \rightarrow p_3) \rightarrow p_4)$
(viii) $(p \rightarrow (p \rightarrow q) \rightarrow q)$
(ix) $(p \vee (q \vee r))$
(x) $(p \vee q \vee r)$
(xi) $(\neg p \vee \neg\neg p)$
(xii) $(p \vee p)$

Leaving off the outer brackets of formulas makes them easier to read and does not carry any danger of ambiguity. So in most of what follows, we prefer to abbreviate $((\neg p \wedge q) \wedge r)$ as $(\neg p \wedge q) \wedge r$, $((\neg(p \vee q) \rightarrow \neg\neg\neg q) \leftrightarrow r)$ as $(\neg(p \vee q) \rightarrow \neg\neg\neg q) \leftrightarrow r$, $((p \wedge q) \wedge (q \wedge p))$ as $(p \wedge q) \wedge (q \wedge p)$, $(p \rightarrow q)$ as $p \rightarrow q$, and $(\neg p \rightarrow q)$ as $\neg p \rightarrow q$. Analogously, we shall write $\phi \wedge \psi$, $\phi \vee \psi$, $\phi \rightarrow \psi$, $\phi \leftrightarrow \psi$, $\phi \wedge (\phi \vee \chi)$, etc.

Definition 1 enables us to associate a unique *construction tree* with each formula. $(\neg(p \vee q) \rightarrow \neg\neg\neg q) \leftrightarrow r$, for example, must have been constructed according to the tree given in figure (30).

(30)



That each formula has a unique construction tree is due to the fact that, because of the brackets, logical formulas are unambiguous. Beside each *node* in the tree we see the number of the clause from definition 1 according to which the formula at that node is a formula. A formula obtained by applying clause (ii) is said to be a negation, and $\neg$ is said to be its *main sign;* similarly, the main sign of a formula obtained by clause (iii) is the connective thereby introduced (in the example, it is written next to the formula). The main sign of the formula at the top of the tree is, for example, $\leftrightarrow$, and the formula is an equivalence. Note that the formulas at the lowest nodes are all atomic.

A formula ϕ appearing in the construction tree of ψ is said to be a *subformula* of ψ. The subformulas of $\neg(p \vee q)$ are thus: p, q, $p \vee q$, and $\neg(p \vee q)$, while the subformulas of $(\neg(p \vee q) \rightarrow \neg\neg\neg q) \leftrightarrow r$ are: p, q, r, $\neg q$, $\neg\neg q$, $\neg\neg\neg q$, $p \vee q$, $\neg(p \vee q)$, $\neg(p \vee q) \rightarrow \neg\neg\neg q$, and $(\neg(p \vee q) \rightarrow \neg\neg\neg q) \leftrightarrow r$. Any subformula ϕ of ψ is a string of consecutive symbols occurring in the string of symbols ψ, which is itself a formula. And conversely, it can be shown that any string of consecutive symbols taken from ψ which is itself a formula is a subformula of ψ. The proof will be omitted here.

Exercise 2

(a) Draw the construction trees of $(p_1 \leftrightarrow p_2) \vee \neg p_2$ and $p_1 \leftrightarrow (p_2 \vee \neg p_2)$ and of $((p \vee q) \vee \neg r) \leftrightarrow (p \vee (q \vee \neg r))$. In each of the three cases give the subformulas of the formula under consideration.

(b) Give all formulas that can be made out of the following sequence of symbols by supplying brackets: $p \wedge \neg q \rightarrow r$. Also supply their construction trees.

(c) Classify each of the following sentences as an atomic formula, a negation, a conjunction, a disjunction, an implication, or an equivalence.

(i)	$p \rightarrow q$	(vi)	$(p \rightarrow q) \vee (q \rightarrow \neg\neg p)$
(ii)	$\neg p$	(vii)	p_4
(iii)	p	(viii)	$(p_1 \leftrightarrow p_2) \vee \neg p_2$
(iv)	$(p \wedge q) \wedge (q \wedge p)$	(ix)	$\neg(p_1 \wedge p_2) \wedge \neg p_2$
(v)	$\neg(p \rightarrow q)$	(x)	$(p \wedge (q \wedge r)) \vee p$

We now discuss the nature of the last clause of definition 1, which reads:

> Only that which can be generated by the clauses (i)–(iii) in a finite number of steps is a formula in L.

A clause like this is sometimes called the *induction clause* of a definition. It plays a special and important role. If someone were to define a sheep as that which is the offspring of two sheep, we would not find this very satisfactory. It doesn't seem to say very much, since if you don't know what a sheep is, then you are not going to be much wiser from hearing the definition. The definition of a sheep as the offspring of two sheep is *circular*. Now it might seem that definition 1 is circular too: clause (ii), for example, states that a $\neg$ followed by a formula is a formula. But there is really no problem here, since the formula ϕ occurring after the $\neg$ is simpler than the formula $\neg\phi$, in the sense that it contains fewer connectives, or equivalently, that it can be generated by clauses (i)–(iii) in fewer steps. Given that this ϕ is a formula, it must be a formula according to one of the clauses (i)–(iii). This means that either ϕ is a propositional letter (and we know what these are), or else it is a composite formula built up of simpler formulas. So ultimately everything reduces to propositional letters.

In a definition such as definition 1, objects are said to have a given property (in this case that of being a formula) if they can be constructed from other, 'simpler' objects with that property, and ultimately from some group of objects which are simply said to have that property. Such definitions are said to be *inductive* or *recursive*.

The circular definition of a sheep as the offspring of two sheep can be turned into an inductive definition (i) by stipulating two ancestral sheep, let us call them Adam and Eve; and (ii) by ruling that precisely those things are sheep which are required to be sheep by (i) and the clause saying that the offspring of two sheep is a sheep. The construction tree of any given sheep, according to this inductive definition, would be a complete family tree going

back to the first ancestral sheep Adam and Eve (though contrary to usual practice with family trees, Adam and Eve will appear at the bottom).

Most of what follows applies equally to all propositional languages, so instead of referring to the formulas of any particular propositional language, we shall refer to *the formulas of propositional logic*.

Because the concept of a formula is defined inductively, we have at our disposal a simple method by which we can prove that all formulas have some particular property which we may be interested in. It is this. In order to prove that all formulas have a property A, it is sufficient to show that:

(i) The propositional letters all have property A;
(ii) if a formula ϕ has A, then $\neg\phi$ must too;
(iii) if ϕ and ψ have property A, then $(\phi \wedge \psi)$, $(\phi \vee \psi)$, $(\phi \rightarrow \psi)$, and $(\phi \leftrightarrow \psi)$ must too.

