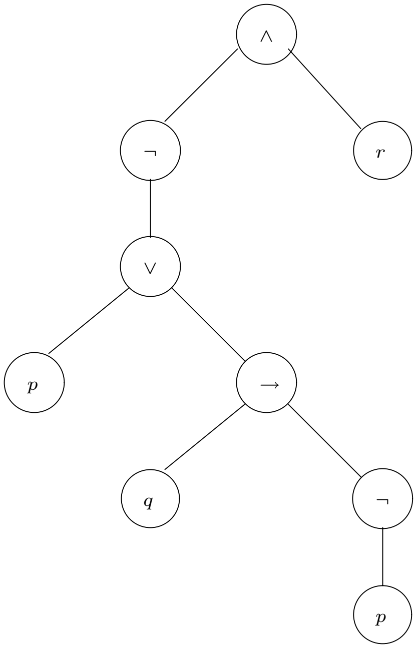

Figure 1.4. Given: a tree; wanted: its linear representation as a logical formula.

The tree in Figure 1.21 on page 82, however, does *not* represent a well-formed formula for two reasons. First, the leaf $\wedge$ (and a similar argument applies to the leaf $\neg$), the left subtree of the node $\rightarrow$, is not a propositional atom. This could be fixed by saying that we decided to leave the left and right subtree of that node unspecified and that we are willing to provide those now. However, the second reason is fatal. The p node is not a leaf since it has a subtree, the node $\neg$. This cannot make sense if we think of the entire tree as some logical formula. So this tree does not represent a well-formed logical formula.

1.4 Semantics of propositional logic

1.4.1 The meaning of logical connectives

In the second section of this chapter, we developed a calculus of reasoning which could verify that sequents of the form $\phi_1, \phi_2, \ldots, \phi_n \vdash \psi$ are valid, which means: from the premises ϕ_1, ϕ_2, $\ldots$, ϕ_n, we may conclude ψ.

In this section we give another account of this relationship between the premises ϕ_1, ϕ_2, $\ldots$, ϕ_n and the conclusion ψ. To contrast with the sequent above, we define a new relationship, written

$$\phi_1, \phi_2, \ldots, \phi_n \vDash \psi.$$

This account is based on looking at the 'truth values' of the atomic formulas in the premises and the conclusion; and at how the logical connectives manipulate these truth values. What is the truth value of a declarative sentence, like sentence (3) 'Every even natural number > 2 is the sum of two prime numbers'? Well, declarative sentences express a fact about the real world, the physical world we live in, or more abstract ones such as computer models, or our thoughts and feelings. Such factual statements either match reality (they are *true*), or they don't (they are *false*).

If we combine declarative sentences p and q with a logical connective, say $\wedge$, then the truth value of $p \wedge q$ is determined by three things: the truth value of p, the truth value of q and the meaning of $\wedge$. The meaning of $\wedge$ is captured by the observation that $p \wedge q$ is true iff p *and* q are both true; otherwise $p \wedge q$ is false. Thus, as far as $\wedge$ is concerned, it needs only to know whether p and q are true, it does *not* need to know what p and q are actually saying about the world out there. This is also the case for all the other logical connectives and is the reason why we can compute the truth value of a formula just by knowing the truth values of the atomic propositions occurring in it.

Definition 1.28 1. The set of truth values contains two elements T and F, where T represents 'true' and F represents 'false'.
2. A *valuation* or *model* of a formula ϕ is an assignment of each propositional atom in ϕ to a truth value.

Example 1.29 The map which assigns T to q and F to p is a valuation for $p \vee \neg q$. Please list the remaining three valuations for this formula.

We can think of the meaning of $\wedge$ as a function of two arguments; each argument is a truth value and the result is again such a truth value. We specify this function in a table, called the *truth table for conjunction*, which you can see in Figure 1.5. In the first column, labelled ϕ, we list all possible

ϕ	ψ	$\phi \wedge \psi$
T	T	T
T	F	F
F	T	F
F	F	F

Figure 1.5. The truth table for conjunction, the logical connective $\wedge$.

ϕ	ψ	$\phi \wedge \psi$
T	T	T
T	F	F
F	T	F
F	F	F

ϕ	ψ	$\phi \vee \psi$
T	T	T
T	F	T
F	T	T
F	F	F

ϕ	ψ	$\phi \to \psi$
T	T	T
T	F	F
F	T	T
F	F	T

ϕ	$\neg\phi$
T	F
F	T

$\top$
T

$\bot$
F

Figure 1.6. The truth tables for all the logical connectives discussed so far.

truth values of ϕ. Actually we list them *twice* since we also have to deal with another formula ψ, so the possible number of combinations of truth values for ϕ and ψ equals $2 \cdot 2 = 4$. Notice that the four pairs of ϕ and ψ values in the first two columns really exhaust all those possibilities (TT, TF, FT and FF). In the third column, we list the result of $\phi \wedge \psi$ according to the truth values of ϕ and ψ. So in the first line, where ϕ and ψ have value T, the result is T again. In all other lines, the result is F since at least one of the propositions ϕ or ψ has value F.

In Figure 1.6 you find the truth tables for all logical connectives of propositional logic. Note that $\neg$ turns T into F and vice versa. Disjunction is the mirror image of conjunction if we swap T and F, namely, a disjunction returns F iff both arguments are equal to F, otherwise (= at least one of the arguments equals T) it returns T. The behaviour of implication is not quite as intuitive. Think of the meaning of $\to$ as checking whether *truth is being preserved.* Clearly, this is not the case when we have $\mathtt{T} \to \mathtt{F}$, since we infer something that is false from something that is true. So the second entry in the column $\phi \to \psi$ equals F. On the other hand, $\mathtt{T} \to \mathtt{T}$ obviously preserves truth, but so do the cases $\mathtt{F} \to \mathtt{T}$ and $\mathtt{F} \to \mathtt{F}$, because there is no truth to be preserved in the first place as the assumption of the implication is false.