This is sufficient because of induction clause (iv), which ensures that every composite formula must be composed of some simpler formula(s) from which it inherits property A. A proof of this sort is called a *proof by induction on the complexity of the formula* (or a *proof by induction on the length of the formula*). As an example of a proof by induction on the complexity of a formula, we have the following simple, rigorous proof of the fact that all formulas of propositional logic have just as many right brackets as left brackets:

(i) Propositional letters have no brackets at all.
(ii) If ϕ has the same number of right brackets as left brackets, then $\neg\phi$ must too, since no brackets have been added or taken away.
(iii) If ϕ and ψ each have as many right brackets as left brackets, then $(\phi \wedge \psi)$, $(\phi \vee \psi)$, $(\phi \rightarrow \psi)$, and $(\phi \leftrightarrow \psi)$ must too, since in all of these exactly one left and one right bracket have been added.

Quite generally, for every inductive definition there is a corresponding kind of proof by induction.

There are various points in this book where if complete mathematical rigor had been the aim, inductive proofs would have been given. Instead we choose merely to note that strictly speaking, a proof is required.

The fact that the concept of a formula has been strictly defined by definition 1 enables us to give strict inductive definitions of notions about formulas. For example, let us define the function $(\phi)^0$ from formulas to natural numbers by:

$$(p)^0 = 0,$$
$$(\neg\phi)^0 = (\phi)^0$$
$$((\phi * \psi))^0 = (\phi)^0 + (\psi)^0 + 2, \text{ for each two-place connective } *.$$

Then, for each formula ϕ, $(\phi)^0$ gives the number of brackets in the formula ϕ.

Exercise 3 ◇

(a) The *operator depth* of a formula of propositional logic is the maximal length of a 'nest' of operators occurring in it. E.g., $((\neg p \wedge q) \wedge \neg r)$ has operator depth 3. Give a precise definition of this notion, using the inductive definition of formulas.

(b) Think of the construction trees of formulas. What concepts are defined by means of the following ('simultaneous') induction?

$A(p) = 1$	$B(p) = 1$
$A(\neg\psi) = A(\psi) + 1$	$A(\neg\psi) = \max(B(\psi), A(\psi) + 1)$
$A(\psi \circ \chi) = \max(A(\psi), (A\chi)) + 1$	$B(\psi \circ \chi) = \max(B(\psi), B(\chi),$
for the two-place connectives $\circ$	$A(\psi) + A(\chi) + 1),$

Exercise 4 ◇

(a) What notions are described by the following definition by induction on formulas?

$p^* = 0$ for propositional letters p

$(\neg\phi)^* = \phi^*$

$(\phi \circ \psi)^* = \phi^* + \psi^* + 1$ for two-place connectives $\circ$

$p^+ = 1$

$(\neg\phi)^+ = \phi^+$

$(\phi \circ \psi)^+ = \phi^+ + \psi^+$ for two-place connectives $\circ$

(b) Prove by induction that for all formulas ϕ, $\phi^+ = \phi^* + 1$.

Exercise 5

In this exercise, the reader is required to translate various English sentences into propositional logic. An example is given which shows the kind of thing that is expected. We want a translation of the sentence:

If I have lost if I cannot make a move, then I have lost.

This sentence might, for example, be said by a player in a game of chess or checkers, if he couldn't see any move to make and didn't know whether the situation amounted to his defeat.

Solution

Translation: $(\neg p \to q) \to q$
Key: $+\delta$ p: I can make a move; q: I have lost.

Translate the following sentences into propositional logic. Preserve as much of the structure as possible and in each case give the key.

(1) This engine is not noisy, but it does use a lot of energy.
(2) It is not the case that Guy comes if Peter or Harry comes.
(3) It is not the case that Cain is guilty and Abel is not.
(4) This has not been written with a pen or a pencil.
(5) John is not only stupid but nasty too.
(6) Johnny wants both a train and a bicycle from Santa Claus, but he will get neither.
(7) Nobody laughed or applauded.

(8) I am going to the beach or the movies on foot or by bike.
(9) Charles and Elsa are brother and sister or nephew and niece.
(10) Charles goes to work by car, or by bike and train.
(11) God willing, peace will come.
(12) If it rains while the sun shines, a rainbow will appear.
(13) If the weather is bad or too many are sick, the party is not on.
(14) John is going to school, and if it is raining so is Peter.
(15) If it isn't summer, then it is damp and cold, if it is evening or night.
(16) If you do not help me if I need you, I will not help you if you need me.
(17) If you stay with me if I won't drink any more, then I will not drink any more.
(18) Charles comes if Elsa does and the other way around.
(19) John comes only if Peter does not come.
(20) John comes exactly if Peter does not come.
(21) John comes just when Peter stays home.
(22) We are going, unless it is raining.
(23) If John comes, then it is unfortunate if Peter and Jenny come.
(24) If father and mother both go, then I won't, but if only father goes, then I will go too.
(25) If Johnny is nice he will get a bicycle from Santa Claus, whether he wants one or not.
(26) You don't mean it, and if you do, I don't believe you.
(27) If John stays out, then it is mandatory that Peter or Nicholas participates.

2.4 Functions

Having given an exact treatment of the syntax of languages for propositional logic, we shall now move on to their semantics, which is how they are interpreted. The above has shown that what we have in mind when we speak of the interpretation of a propositional language is the attribution of truth values to its sentences. Such attributions are called *valuations*. But these valuations are functions, so first we shall say some more about functions.

A function, to put it quite generally, is an attribution of a unique *value* (or *image*, as it is sometimes called) to each entity of some specific kind (for a valuation, to each sentence of the language in question). These entities are called the *arguments* (or *originals*) of the function, and together they form its *domain*. The entities which figure as the possible values of a function are collectively called its *range*. If x is an argument of the function f then f(x) is the value which results when f is *applied* to x. The word *value* must not be taken to imply that we are dealing with a truth value or any other kind of number here, since any kind of thing may appear in the range of a function. The only requirement is that no argument may have more than a single value. A few examples of functions are given in table 2.2. The left column of the table is understood to contain names of functions, so that *date of birth of x*, for ex-

Table 2.2 Examples of Functions

Function	*Domain*	*Range*
Date of birth of x	People	Dates
Mother of x	People	Women
Head of state of x	Countries	People
Frame number of x	Bicycles	Numbers
Negation of x	Formulas of propositional logic	Formulas of propositional logic
Capital city of x	Countries	Cities
Sex of x	People	The two sexes (masculine, feminine)

ample, is a name of the function which accepts people as its arguments and attributes to them as their values their dates of birth. The value of *date of birth of x* for the argument Winston Churchill is, for example, 30 November 1874.

In order to make what we mean clearer, compare the following expressions similar to those in the table, which may *not* be considered names of functions: *eldest brother of x* (domain: people) may not be taken as a function, since not everyone has a brother, so it is not possible to attribute a value to every argument. *Parent of x* is not a function either, but not because some people lack parents; the problem is that everyone has, or at least has had, no less than two parents. So the values are not unique. Similarly, *direct object of x* (domain: English sentences) is not a function, because not every sentence has a direct object, and *verb of x* is not a function since some sentences have more than one verb.

In addition, there are also functions which require two domain elements in order to specify a value, or three elements or more. Some examples are given in table 2.3. Functions which require two arguments from the domain in order to specify a value are said to be *binary,* and to generalize, functions which require *n* arguments are said to be *n*-ary. An example of an expression which accepts two arguments but which nevertheless does not express a binary function is *quotient of x and y* (domain: numbers). We are not dealing with a function here because the value of this expression is undefined for any x if y is taken as 0.

Functions can be applied to their own values or to those of other functions provided these are of the right kind, that is, that they fall within the domain of the function in question. Examples of functions applied to each other are *date of birth of the mother of John, mother of the head of state of France, mother of the mother of the mother of Peter, sex of the mother of Charles* and *sum of the difference between 6 and 3 and the difference between 4 and 2:* $(6 - 3) + (4 - 2)$.

As we have said, each function has its own particular domain and range. If A is the domain of a function f and B is its range, then we write $f: A \rightarrow B$ and we say that f is a *function from A to B*, and that f *maps A into B*. There is one

important asymmetry between the domain of a function and its range, and that is that while a function must carry each element of its domain to some element of its range, this is not necessarily true the other way around: not every element of the range of a function needs to appear as the value of the function when applied to some element of its domain. The range contains all *possible* values of a function, and restricting it to the values which do in fact appear as values of the function is often inefficient. In the examples given above, a larger range has been chosen than is strictly necessary: all women instead of just those that are mothers in the case of *mother of x,* all people instead of just heads of state in *head of state of x,* and roads instead of roads forming the shortest route between cities in the case of *the shortest route between x and y.* In the special case in which every element of the range B of a function f appears as the value of that function when it is applied to some element of its domain A, we say that f is a function of A *onto* B. Of the functions in table 2.2, only *sex of x* is a function onto its range, and in table 2.3 only the sum and difference functions are, since every number is the sum of two other numbers and also the difference of two others.

The order of the arguments of a function can make a difference: the difference between 1 and 3 is -2, whereas that between 3 and 1 is $+2$. A binary function for which the order of the arguments makes no difference is said to be *commutative.* The sum function is an example of a commutative function, since the sum of x and y is always equal to the sum of y and x. One and the same object may appear more than once as an argument: there is, for example, a number which is the sum of 2 and 2.

The value of a function f when applied to arguments $x_1, \ldots, x_n$ is generally written in *prefix notation* as $f(x_1, \ldots, x_n)$, though *infix* notation is more usual for some well-known binary functions, such as $x + y$ for the sum of x and y, and $x - y$ for their difference, instead of $+(x, y)$ and $-(x, y)$, respectively.

A binary function f is said to be *associative* if for all objects x, y, z in its domain $f(x, f(y, z)) = f(f(x, y), z)$, or, in infix notation, if $xf(yfz) = (xfy)fz$. Clearly this notion only makes sense if f's range is part of its domain, since otherwise it will not always be possible to apply it to xfy and z. In other words, f is associative if it doesn't make any difference whether f is applied first to the first two of three arguments, or first to the second two. The sum

Table 2.3 Examples of Binary and Ternary Functions

Function	*Domain*	*Range*
Sum of x and y	Numbers	Numbers
Difference between x and y	Numbers	Numbers
Shortest route between x and y	Cities	Roads
Time at which the last train from x via y to z departs.	Stations	Moments of time

and the product of two numbers are associative functions, since for all numbers x, y, and z we have: $(x + y) + z = x + (y + z)$ and $(x \times y) \times z = x \times (y \times z)$. The difference function is not associative: $(4 - 2) - 2 = 0$, but $4 - (2 - 2) = 4$. The associativity of a function f means that that we can write $x_1fx_2fx_3 \ldots x_{n-1}fx_n$ without having to insert any brackets, since the value of the expression is independent of where they are inserted. Thus, for example, we have: $(x_1 + x_2) + (x_3 + x_4) = x_1 + ((x_2 + x_3) + x_4)$. First one has $(x_1 + x_2) + (x_3 + x_4) = x_1 + (x_2 + (x_3 + x_4))$, since $(x + y) + z = x + (y + z)$ for any x, y, and z, so in particular for $x = x_1$, $y = x_2$, and $z = x_3 + x_4$. And $x_1 + (x_2 + (x_3 + x_4)) = x_1 + ((x_2 + x_3) + x_4)$, since $x_2 + (x_3 + x_4) = (x_2 + x_3) + x_4$.

2.5 The Semantics of Propositional Logic

The valuations we have spoken of can now, in the terms just introduced, be described as (unary) functions mapping formulas onto truth values. But not every function with formulas as its domain and truth values as its range will do as a valuation. A valuation must agree with the interpretations of the connectives which are given in their truth tables. A function which attributes the value 1 to both p and ¬p, for example, cannot be accepted as a valuation, since it does not agree with the interpretation of negation. The truth table for ¬ (see (14)) rules that for every valuation V and for all formulas ϕ:

(i) $V(\neg\phi) = 1$ iff $V(\phi) = 0$.

This is because the truth value 1 is written under $\neg\phi$ in the truth table just in case a 0 is written under ϕ. Since $\neg\phi$ can only have 1 or 0 as its truth value (the range of V contains only 1 and 0), we can express the same thing by:

(i′) $V(\neg\phi) = 0$ iff $V(\phi) = 1$.

That is, a 0 is written under $\neg\phi$ just in case a 1 is written under ϕ.