If you feel slightly uncomfortable with the semantics (= the meaning) of $\to$, then it might be good to think of $\phi \to \psi$ as an abbreviation of the formula $\neg\phi \vee \psi$ *as far as meaning is concerned*; these two formulas are very different syntactically and natural deduction treats them differently as well. But using the truth tables for $\neg$ and $\vee$ you can check that $\phi \to \psi$ evaluates to T iff $\neg\phi \vee \psi$ does so. This means that $\phi \to \psi$ and $\neg\phi \vee \psi$ are *semantically equivalent*; more on that in Section 1.5.

Given a formula ϕ which contains the propositional atoms $p_1, p_2, \ldots, p_n$, we can construct a truth table for ϕ, at least in principle. The caveat is that this truth table has 2^n many lines, each line listing a possible combination of truth values for $p_1, p_2, \ldots, p_n$; and for large n this task is impossible to complete. Our aim is thus to compute the value of ϕ for each of these 2^n cases for moderately small values of n. Let us consider the example ϕ in Figure 1.3. It involves three propositional atoms ($n = 3$) so we have $2^3 = 8$ cases to consider.

We illustrate how things go for one particular case, namely for the valuation in which q evaluates to F; and p and r evaluate to T. What does $\neg p \wedge q \to p \wedge (q \vee \neg r)$ evaluate to? Well, the beauty of our semantics is that it is *compositional.* If we know the meaning of the subformulas $\neg p \wedge q$ and $p \wedge (q \vee \neg r)$, then we just have to look up the appropriate line of the $\to$ truth table to find the value of ϕ, for ϕ is an implication of these two subformulas. Therefore, we can do the calculation by traversing the parse tree of ϕ in a bottom-up fashion. We know what its leaves evaluate to since we stated what the atoms p, q and r evaluated to. Because the meaning of p is T, we see that $\neg p$ computes to F. Now q is assumed to represent F and the conjunction of F and F is F. Thus, the left subtree of the node $\to$ evaluates to F. As for the right subtree of $\to$, r stands for T so $\neg r$ computes to F and q means F, so the disjunction of F and F is still F. We have to take that result, F, and compute its conjunction with the meaning of p which is T. Since the conjunction of T and F is F, we get F as the meaning of the right subtree of $\to$. Finally, to evaluate the meaning of ϕ, we compute F $\to$ F which is T. Figure 1.7 shows how the truth values propagate upwards to reach the root whose associated truth value is the truth value of ϕ given the meanings of p, q and r above.

It should now be quite clear how to build a truth table for more complex formulas. Figure 1.8 contains a truth table for the formula $(p \to \neg q) \to (q \vee \neg p)$. To be more precise, the first two columns list all possible combinations of values for p and q. The next two columns compute the corresponding values for $\neg p$ and $\neg q$. Using these four columns, we may compute the column for $p \to \neg q$ and $q \vee \neg p$. To do so we think of the first and fourth columns as the data for the $\to$ truth table and compute the column of $p \to \neg q$ accordingly. For example, in the first line p is T and $\neg q$ is F so the entry for $p \to \neg q$ is T $\to$ F $=$ F by definition of the meaning of $\to$. In this fashion, we can fill out the rest of the fifth column. Column 6 works similarly, only we now need to look up the truth table for $\vee$ with columns 2 and 3 as input.

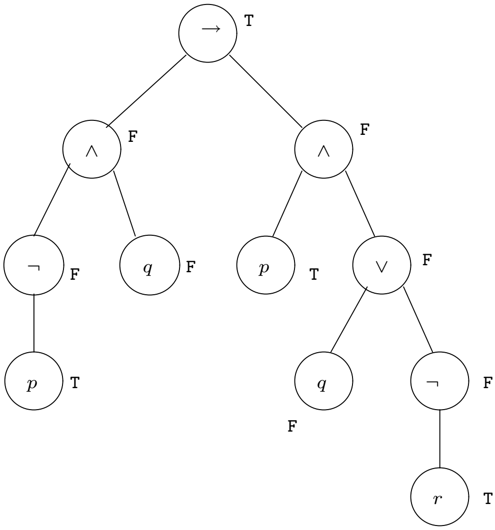

Figure 1.7. The evaluation of a logical formula under a given valuation.

p	q	$\neg p$	$\neg q$	$p \to \neg q$	$q \vee \neg p$	$(p \to \neg q) \to (q \vee \neg p)$
T	T	F	F	F	T	T
T	F	F	T	T	F	F
F	T	T	F	T	T	T
F	F	T	T	T	T	T

Figure 1.8. An example of a truth table for a more complex logical formula.

Finally, column 7 results from applying the truth table of $\to$ to columns 5 and 6.

1.4.2 Mathematical induction

Here is a little anecdote about the German mathematician Gauss who, as a pupil at age 8, did not pay attention in class (can you imagine?), with the result that his teacher made him sum up all natural numbers from 1 to 100. The story has it that Gauss came up with the correct answer 5050 within seconds, which infuriated his teacher. How did Gauss do it? Well, possibly he knew that

$$1 + 2 + 3 + 4 + \cdots + n = \frac{n \cdot (n+1)}{2} \tag{1.5}$$