Similarly, according to the other truth tables we have:

(ii) $V(\phi \wedge \psi) = 1$ iff $V(\phi) = 1$ and $V(\psi) = 1$.
(iii) $V(\phi \vee \psi) = 1$ iff $V(\phi) = 1$ or $V(\psi) = 1$.
(iv) $V(\phi \rightarrow \psi) = 0$ iff $V(\phi) = 1$ and $V(\psi) = 0$.
(v) $V(\phi \leftrightarrow \psi) = 1$ iff $V(\phi) = V(\psi)$.

Recall that *or* is interpreted as *and/or*. Clause (iii) can be paraphrased as: $V(\phi \vee \psi) = 0$ iff $V(\phi) = 0$ and $V(\phi) = 0$; (iv) as: $V(\phi \rightarrow \psi) = 1$ iff $V(\phi) = 0$ or $V(\psi) = 1$ (*or* = *and/or*). And if, perhaps somewhat artificially, we treat the truth values 1 and 0 as ordinary numbers, we can also paraphrase (iv) as: $V(\phi \rightarrow \psi) = 1$ iff $V(\phi) \leqslant V(\psi)$ (since while $0 \leqslant 0$, $0 \leqslant 1$, and $1 \leqslant 1$, we do not have $1 \leqslant 0$).

A valuation V is wholly determined by the truth values which it attributes to the propositional letters. Once we know what it does with the propositions, we can calculate the V of any formula ϕ by means of ϕ's construction tree. If

$V(p) = 1$ and $V(q) = 1$, for example, then $V(\neg(\neg p \wedge \neg q))$ can be calculated as follows. We see that $V(\neg p) = 0$ and $V(\neg q) = 0$, so $V(\neg p \wedge \neg q) = 0$ and thus $V(\neg(\neg p \wedge \neg q)) = 1$. Now it should be clear that only the values which V attributes to the proposition letters actually appearing in ϕ can have any influence on $V(\phi)$. So in order to see how the truth value of ϕ varies with valuations, it suffices to draw up what is called a *composite truth table*, in which the truth values of all subformulas of ϕ are calculated for every possible distribution of truth values among the propositional letters appearing in ϕ. To continue with the same example, the composite truth table for the formula $\neg(\neg p \wedge \neg q)$ is given as (31):

(31)

	1	2	3	4	5	6
	p	q	¬p	¬q	¬p ∧ ¬q	¬(¬p ∧ ¬q)
V_1	1	1	0	0	0	1
V_2	1	0	0	1	0	1
V_3	0	1	1	0	0	1
V_4	0	0	1	1	1	0

The four different distributions of truth values among p and q are given in columns 1 and 2. In columns 3 and 4, the corresponding truth values of ¬p and ¬q have been given; they are calculated in accordance with the truth table for negation. Then in column 5 we see the truth values of ¬p ∧ ¬q, calculated from columns 3 and 4 using the truth table for conjunction. And finally, in column 6 we see the truth values of ¬(¬p ∧ ¬q) corresponding to each of the four possible distributions of truth values among p and q, which are calculated from column 5 by means of the truth table for negation.

The number of rows in the composite truth table for a formula depends only on the number of different propositional letters occurring in that formula. Two different propositional letters give rise to four rows, and we can say quite generally that *n* propositional letters give rise to 2^n rows, since that is the number of different distributions of the two truth values among *n* propositions. Every valuation corresponds to just one row in a truth table. So if we restrict ourselves to the propositional letters p and q, there are just four possible valuations: the V_1, V_2, V_3, and V_4 given in (31). And these four are the only valuations which matter for formulas in which p and q are the only propositional letters, since as we have just seen, what V does with ϕ is wholly determined by what V does with the propositional letters actually appearing in ϕ. This means that we may add new columns to (31) for the evaluation of as many formulas as we wish composed from just the letters p and q together with connectives. That this is of some importance can be seen as follows.

Note that the composite formula ¬(¬p ∧ ¬q) is true whenever any one of the proposition letters p and q is true, and false if both p and q are false. This is just the inclusive disjunction of p and q. Now consider the composite truth table given in (32):

(32)

	1	2	3	4	5	6	7
	p	q	$\neg$p	$\neg$q	$\neg$p $\wedge$ $\neg$q	$\neg$($\neg$p $\wedge$ $\neg$q)	p $\vee$ q
V_1	1	1	0	0	0	1	1
V_2	1	0	0	1	0	1	1
V_3	0	1	1	0	0	1	1
V_4	0	0	1	1	1	0	0

What we have done is add a new column to the truth table mentioned above in which the truth value of p $\vee$ q is given for each distribution of truth values among p and q, this being calculated in accordance with the truth table for the disjunction. This shows clearly that the truth values of $\neg(\neg p \wedge \neg q)$ and p $\vee$ q are the same under each valuation, since

$$V_1(\neg(\neg p \wedge \neg q)) = V_1(p \vee q) = 1;$$
$$V_2(\neg(\neg p \wedge \neg q)) = V_2(p \vee q) = 1;$$
$$V_3(\neg(\neg p \wedge \neg q)) = V_3(p \vee q) = 1;$$
$$V_4(\neg(\neg p \wedge \neg q)) = V_4(p \vee q) = 0.$$

So for every valuation V we have: $V(\neg(\neg p \wedge \neg q)) = V(p \vee q)$. The formulas $\neg(\neg p \wedge \neg q)$ and p $\vee$ q are (*logically*) *equivalent*. To put it more explicitly, ϕ and ψ are said to be (*logically*) *equivalent* just in case for every valuation V we have: $V(\phi) = V(\psi)$. The qualification *logical* is to preclude any confusion with *material equivalence*.

In order to see how all formulas of the form $\neg(\neg\phi \wedge \neg\psi)$ and $\phi \vee \psi$ behave under all possible valuations, a composite truth table just like (32) can be drawn up by means of the truth tables for negation, conjunction, and disjunction. The result is given in (33):

(33)

ϕ	ψ	$\neg\phi$	$\neg\psi$	$\neg\phi \wedge \neg\psi$	$\neg(\neg\phi \wedge \neg\psi)$	$\phi \vee \psi$
1	1	0	0	0	1	1
1	0	0	1	0	1	1
0	1	1	0	0	1	1
0	0	1	1	1	0	0

In this truth table it can clearly be seen that the equivalence of formulas of the form $\neg(\neg\phi \wedge \neg\psi)$ and $\phi \vee \psi$ is quite general (for a general explication of relationships of this sort, see theorem 13 in §4.2.2).

Consider another example. All formulas of the forms $\neg\neg\phi$ and ϕ are equivalent, as is apparent from (34):

(34)

ϕ	$\neg\phi$	$\neg\neg\phi$
1	0	1
0	1	0

This equivalence is known as the *law of double negation*. And the last example we shall give is a truth table which demonstrates that $(\phi \vee \psi) \vee \chi$ is equivalent to $\phi \vee (\psi \vee \chi)$, and $(\phi \wedge \psi) \wedge \chi$ to $\phi \wedge (\psi \wedge \chi)$; see (35):

(35)

ϕ	ψ	χ	$\phi \vee \psi$	$(\phi \vee \psi) \vee \chi$	$\psi \vee \chi$	$\phi \vee (\psi \vee \chi)$	$\phi \wedge \psi$	$(\phi \wedge \psi) \wedge \chi$	$\psi \wedge \chi$	$\phi \wedge (\psi \wedge \chi)$
1	1	1	1	1	1	1	1	1	1	1
1	1	0	1	1	1	1	1	0	0	0
1	0	1	1	1	1	1	0	0	0	0
1	0	0	1	1	0	1	0	0	0	0
0	1	1	1	1	1	1	0	0	1	0
0	1	0	1	1	1	1	0	0	0	0
0	0	1	0	1	1	1	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0

The latter two equivalences are known as the *associativity of* $\wedge$ and the *associativity of* $\vee$, respectively, by analogy with the concept which was introduced in connection with functions and which bears the same name. (For a closer connection between these concepts, see §2.6.) Just as with functions, the associativity of $\vee$ and $\wedge$ means that we can omit brackets in formulas, since their meaning is independent of where they are placed. This assumes, of course, that we are only interested in the truth values of the formulas. In general, then, we shall feel free to write $\phi \wedge \psi \wedge \chi$, $(\phi \to \psi) \wedge (\psi \to \chi) \wedge (\chi \to \phi)$ etc. $\phi \wedge \psi \wedge \chi$ is true just in case all of ϕ, ψ, and χ are true, while $\phi \vee \psi \vee \chi$ is true just in case any one of them is true.

Exercise 6

A large number of well-known equivalences are given in this exercise. In order to get the feel of the method, it is worthwhile to demonstrate that a few of them are equivalences by means of truth tables and further to try to understand why they must hold, given what the connectives mean. The reader may find this easier if the metavariables ϕ, ψ, and χ are replaced by sentences derived from natural language.

Prove that in each of the following, all the formulas are logically equivalent to each other (independently of which formulas are represented by ϕ, ψ, and χ):

(a) $\phi, \neg\neg\phi, \phi \wedge \phi, \phi \vee \phi, \phi \wedge (\phi \vee \psi), \phi \vee (\phi \wedge \psi)$
(b) $\neg\phi, \phi \to (\psi \wedge \neg\psi)$
(c) $\neg(\phi \vee \psi), \neg\phi \wedge \neg\psi$ (*De Morgan's Law*)
(d) $\neg(\phi \wedge \psi), \neg\phi \vee \neg\psi$ (*De Morgan's Law*)
(e) $\phi \vee \psi, \psi \vee \phi, \neg\phi \to \psi, \neg(\neg\phi \wedge \neg\psi), (\phi \to \psi) \to \psi$
(f) $\phi \wedge \psi, \psi \wedge \phi, \neg(\phi \to \neg\psi), \neg(\neg\phi \vee \neg\psi)$
(g) $\phi \to \psi, \neg\phi \vee \psi, \neg(\phi \wedge \neg\psi), \neg\psi \to \neg\phi$
(h) $\phi \to \neg\psi, \psi \to \neg\phi$ (*law of contraposition*)
(i) $\phi \leftrightarrow \psi, (\phi \to \psi) \wedge (\psi \to \phi), (\phi \wedge \psi) \vee (\neg\phi \wedge \neg\psi)$
(j) $(\phi \vee \psi) \wedge \neg(\phi \wedge \psi), \neg(\phi \leftrightarrow \psi), \neg\phi \leftrightarrow \psi$, (and $\phi \infty \psi$, though officially it is not a formula of propositional logic according to the definition)
(k) $\phi \wedge (\psi \vee \chi), (\phi \wedge \psi) \vee (\phi \wedge \chi)$ (*distributive law*)
(l) $\phi \vee (\psi \wedge \chi), (\phi \vee \psi) \wedge (\phi \vee \chi)$ (*distributive law*)
(m) $(\phi \vee \psi) \to \chi, (\phi \to \chi) \wedge (\psi \to \chi)$
(n) $\phi \to (\psi \wedge \chi), (\phi \to \psi) \wedge (\phi \to \chi)$
(o) $\phi \to (\psi \to \chi), (\phi \wedge \psi) \to \chi$

The equivalence of $\phi \vee \psi$ and $\psi \vee \phi$ and of $\phi \wedge \psi$ and $\psi \wedge \phi$ as mentioned under (e) and (f) in exercise 6 are known as the *commutativity* of $\vee$ and $\wedge$, respectively. (For the connection with the commutativity of functions, see §2.6.) Both the equivalence mentioned under (h) and the equivalence of $\phi \to \psi$ and $\neg\psi \to \neg\phi$ given in (g) in exercise 6 are known as the *law of contraposition.*

Logically equivalent formulas always have the same truth values. This means that the formula χ' which results when one subformula ϕ of a formula χ is replaced by an equivalent formula ψ must itself be equivalent to χ. This is because the truth value of χ' depends on that of ψ in just the same way as the truth value of χ depends on that of ϕ. For example, if ϕ and ψ are equivalent, then $\phi \to \theta$ and $\psi \to \theta$ are too. One result of this is that the brackets in $(\phi \wedge \psi) \wedge \chi$ can also be omitted where it appears as a subformula of some larger formula, so that we can write $(\phi \wedge \psi \wedge \chi) \to \theta$, for example, instead of $((\phi \wedge \psi) \wedge \chi) \to \theta$, and $\theta \to ((\phi \to \psi) \wedge (\psi \leftrightarrow \chi) \wedge (\chi \vee \psi))$ instead of $\theta \to (((\phi \to \psi) \wedge (\psi \leftrightarrow \chi)) \wedge (\chi \vee \psi))$. More generally, we have here a useful way of proving equivalences on the basis of other equivalences which are known to hold. As an example, we shall demonstrate that $\phi \to (\psi \to \chi)$ is equivalent to $\psi \to (\phi \to \chi)$. According to exercise 6(o), $\phi \to (\psi \to \chi)$ is equivalent to $(\phi \wedge \psi) \to \chi$. Now $\phi \wedge \psi$ is equivalent to $\psi \wedge \phi$ (commutativity of $\wedge$), so $(\phi \wedge \psi) \to \chi$ is equivalent to $(\psi \wedge \phi) \to \chi$. Applying 6(o) once more, this time with ψ, ϕ, and χ instead of ϕ, ψ, and χ, we see that $(\psi \wedge \phi) \to \chi$ is equivalent to $\psi \to (\phi \to \chi)$. If we now link all these equivalences, we see that $(\phi \wedge \psi) \to \chi$ is equivalent to $\psi \to (\phi \to \chi)$, which is just what we needed.

Exercise 7 ◇

Show on the basis of equivalences of exercise 6 that the following formulas are equivalent:

(a) $\phi \leftrightarrow \psi$ and $\psi \leftrightarrow \phi$ (*commutativity of* $\leftrightarrow$)
(b) $\phi \to \neg\phi$ and $\neg\phi$
(c) $\phi \wedge (\psi \wedge \chi)$ and $\chi \wedge (\psi \wedge \phi)$
(d) $\phi \to (\phi \to \psi)$ and $\phi \to \psi$
(e) $\phi \infty \psi$ and $\phi \leftrightarrow \neg\psi$
(f) $\phi \infty \neg\psi$, $\neg\phi \infty \psi$, and $\phi \leftrightarrow \psi$

In a sense two equivalent formulas ϕ and ψ have the same meaning. We say that ϕ and ψ have the same *logical meaning*. So the remark made above can be given the following concise reformulation: logical meaning is conserved under replacement of a subformula by another formula which has the same logical meaning.

It is worth dwelling on the equivalence of $\phi \infty \psi$ and $\phi \leftrightarrow \neg\psi$ for a moment (exercise 7e). What this means is that *A unless B* and *A provided not B* have the same logical meaning: in logical terms then, (36) means the same as (37) (= (20)):

(36) We are going to see a film tonight, provided we are not going to the beach this afternoon.

(37) We are going to see a film tonight, unless we are going to the beach this afternoon.

Analogous points can be made with reference to the equivalences given in exercise 7f: *A unless not B* and *not A unless B* have the same logical meaning as *A provided B*, which means, among other things, that (38), (39) and (40) (= (29)) all express the same logical meaning:

(38) We are going to see a film tonight unless the dishes have not been done.

(39) We are not going to see a film tonight unless the dishes have been done.

(40) We are going to see a film tonight, provided the dishes have been done.

There are, of course, various reasons why one sentence may be preferred to another in any given context. What the equivalence of (38), (39), and (40) shows is that the reasons have nothing to do with the logical meaning of the sentences. The differences between these sentences are presumably to be explained in terms of their conditions of use, and it is there also that an explanation is to be sought for the peculiar nature of a sentence like:

(41) We are not going to see a film tonight provided we go to the beach this afternoon.

That there is a connection between material and logical equivalence is apparent if we compare the truth tables of the logically equivalent formulas p and $\neg\neg p$, and $p \wedge q$ and $q \wedge p$, with those of the material equivalences $p \leftrightarrow \neg\neg p$ and $(p \wedge q) \leftrightarrow (q \wedge p)$; see figures (42) and (43):

(42)

p	$\neg p$	$\neg\neg p$	$p \leftrightarrow \neg\neg p$
1	0	1	1
0	1	0	1

(43)

p	q	$p \wedge q$	$q \wedge p$	$(p \wedge q) \leftrightarrow (q \wedge p)$
1	1	1	1	1
1	0	0	0	1
0	1	0	0	1
0	0	0	0	1

In both cases we see that just one truth value occurs in the columns for the material equivalences, namely, 1. This is of course not entirely coincidental. It is precisely because under any valuation V, $V(p) = V(\neg\neg p)$ and $V(p \wedge q) = V(q \wedge p)$ that we always have $V(p \leftrightarrow \neg\neg p) = 1$ and $V((p \wedge q) \leftrightarrow (q \wedge p)) = 1$. Now this insight can be formulated as a general theorem:

Theorem 1

ϕ and ψ are logically equivalent iff for every valuation V, $V(\phi \leftrightarrow \psi) = 1$.

Proof: Generally speaking, a proof of a theorem of the form: A iff B is divided into (i) a proof that if A then B; and (ii) a proof that if B then A. The proof under (i) is headed by a $\Rightarrow$: and usually proceeds by first assuming A and then showing that B inevitably follows. The proof under (ii) is headed by

a $\Leftarrow$: and usually proceeds by first assuming B and then showing that A inevitably follows. So the proof of our first theorem goes like this:

$\Rightarrow$: Suppose ϕ and ψ are logically equivalent. This means that for every valuation V for the propositional letters occurring in ϕ and ψ, $V(\phi) = V(\psi)$. Then condition (v) on valuations says that we must have $V(\phi \leftrightarrow \psi) = 1$.

$\Leftarrow$: Suppose that $V(\phi \leftrightarrow \psi) = 1$ for all valuations V. Then there can be no V such that $V(\phi) \neq V(\psi)$, since otherwise $V(\phi \leftrightarrow \psi) = 0$; so for every V it must hold that $V(\phi) = V(\psi)$, whence ϕ and ψ are logically equivalent. □

The box □ indicates that the proof has been completed.

In theorem 2 in ‡4.2.2 we shall see that formulas ϕ such that $V(\phi) = 1$ for every valuation V are of special interest. These formulas can be known to be true without any information concerning the truth of the parts of which they are composed. Such formulas ϕ are called *tautologies*, and that ϕ is a tautology is expressed by $\vDash\phi$. So theorem 1 can now be rewritten as follows:

$\vDash\phi \leftrightarrow \psi$ iff ϕ and ψ are logically equivalent.

Now theorem 1 gives us an ample supply of tautologies all at once, for example: $((\phi \vee \psi) \vee \chi) \leftrightarrow (\phi \vee (\psi \vee \chi))$, $(\phi \vee \psi) \leftrightarrow \neg(\neg\phi \wedge \neg\psi)$, de Morgan's laws, etc. And given that $\vDash\phi \rightarrow \psi$ and $\vDash\psi \rightarrow \phi$ whenever $\vDash\phi \leftrightarrow \psi$, we have even more. (This last is because if for every V, $V(\phi) = V(\psi)$, then we can be sure that for every V, $V(\phi) \leq V(\psi)$ and $V(\psi) \leq V(\phi)$.) As examples of tautologies we now have all formulas of the form $(\phi \rightarrow \neg\psi) \rightarrow (\psi \rightarrow \neg\phi)$, and all those of the form $((\phi \vee \psi) \rightarrow \chi) \rightarrow ((\phi \rightarrow \chi) \wedge (\psi \rightarrow \chi))$. But there are many more, for example, all formulas of the form $\phi \rightarrow (\psi \rightarrow \phi)$, as is apparent from figure (44):

(44)

ϕ	ψ	$\psi \rightarrow \phi$	$\phi \rightarrow (\psi \rightarrow \phi)$
1	1	1	1
1	0	1	1
0	1	0	1
0	0	1	1

Exercise 8

Show of the following formulas that they are tautologies (for each ϕ, ψ, and χ):

(i) $\phi \rightarrow \phi$ (this actually follows from the equivalence of ϕ to itself)
(ii) $(\phi \wedge \psi) \rightarrow \phi$
(iii) $\phi \rightarrow (\phi \vee \psi)$
(iv) $\neg\phi \rightarrow (\phi \rightarrow \psi)$ (*ex falso sequitur quodlibet*)
(v) $\phi \vee \neg\phi$ (*law of the excluded middle*)
(vi) $(\phi \rightarrow (\psi \rightarrow \chi)) \rightarrow ((\phi \rightarrow \psi) \rightarrow (\phi \rightarrow \chi))$

(vii) $(\phi \to \psi) \vee (\psi \to \phi)$
(viii) $((\phi \to \psi) \to \phi) \to \phi$ (*Peirce's law*)

Obviously all tautologies are equivalent to each other; if we always have $V(\phi) = 1$ and $V(\psi) = 1$, then we certainly always have $V(\phi) = V(\psi)$.

That a formula ϕ is not a tautology is expressed as $\nvDash \phi$. If $\nvDash \phi$, then there is a valuation V such that $V(\phi) = 0$. Any such V is called a *counterexample to* ϕ (*'s being a tautology*). In §4.2.1 we shall go into this terminology in more detail. As an example we take the formula $(p \to q) \to (\neg p \to \neg q)$, which can be considered as the schema for invalid arguments like this: *If one has money, then one has friends. So if one has no money, then one has no friends.* Consider the truth table in (45):

(45)

p	q	$\neg p$	$\neg q$	$p \to q$	$\neg p \to \neg q$	$(p \to q) \to (\neg p \to \neg q)$
1	1	0	0	1	1	1
1	0	0	1	0	1	1
0	1	1	0	1	0	0
0	0	1	1	1	1	1

It appears that $\nvDash (p \to q) \to (\neg p \to \neg q)$, since a 0 occurs in the third row of the truth table. This row is completely determined by the circumstance that $V(p) = 0$ and $V(q) = 1$, in the sense that for every valuation V with $V(p) = 0$ and $V(q) = 1$ we have $V((p \to q) \to (\neg p \to \neg q)) = 0$. For this reason we can say that $V(p) = 0$, $V(q) = 1$ is a counterexample to $(p \to q) \to (\neg p \to \neg q)$.

We must be very clear that in spite of this we cannot say whether a sentence of the form $(\phi \to \psi) \to (\neg\phi \to \neg\psi)$ is a tautology or not without more information about the ϕ and ψ. If, for example, we choose p for both ϕ and ψ, then we get the tautology $(p \to p) \to (\neg p \to \neg p)$, and if we choose $p \vee \neg p$ and q for ϕ and ψ, respectively, then we get the tautology $((p \vee \neg p) \to q) \to (\neg(p \vee \neg p) \to \neg q)$. But if we choose p and q for ϕ and ψ, respectively, then we arrive at the sentence $(p \to q) \to (\neg p \to \neg q)$, which, as we saw in (45), is not a tautology.

Exercise 9

Determine of the following formulas whether they are tautologies. If any is not, give a counterexample. (Why is this exercise formulated with p and q, and not with ϕ and ψ as in exercise 8?)

(i) $(p \to q) \to (q \to p)$
(ii) $p \vee (p \to q)$
(iii) $(\neg p \vee \neg q) \to \neg(p \vee q)$
(iv) $((p \vee q) \wedge (\neg p \to \neg q)) \to q$
(v) $((p \to q) \to p) \to ((p \to q) \to q)$
(vi) $((p \to q) \to r) \to (p \to (q \to r))$

Closely related to the tautologies are those sentences ϕ such that for every valuation V, $V(\phi) = 0$. Such formulas are called *contradictions*. Since they are never true, only to utter a contradiction is virtually to contradict oneself. Best known are those of the form $\phi \wedge \neg\phi$ (see figure (46)).

(46)

ϕ	$\neg\phi$	$\phi \wedge \neg\phi$
1	0	0
0	1	0

We can obtain many contradictions from

Theorem 2

If ϕ is a tautology, then $\neg\phi$ is a contradiction.

Proof: Suppose ϕ is a tautology. Then for every V, $V(\phi) = 1$. But then for every V it must hold that $V(\neg\phi) = 0$. So according to the definition, ϕ is a contradiction. □

So $\neg((\phi \wedge \psi) \leftrightarrow (\psi \wedge \phi))$, $\neg(\phi \to \phi)$, and $\neg(\phi \vee \neg\phi)$ are contradictions, for example. An analogous proof gives us

Theorem 3

If ϕ is a contradiction, then $\neg\phi$ is a tautology.

This gives us some more tautologies of the form $\neg(\phi \wedge \neg\phi)$, the *law of noncontradiction*. All contradictions are equivalent, just like the tautologies. Those formulas which are neither tautologies nor contradictions are called (*logical*) *contingencies*. These are formulas ϕ such that there is both a valuation V_1 with $V_1(\phi) = 1$ and a valuation V_2 with $V_2(\phi) = 0$. The formula ϕ has, in other words, at least one 1 written under it in its truth table and at least one 0. Many formulas are contingent. Here are a few examples: p, q, $p \wedge q$, $p \to q$, $p \vee q$, etc. It should be clear that not all contingencies are equivalent to each other. One thing which can be said about them is:

Theorem 4

ϕ is a contingency iff $\neg\phi$ is a contingency.

Proof: (Another proof could be given from theorems 2 and 3, but this direct proof is no extra effort.)

$\Rightarrow$: Suppose ϕ is contingent. Then there is a V_1 with $V_1(\phi) = 1$ and a V_2 with $V_2(\phi) = 0$. But then we have $V_2(\neg\phi) = 1$ and $V_1(\neg\phi) = 0$, from which it appears that ϕ is contingent.
$\Leftarrow$: Proceeds just like $\Rightarrow$. □

Exercise 10

Let ϕ be a tautology, ψ a contradiction, and χ a contingency. Which of the following sentences are (i) tautological, (ii) contradictory, (iii) contingent, (iv) logically equivalent to χ.
(1) $\phi \wedge \chi$; (2) $\phi \vee \chi$; (3) $\psi \wedge \chi$; (4) $\psi \vee \chi$; (6) $\phi \vee \psi$; (7) $\chi \to \psi$.

(i) Prove the following general assertions:

(a) If $\phi \to \psi$ is a contradiction, then ϕ is a tautology and ψ a contradiction.

(b) $\phi \wedge \psi$ is a tautology iff ϕ and ψ are both tautologies.

(ii) Refute the following general assertion by giving a formula to which it does not apply.

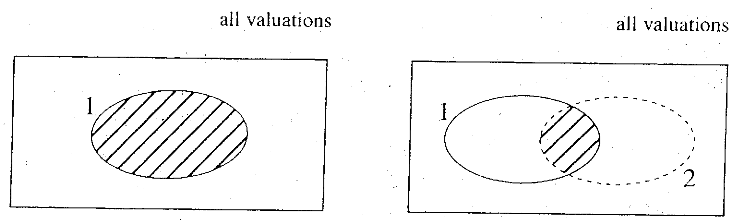
If $\phi \vee \psi$ is a tautology, then ϕ is a tautology or ψ is a tautology.

(iii) ◇ Prove the following general assertion:

If ϕ and ψ have no propositional letters in common, then $\phi \vee \psi$ is a tautology iff ϕ is a tautology or ψ is a tautology.

Before we give the wrong impression, we should emphasize that propositional logic is not just the science of tautologies or inference. Our semantics can just as well serve to model other important intellectual processes such as *accumulation of information*. Valuations on some set of propositional letters may be viewed as (descriptions of) states of the world, or situations, as far as they are expressible in this vocabulary. Every formula then restricts attention to those valuations ('worlds') where it holds: its 'information content'. More dynamically, successive new formulas in a discourse narrow down the possibilities, as in figure (47).

(47)



In the limiting case a unique description of one actual world may result. Note the inversion in the picture: the more worlds there still are in the information range, the less information it contains. Propositions can be viewed here as transformations on information contents, (in general) reducing uncertainty.

Exercise 12 ◇

Determine the valuations after the following three successive stages in a discourse (see (47)):

(1) $\neg(p \wedge (q \to r))$; (2) $\neg(p \wedge (q \to r))$, $(p \to r) \to r$; (3) $\neg(p \wedge (q \to r))$, $(p \to r) \to r$, $r \to (p \vee q)$.

2.6 Truth functions

The connectives were not introduced categorematically when we discussed the syntax of propositional logic, but syncategorematically. And parallel to this, they were not interpreted directly in §2.5, but contextually. We did not interpret $\wedge$ itself; we just indicated how $\phi \wedge \psi$ should be interpreted once interpretations are fixed for ϕ and ψ. It is, however, quite possible to interpret $\wedge$ and the other connectives directly, as *truth functions;* these are functions with truth values as not only their range but also their domain.

The connective $\wedge$, for example, can be interpreted as the function $f_\wedge$ such that $f_\wedge(1, 1) = 1$, $f_\wedge(1, 0) = 0$, $f_\wedge(0, 1) = 0$, and $f_\wedge(0, 0) = 0$. Analogously, as interpretations of $\vee$, $\to$, and $\leftrightarrow$, the functions $f_\vee$, $f_\to$, and $f_\leftrightarrow$ can be given, these being defined by:

$f_\vee(1, 1) = f_\vee(1, 0) = {=}_\vee(0, 1) = 1$ and $f_\vee(0, 0) = 0$.
$f_\to(1, 1) = f_\to(0, 1) = f_\to(0, 0) = 1$ and $f_\to(1, 0) = 0$.
$f_\leftrightarrow(1, 1) = f_\leftrightarrow(0, 0) = 1$ and $f_\leftrightarrow(1, 0) = f_\leftrightarrow(0, 1) = 0$.

Finally, $\neg$ can be interpreted as the unary truth function $f_\neg$ defined by $f_\neg(1) = 0$ and $f_\neg(0) = 1$. Then, for every V we have $V(\neg\phi) = f_\neg(V(\phi))$; and if $\circ$ is any one of our binary connectives, then for every V we have $V(\phi \circ \psi) = f_\circ(V(\phi), V(\psi))$.

The language of propositional logic can very easily be enriched by adding new truth-functional connectives, such as, for example, the connective ∞ with, as its interpretation, f_∞ defined by $f_\infty(1, 0) = f_\infty(0, 1) = 1$ and $f_\infty(1, 1) = f_\infty(0, 0) = 0$. Conversely, a connective can be introduced which is to be interpreted as any truth function one might fancy.

But it turns out that there is a sense in which all of this is quite unnecessary, since we already have enough connectives to express any truth functions which we might think up. Let us begin with the unary truth functions. Of these there are just four (see figures (48a–d)):

(48) a. f_1

x	$f_1(x)$
1	1
0	1

b. f_2

x	$f_2(x)$
1	1
0	0

c. f_3

x	$f_3(x)$
1	0
0	1

d. f_4

x	$f_4(x)$
1	0
0	0

Apparently f_1, f_2, f_3, and f_4 are the only candidates to serve as the interpretation of a truth-functional unary connective. Now it is easy enough to find formulas whose truth tables correspond precisely to those truth functions. Just take $p \vee \neg p$, p, $\neg p$, and $p \wedge \neg p$.

There are exactly sixteen binary truth functions, and as it is not difficult to see, the general expression for the number of n-ary truth functions is 2^{2^n}. Now it can be proved that all of these truth functions can be expressed by means of the connectives which we already have at our disposal. That is, there is a general method which generates, given the table of any truth function at all, a formula with this table as its truth table. That is, the following theorem can be proved: