Chapter 2

Binary Boolean Arithmetic: Calculating Truth Functions

Most people seem to think that once you have solved a problem, that's it. It's all over. Mathematicians, though, are seldom satisfied with mere answers; they have an ingrained habit of using the solutions as a springboard for asking new questions. If a calculation miraculously works out with a clean answer, instead of grinding to a halt in the usual morass, most people would breathe a sigh of relief and accept the miracle. Mathematicians, ever suspicious, always want to take the miracle to bits and see what makes it tick. There is a good reason for doing this. Sometimes you find out that it wasn't a miracle at all, but a mistake. Sometimes you find a better way of working miracles.
–Ian Stewart[92]

Fall in New England only happens in New England.
–Southwest Airlines

2.1 Reasoning by Calculation from Leibniz to Boole

2.1.1 Boolean Algebra and Modern Logic: Introductory Remarks*

Euclid's *Elements*[25] introduced the axiomatic method to the world and was its first triumph. Starting with a small number of "first principles", **axioms** and **postulates** which he assumed without proof, he endeavored to deduce all known propositions of geometry. He stated the first principles in terms of certain undefined or **primitive** concepts, such as that of point, line and distance, in

*This part may be skipped without loss of continuity.

terms of which other concepts, for instance that of a triangle or a circle can be defined[1].

Euclid's *Elements* remained a model for all deductive reasoning, and for the attainment of knowledge by deduction from self evident principles, which was not seriously challenged until the Renaissance.

Nor, curiously, was Aristotle's syllogistic reasoning seriously challenged until the 19th century, even though it is incapable of accounting for the principles of inference, which Euclid used informally, and which are fundamental for mathematical reasoning. And yet even in the late eighteenth century, Kant[40] could still claim that the syllogistic of Aristotle, already two thousand years old, "may be considered as completed and perfect".

But outside of geometry, arithmetic, algebra and the calculus continued to be based primarily on calculation. The axiomatic method overshadowed calculation in these subjects only in the the late 19th century. And from then on, those principles of reasoning, which go beyond the Aristotelian syllogistic began to be subjected to closer scrutiny, starting with the work of Boole and de Morgan[2], and continuing with that of Frege, Russell and then many others. In all these systems of mathematical or formal logic, which we discuss in this book, every proof can in principle be reduced to one in which each step can be verified as correct by a mechanical procedure or algorithm akin to calculation.

The idea that a step in logical inference is essentially the same as a step in calculation in arithmetic goes back to the ancient Greeks. Leibniz went further. In 1686 he suggested that all disputes and differences of opinion can be reduced to calculation[52]. He called for a "universal characteristic", a system akin to algebra in which "every paralogism be recognized as an *error in calculation*", so that "disputation will no more be needed between two philosophers than two computers" (which in those days were people, not electronic machines).

But this ambitious proposal had no effect for over a century, until 1847, with the publication of George Boole's ground breaking work, *The Mathematical Analysis of Logic*[13]. That book was the beginning of modern logic. Ever since, the subject has grown explosively, like the rest of modern mathematics and science, far beyond anything which Leibniz could have imagined.

At the same time, modern mathematics, logic and science are much narrower

[1]Unfortunately, Euclid did not clearly recognize that you have to have primitive concepts which are left undefined. For instance, Euclid's famous attempt to define a line as "that which lies evenly with the points of itself" would replace the notion of a line with the vaguer one of "lying evenly" with the points of itself.

Hilbert's famous *Foundations of Geometry*[36] provided a completely rigorous axiomatization of Euclid's principles. Other formulations soon followed, for example, making distance a primitive notion, and assuming appropriate properties of distance, or making congruence and "rigid motion" primitive. and assuming appropriate properties of "groups" of rigid motions.

Euclid also failed to state all of his assumptions explicitly, so that some of his propositions do not logically follow from his stated postulates. For example, the notion of continuity was not even mentioned, so that the assumption that all lines are continuous was used tacitly in his proof of Proposition 21 of Book I of the *The Elements*. The need postulate for continuity, first introduced by Moritz Pasch[64] (1843-1930) now bears his name. It says that a line which intersects one side of a triangle but none of its vertices must intersect another side.

[2]Augustus De Morgan, 1806-1871

in scope than what Leibniz had hoped for. Politics, religion and morality, for example, are still beyond the reach of any "universal characteristic" and probably always will be. For one thing, everyday reasoning, as well as reasoning about politics, religion and morality, is conducted in natural languages, such as English and Chinese, in which both peasants and philosophers reason and debate. And they are much more complicated than any formal language.

It would be a futile exercise, for example, to try to symbolize an article from the *Atlantic Monthly*, and attempt to evaluate with complete formal rigor the arguments it contains. Such arguments, like those we use consciously or not to arrive at decisions without which we could hardly function on a day to day basis, leave too much unsaid, both in what is assumed and what is inferred.

Even in the most simple decisions and inferences presuppose a staggering number of unstated "common sense" assumptions which every child learns but which linguists and computer scientists in "artificial intelligence" have sought in vain to discover. Which is why today's computers lack "common sense"[3].

For this among other reasons, not all **paralogisms** are mistakes in inference. A far more common problem in matters political, religious or moral, lies in the *assumptions* from which the inferences are made. And such assumptions, were then as now hotly disputed. "What is madness?" asked Voltaire[95], "it is to have false perceptions and to reason correctly from them".

To a large extent, the exchange of ideas among scientists has forged a consensus, especially in the mathematical and physical sciences, which moralists, theologians and philosophers can only dream of, and there is some truth in Voltaire's famous quip "there are no sects in geometry". But he would also have applauded Russell's observation echoing one of La Rochefoucauld's[50]:

> *When there are rational grounds for an opinion, people are content to set them forth and wait for them to operate. In such cases, people do not hold their opinions with passion; they hold them calmly, and set forth their reasons quietly. The opinions that are held with passion are always those for which no good ground exists; indeed the passion is the measure of the holder's lack of rational conviction. Opinions in politics and religion are almost always held passionately.*

Boole had observed that his algebra, now called **Boolean algebra**, and originally intended to be about classes, with a view to reforming Aristotelian **syllogistics**, could also be interpreted as being an **algebra of truth values**. From the algebra of truth values arose **two valued propositional logic**, or **Boolean logic**, cast in arithmetical form. It is the study of the two element-Boolean algebra $\mathbf{B}_2$, in which the variables range over the set $\{0, 1\}$ of truth values.

Boolean logic, while far from universal, is certainly a "characteristic" in one sense: any mistake in inference, in the application of the stated rules, in the

[3]For instance, in the well known medical diagnostic program MYCIN, when investigators typed in the symptoms of a rusty 1967 Chevrolet, the program announced that the car had measles!

course of a step by step proof of a conclusion from given assumptions, can be corrected in much the same way as a mistake in the application of arithmetical rules in a step by step calculation.

For the truth value of any statement susceptible to analysis by Boole's methods is completely determined, once the truth values of each of its smallest parts is given. For example, 'Mars is a planet and the sun is a star' is true `iff` both of the statements 'Mars is a planet' and 'the sun is a star' are true; their truth values determine the truth value of the whole, the natural language "conjunction" of 'Mars is a planet' with 'the sun is a star'. In this sense the word 'and' which "connected" the two parts of the "conjunction" into a single whole, is "truth functional".

But not all natural language uses of 'and' are truth functional. Benson Mates' famous distinction between 'he brushed his teeth and went to bed' and 'he went to bed and brushed his teeth' comes to mind. Here tense, the culprit of Benson Mates' example[57], is not an issue in classical mathematics, where conjunctions and other such compounds are in fact truth functional. The word 'because' is also not truth functional: to take Quine's example[68], knowing that Jones died and ate fish with ice cream cannot tell us whether Jones died because he ate fish with ice cream.

Many introductory texts use carefully chosen examples from ordinary language in an effort to persuade the unwary reader that all statements, even all implications, (of which more below) are truth functional[4]. Boole's methods apply only to arguments in which the truth value of the parts determine the truth value of the whole. Thus it is far from being the "universal characteristic" that Leibniz sought. But in one sense, it is a "characteristic", for the truth value of a Boolean formula is determined by those assigned to its smallest parts, and is subject to calculation.

So while on occasion we do also use non-mathematical statements in our examples connected with inferences to illustrate a (hopefully legitimate) point, such examples are a means only and are not intended as a serious analysis of ordinary usage. They are no more central a concern of Boolean algebra than are word problems about John and Mary's ages of ultimate importance in the algebra of the real number system.

What Leibniz did not sufficiently appreciate was that there is more to systems of logic, even as narrow in scope as Boolean logic, than checking (by computation) for the correctness of *proofs* of a conclusion from given assumptions. One also wants to know whether a proof of a given proposition from given assumptions *exists* in the first place. In Boolean logic, this question too can be settled by calculation. But it is a practical impossibility to do so in all but the simplest cases, such as the set exercises in logic books.

[4]We are reminded of a parallel case which H. D. F. Kitto recounts[44]: the book on sixteenth century counterpoint which for many years was the standard text on the subject, whose renowned author Rockstro explained that most of the examples in his work were his own invention, since Palestrina, Vittoria and the other masters of the art did not provide examples to suit his analysis! As Kitto drily remarked, "academic criticism, at its worst, can be exceedingly funny".

For as far as anyone knows so far, in Boolean logic there is no way which is always shorter to test for provability than to exhaust all possibilities, and every introduction of a new variable doubles the number of a calculations required. So if a particular problem of this sort involved a hundred variables, a computer capable of a trillion calculations per second would take about 300 billion years to finish. A computer 1,000 times faster would take as long if such a problem involved ten more variables.

First order logic, which we introduce in Chapters 6 and 7, extends propositional logic to allow inferences involving generalities and statements of existence. In that system, it is generally not just impractical, but *impossible* for any computer to determine, even in as limited a system as "elementary" (i.e. first order) arithmetic, whether a given proposition follows from or is *entailed by* its axioms (provided that a computer can tell the difference between an axiom and a non-axiom in the first place). And even in elementary arithmetic, it is not even possible in general, for a computer to determine whether the given statement is true, whether an axiom or not.

But if the power of Boolean logic, and of formal systems in general, falls short of what Leibniz envisioned, it will be interesting to see why, and here mathematical reasoning will help us. For Boolean logic, like first order logic and the various systems of formal logic, not only formalize parts of mathematical reasoning, but are themselves part of mathematics. And the structure of the entailment relations they produce are themselves objects of mathematical study.

2.2 Boole's Arithmetic: The Complement of a Truth Value, and the Rules for Adding and Multiplying Truth Values

Boolean Arithmetic and Symbolization: An Overview

In ordinary algebra, the variables 'x', 'y', 'z', and so on, stand for numbers. You can calculate what number a formula, say '$(x \cdot y) + (x \cdot z)$' stands for, when you are given the numbers which the variables 'x', 'y' and 'z', which appear in the formula, stand for. For instance, if $x = 2$, $y = 3$ and $z = 4$, then,

$$\begin{aligned}(x \cdot y) + (x \cdot z) &= (2 \cdot 3) + (2 \cdot 4) \\ &= 6 + 8 \\ &= 12 \end{aligned} \tag{2.1}$$

On the first line, we *substituted* the numeral '2' for the variable 'x' in the formula on the left, that is, we replaced each occurrence of 'x' by '2', and also substituted '3' for 'y' and '4' for 'z', to obtain the term on the right, and then did the calculation.

In the **two-element Boolean algebra $\mathbf{B}_2$**, the variables are capital letters which range over the numbers 0 and 1, which we call **truth values**. The symbol

' $\wedge$ ' is the **logical multiplication sign**. The logical multiplication table for $\mathbf{B}_2$ is:

$\wedge$	0	1
0	0	0
1	0	1

Table 2.1: Logical Multiplication for $\mathbf{B}_2$

So logical multiplication of truth values is the same as ordinary arithmetical multiplication when the only numbers you are multiplying are 0 and 1. The symbol ' $\vee$ ' is the **logical addition sign**. The logical addition table for $\mathbf{B}_2$ is this:

$\vee$	0	1
0	0	1
1	1	1

Table 2.2: Logical Addition for $\mathbf{B}_2$

Logical addition of truth values is therefore the same as ordinary arithmetical addition of 1's and 0's, except that 1 plus 1 in $\mathbf{B}_2$ is 1. So for example, given that $A = 1,\ B = 1$ and $C = 0$, we find that on substitution of ' 1 ' for both ' A ' and ' B ', and ' 0 ' for ' C ', in the formula ' $(A \wedge B) \vee (A \wedge C)$ ', we obtain,

$$\begin{aligned} (A \wedge B) \vee (A \wedge C) &= (1 \wedge 1) \vee (1 \wedge 0) \\ &= 1 \wedge 0 \\ &= 1 \ . \end{aligned} \tag{2.2}$$

The **logical complementation** sign ' $-$ ' denotes the the truth function defined by the condition that $-0 = 1$ and $-1 = 0$, or by the "complementation table":

A	$-A$
1	0
0	1

Table 2.3: Logical Complementation for $\mathbf{B}_2$

An **atomic statement** is any statement, such as 'the sun is a star', 'Io is a moon of Jupiter' or 'the asteroid belt is between the orbit or Mars and the orbit of Jupiter', which expresses a property of something, or a relation between two or more things, is not a generality or a statement of existence, and contains no not's, and's and or's. A **Boolean statement** is any atomic statement, or is synonymous to the **negation** 'it is not the case that S' of a Boolean statement S, or if S and T are Boolean statements, is synonymous to the **conjunction** ' S and T ' or is synonymous to the **disjunction** ' S or T '.

We may calculate the truth value of any Boolean statement S by first *symbolizing* it, that is, finding a formula for its truth value. We may begin by putting S into a "standard form": replace each negation of a statement R which occurs in S by the statement 'it is not the case that R'. Then replace each conjunction (disjunction) of the statements Q and R occurring in S by the statements 'both Q and R' (either Q or R). Now replace each negation of of the form 'it is not the case that Q' in the resulting statement S_0 in standard form by $-Q$, every conjunction (disjunction) of statements Q and R in S_0 by $(Q \wedge R)$ (by $(Q \vee R)$). Finally, substitute for each atomic statement to occur in the resulting expression, a Boolean variable. Here you may choose any variable, provided only that separate occurrences in S of any two atomic statements must be replaced by the same variable if they are synonymous, but by different ones otherwise.

The resulting formula F then *symbolizes* S, and if we substitute '1' for every variable in F which symbolizes a true atomic statement in S and '0' for every variable which symbolizes a false atomic statement in S, then S will be true `iff` $F = 1$. We often write $[F]$ for any Boolean statement symbolized by F. Any two statements symbolized by the same formula will be synonymous.

2.2.1 Symbolizing Boolean Statements and Calculating their Truth Values*

It all started with George Boole[13], to whom it occurred to assign the number 1 to true statements and the number 0 to false statements as their **truth-values**.

For Boole, a false statement is a statement which is not true, and every statement which is not true is false, so that 1 and 0 are the only truth values. Boolean logic is **two-valued**.

Statements such as '2 is even', '$2 > 3$', '$2 + 3 = 5$', etc., that attribute properties to individual things or express relations between them, and which contain no "not's," "and's," "or's," generalities, or statements of existence, are called **atomic**, and are also **Boolean statements**.

We use **Boolean variables** 'A_1', 'A_2', 'A_3', ... (often abbreviated informally by arbitrary capital letters) to *symbolize* atomic statements, and every Boolean variable is an **atomic**(Boolean) **formula**. If the variable θ symbolizes an atomic statement Σ, then θ denotes the truth value of Σ, and we write $[\theta]$ for Σ. Then θ symbolizes Σ, and Σ **interprets** θ.

For instance, we might choose to symbolize the atomic Boolean statement 'Ganymede is a moon of Jupiter' by the letter 'G', which might abbreviate, say, the Boolean variable 'A_2'. In that case, we may write $[G]$ or $[A_2]$ for the Boolean statement above, which 'G' (i.e. 'A_2') symbolizes. In this case, 'G' denotes 1, so that $G = 1$, since Ganymede is in fact a moon of Jupiter.

If an atomic statement, say 'Ganymede is a moon of Jupiter' or a synonymous statement such as 'Ganymede is a satellite of Jupiter' appears more than once in a discussion, we must symbolize it by the same Boolean variable throughout, but we must symbolize an atomic statement not synonymous to 'Ganymede is

*This subsection presupposes sections 1.3.1 through 1.4.1 (pages 14-24).

a moon of Jupiter', say 'Ganymede is a comet', by another variable than the one used to symbolize [G].

If [X] is any Boolean statement, symbolized by the formula X, then 'it is not the case that [X] ' (or any synonymous statement) will be symbolized as $-X$, so that we may write [$-X$] for **Boolean statement** 'it is not the case that X '. For example if ' G ' symbolizes 'Ganymede is a moon of Jupiter', we must symbolize both of the statements 'it is not the case that Ganymede is a moon of Jupiter' and 'Ganymede is not a moon of Jupiter' by the **Boolean formula** ' $-G$ '. Boole would have written ' $1-G$ ' instead, and indeed that is the arithmetical value, since $1-1=0$ and $1-0=1$.

We symbolize statements synonymous to the **conjunction** 'both [X] and [Y] ' of [X] and [Y] by the **Boolean formula** $X \wedge Y$, so that the **Boolean statement** 'both [X] and [Y] ' is [$X \wedge Y$]. If [$X \wedge Y$] is true, then [X] and [Y] are both true, so that $X=1$ and $Y=1$. Boole wrote XY or instead of $X \wedge Y$, and indeed $X \cdot Y = 1$ if $X = Y = 1$, and $X \cdot 0 = 0 \cdot X = 0$. This covers all the possibilities, for X and Y denote truth values only, so that $X \cdot Y$ is in fact the arithmetical value of $X \wedge Y$.

Dually, any **Boolean statement** synonymous to 'either [X] or [Y] ' is the **disjunction** of [X] and [Y], and is true `iff` either $X=1$ or $Y=1$ (or both), and 'either [X] or [Y] ' is [$X \vee Y$], so that the **Boolean formula** $X \vee Y$ denotes the truth value of the Boolean statement [$X \vee Y$]. Boole wrote $X+Y$ instead of $X \vee Y$. The arithmetical value of $X \vee Y$ is $X+Y$ when either $X=0$ or $Y=0$ (or both), and differs from the arithmetical value $X+Y$ only if $X=Y=1$, when $X \vee Y = 1$.

The negation, conjunction, and disjunction signs are called **Boolean connectives** or just **connectives**. Because they denote truth functions, they are said to be **truth functional**. We introduce another connective ' $\Rightarrow$ ' in section 2.3.1 (page 36), where $X \Rightarrow Y = 1$ `iff` $X \leq Y$, and ' $\Leftrightarrow$ ' and ' $+$ ' in section 2.3.2 (page 39). Here $X+Y$ symbolizes (contrary to Boole[5]) 'either [X] or [Y] (but not both)', so $X+Y=1$ `iff` $X \neq Y$, and $X \Leftrightarrow Y = 1$ `iff` $X=Y$. The negation sign is a **unary connective**, and the others (the conjunction and disjunction signs and so on) are **binary connectives**. To summarize:

Statement	Symbolization	Arithmetical Value
it is not the case the [X]	$-X$	$1-X$
both [X] and [Y]	$X \wedge Y$	$X \cdot Y$
either [X] or [Y]	$X \vee Y$	$X+Y-(X \cdot Y)$

Table 2.4: Truth Functions

The third column gives the outputs of the three truth functions defined by Boole, in terms of ordinary arithmetical operations. The modern notation for the connectives we have introduced reflects the direct relationship between

[5]Boole used the ' $+$ ' for inclusive OR, whereas modern logicians now use the $\vee$ which stands for the latin word "vel" (which means 'or' in the inclusive sense).

Boolean operations and operations in more general structures used throughout mathematics called "lattices".

Indeed, $\mathbf{B}_2$ is a **lattice**, which we depict in figure (2.1), in which $X \vee Y$ is also largest or **maximum** of the numbers X and Y, and $X \wedge Y$ is the smallest or **minimum** of the numbers X and Y. $X \vee Y$ is then the highest and $X \wedge Y$ the lowest of the two numbers in the diagram[6]:

$$\begin{array}{c} 1 \\ | \\ 0 \end{array}$$

Figure 2.1: The Two Element Boolean Lattice $\mathbf{B}_2$

Of course if $X = Y$, then $X \wedge Y = X \vee Y = X = Y$.

PROBLEMS

1. Show that the above statement is true.

2.2.2 Calculating Truth Values: Some Examples

With Boole's rules for assigning truth values to statements synonymous to negations, conjunctions and disjunctions, you can always calculate the truth value of any Boolean statement, once you know the truth values of all the atomic statements which it contains.

To determine the truth value of a Boolean statement, first symbolize it. The Boolean formula X you get will be determined once you have chosen the Boolean variables to symbolize the atomic statements in $[X]$.

Now **substitute** ' 1 ' or ' 0 ' for each Boolean variable θ in X, replacing *every* occurrence of θ in X by ' 1 ' or ' 0 ', depending on whether θ symbolizes a true of false atomic statement. The resulting expression is a **Boolean term**, which contains no variables, and denotes a truth value which can be found by calculation. A cosmological example will serve:

[**U**]
```
The universe is unbounded
 and the earth is not at its center,
 and is expanding or contracting, but not both.
```

Since Newton's time, we have known that the universe is unbounded and has no center, so the earth is not at its center, and since the 1920's, we have also known that the universe is expanding, and not static.

Let B be the truth value of 'the universe is unbounded', C the truth value of 'the earth is at the center of the universe', E the truth value of 'the universe is expanding', and R the truth value of 'the universe is contracting'.

[6] We present an exact definition of lattices in Chapter 7 (page 114).

The entire statement is a Boolean statement, the conjunction of 'the universe is unbounded and the earth is not at the center of the universe' and 'the universe is expanding or contracting (but not both)'.

The truth value of 'the universe is unbounded and the earth is not at its center' is,

$$B \wedge -C \ , \tag{2.3}$$

while the truth value of 'the universe is expanding or contracting (but not both)' is,

$$(E \vee R) \wedge -(E \wedge R) \ . \tag{2.4}$$

Putting all this together, the entire statement [U] is symbolized by the conjunction U of the formulas (2.3) and (2.4),

$$U : \qquad (B \wedge -C) \wedge \Big((E \vee R) \wedge -(E \wedge R)\Big) \ . \tag{2.5}$$

Since the universe is unbounded and the earth is not at its center, and is expanding and not contracting, $B = 1$, $C = 0$, $E = 1$ and $R = 0$. These equations form a **solution** to the equation $U = 1$, for on substituting ' 1 ' for ' B ', ' 0 ' for ' C ', ' 1 ' for ' E ' and ' 0 ' for ' R ' in (2.5), the resulting term

$$(1 \wedge -0) \wedge ((1 \vee 0) \wedge -(1 \wedge 0)) \ , \tag{2.6}$$

reduces on calculation to 1, by substituting successively on the right ' 1 ' for ' -0 ', ' $(1 \vee 0)$ ' and ' 0 ' for ' $(1 \wedge 0)$ '; then ' 1 ' for ' -0 ' and ' $(1 \wedge 1)$ ' directly below; and ' 1 ' for ' $(1 \vee 1)$ ' below that; and finally ' 1 ' on the last line, which completes the calculation:

$$\begin{aligned} (1 \wedge -0) \wedge \Big((1 \vee 0) \wedge -(1 \wedge 0)\Big) &= (1 \wedge 1) \wedge (1 \wedge -0) \\ &= 1 \wedge (1 \wedge 1) \\ &= 1 \wedge 1 \\ &= 1 \ . \end{aligned} \tag{2.7}$$

The set $B = 1$, $C = 0$, $E = 1$, $R = 0$ of **atomic equations** is a **solution** to the **Boolean equation** $U = 1$. This set is **compatible**: no variable in it appears twice, and so it determines a specific **substitution** of constants for variables in U: we write $U[1/B, 0/C, 1/E, 0/R]$ for the term (2.6) that results. That term is on the left in the first of the above equations. So $U[1/B, 0/C, 1/E, 0/R] = 1$, and [**U**] is true.

A century ago, all agreed that the universe is unbounded and the earth is not at its center, but nobody guessed that the universe is expanding or contracting. They would have agreed that $B = 1$ and $C = E = R = 0$. The resulting substitution and calculation now makes [**U**] false:

$$\begin{aligned} U[1/B, 0/C, 1/E, 0/R] &= (1 \wedge -0) \wedge \Big((0 \vee 0) \wedge -(0 \wedge 0)\Big) \\ &= (1 \wedge 1) \wedge (0 \wedge -0) \end{aligned}$$

$$
\begin{aligned}
&= 1 \wedge (0 \wedge 1) \\
&= 1 \wedge 0 \\
&= 0\ . \qquad (2.8)
\end{aligned}
$$

Neither logic nor mathematics alone can inform us whether [**U**] is true: it could be true and it could be false, depending on the way things are.

But 'the universe is either expanding or not expanding' must always be true, regardless of the state of the universe, for it is symbolized by the formula '$E \vee -E$', which is a **tautology**: it is true no matter what truth values the variables in the formula denote. We may write $\models X$ when X is a tautology. But 'the universe is both expanding and not expanding', which is symbolized by '$E \wedge -E$', can never be true, whatever the state of the universe. It is symbolized by a formula which is a (truth functional) **contradiction**: it is false no matter what truth values the variables in it denote, so that its negation is a tautology.

Of course, tautologies and contradictions are of no scientific interest to physicists because they cannot tell us anything about the physical world. But they are of central interest in logic. In particular, the problem of determining whether a given set of mathematical propositions entails a contradiction is much deeper than it appears.

PROBLEMS

1. The weather forecast was: it will be cold and either sunny or windy. As it turned out, it was cold and sunny but not windy.

   - Let S be the truth value of 'it will be sunny'.
   - Let W be the truth value of 'it will be windy'.
   - Let C be the truth value of 'it will be cold'.

   So $C = S = 1$ and $W = 0$. Determine by calculation whether the weather forecast was correct.

2. Determine by calculation which of the following formulas are tautologies and which are contradictions:

   (a) $(A \wedge B) \vee (-A \wedge -B)$

   (b) $(A \vee B) \wedge (-A \wedge -B)$

   (c) $(A \wedge B) \vee (-A \wedge -B)$

   You will need consider four cases:

   Case 1: $A = 1$ and $B = 1$
   Case 2: $A = 1$ and $B = 0$
   Case 3: $A = 0$ and $B = 1$
   Case 4: $A = 0$ and $B = 0$

2.3 Material Implication, Equivalence and The Exclusive 'Or'

2.3.1 Truth Functional or Boolean Implication

Any statement of the form 'if $[X]$ then $[Y]$ ' is a an **implication** or a **conditional**, and its **antecedent** is $[X]$ and its **consequent** is $[Y]$.

Here are some other equivalent forms of the conditional 'if $[X]$ then $[Y]$'; in every case, the antecedent is $[X]$ and the consequent is $[Y]$.

- If $[X]$ then $[Y]$
- $[X]$ implies $[Y]$
- $[X]$ only if $[Y]$
- $[Y]$ if $[X]$
- $[Y]$ provided that $[X]$
- $[X]$ is a sufficient condition for $[Y]$
- $[Y]$ is a necessary condition for $[X]$

By definition, implication in Boolean logic is always **truth functional**, (and also called **material**) in the sense that the truth value of ' $[X]$ implies $[Y]$ ' is determined, once the truth values of $[X]$ and $[Y]$ are known. ' $[X]$ implies $[Y]$ ' is to be false if the antecedent is true and the consequent false, and true otherwise, i.e. if the antecedent is false or the consequent true.

If we let $X \Rightarrow Y$ be the truth value of the material conditional 'if $[X]$ then $[Y]$ ', then ' $\Rightarrow$ ' will be a truth functional connective, which we may use to construct a conditional formula from the formulas X and Y, with X the antecedent and Y the consequent. The definition of the material conditional then gives us the following table or **matrix** for $\Rightarrow$:

$\Rightarrow$	0	1
0	1	1
1	0	1

Table 2.5: Matrix for Material Conditional

Put another way,

$$\Rightarrow \qquad X \Rightarrow Y = 1 \texttt{ iff } X \leq Y \ .$$

Material implication is not strictly needed, however, since $X \Rightarrow Y$ always has the same truth value as $-X \vee Y$ and as $-(X \wedge -Y)$. For the equations $X \Rightarrow Y = -X \vee Y$ and $X \Rightarrow Y = -(X \wedge -Y)$ are **identities**: they hold for all values of the variables in X and Y and so hold for all values of X and Y.

Take, for example, the equation $A \Rightarrow B = -A \vee B$. Since,

$$\begin{aligned} 1 \Rightarrow 1 &= -1 \vee 1 = 1 \ , \\ 1 \Rightarrow 0 &= -1 \vee 0 = 0 \ , \\ 0 \Rightarrow 1 &= -0 \vee 1 = 1 \ , \\ 0 \Rightarrow 0 &= -0 \vee 0 = 1 \ . \end{aligned} \tag{2.9}$$

The equation $A \Rightarrow B = -A \vee B$ is indeed an **identity**: the equation holds no matter what the (truth) values of the Boolean variables occurring in the equation. We may write $A \Rightarrow B \equiv -A \vee B$ to assert that the given equation is an identity, and we also say that the formulas ' $A \Rightarrow B$ ' and ' $-A \vee B$ ' are (**truth functionally**) **equivalent truth functionally equivalent**.

You get the **converse** of a conditional $X \Rightarrow Y$ by exchanging X and Y. A conditional is not in general equivalent to its converse, since a conditional differs in truth value from its converse whenever its antecedent and consequent have different truth values. But the material conditional you get from $X \Rightarrow Y$ by taking the converse $Y \Rightarrow X$ and then negating both Y and X to obtain $-Y \Rightarrow -X$, is truth functionally equivalent to the original conditional $X \Rightarrow Y$, for by calculation we find that these two inequalities hold for all Boolean formulas X and Y:

$$X \Rightarrow Y \leq -Y \Rightarrow -X \ , \tag{2.10}$$

$$-Y \Rightarrow -X \leq X \Rightarrow Y \ , \tag{2.11}$$

which is to say that,

$$X \Rightarrow Y \equiv -Y \Rightarrow -X \ . \tag{2.12}$$

This is known as the **principle of contraposition**.

In accordance with common practice, we will abbreviate formulas like $(X \wedge Y) \Rightarrow Z$ or $X \Rightarrow (Y \vee Z)$ thus: $X \wedge Y \Rightarrow Z$ and $X \Rightarrow Y \vee Z$. The convention is that if the antecedent or consequent of a conditional is a conjunction or disjunction, we may dispense with the parentheses around the conjunction or disjunction.

So in the *absence* of parentheses enclosing a conditional sign, we assume that any parentheses we need to resolve ambiguity should not enclose the conditional sign but should go around the antecedent or consequent, as the case may be, if the antecedent or consequent is a conjunction or disjunction. For example, by our convention, we *need* the parentheses in the formulas $X \wedge (Y \wedge \Rightarrow \wedge Z)$ and $(X \Rightarrow Y) \vee Z$, but not in $X \wedge Y \Rightarrow Z$ and $X \Rightarrow Y \vee Z$, which are informal abbreviations for $(X \wedge Y) \Rightarrow Z$ and $X \Rightarrow (Y \vee Z)$, respectively. A similar convention holds for **biconditionals** or "material equivalences" $X \Leftrightarrow Y$, which we introduce in the next section.

Conditionals in ordinary language are in general not material, as we shall see. Then why use material implication at all? The answer is that while material implication was never intended as a serious analysis of the use of the conditional in ordinary language discourse, it does give an adequate account of its use in *mathematical* or *Boolean* reasoning.

But it is especially important to the analysis of **formal arguments**, which consist of a set of **assumptions** and a **conclusion**. The assumptions and the conclusion are formulas, which symbolize Boolean statements. Suppose that there are n assumptions, say G_1, G_2, and ... and G_n. We may write either Γ or $G_1, G_2, \ldots, G_n$ for the set consisting of these formulas, and if H is another formula, we may write either $G_1, G_2, \ldots, G_n, H$ or Γ, H for the set $\Gamma \cup \{H\}$. The formal argument $\Gamma \therefore Y$ consists of a set Γ of formulas called "the assumptions of the argument" and a single formula Y called its "conclusion". We say that the formal argument $\Gamma \therefore Y$ or $G_1, G_2, \ldots, G_n \therefore Y$ is **valid**, and we write $\Gamma \models Y$, `iff` $G_1 \wedge G_2 \wedge \ldots \wedge G_n \Rightarrow Y$ is a tautology.

If given the truth values $A = 1 = c_1, \ldots, A_n = c_n$ where $c_1, \ldots, c_n$ denote the truth values given for the Boolean variables 'A_1', ... 'A_n' that occur in the argument $\Gamma \therefore Y$, it turns out that on substitution, the assumptions $G_1, \ldots, G_n$ are all true while Y is false, the argument $\Gamma \therefore Y$ is not valid, and $\Gamma \not\models Y$. The set $G_1 = 1$, $\ldots G_n = 1$, $Y = 0$ of simultaneous equations then has a solution,

$$\begin{aligned} A_1 &= c_1 \ , \\ &\vdots \\ A_n &= c_n \ , \end{aligned}$$

which serve as a **counterexample** to the claim that the argument $\Gamma \therefore Y$ is valid. An argument $\Gamma \therefore Y$ is therefore valid `iff` it has no counterexample. For instance, argument $A, B \therefore A$ is valid, while $A \vee B \therefore A$ is not, for the equation set $A = 0$, $B = 1$ is a counterexample to the assertion that $A \vee B \models A$.

In the next chapter, we will present various methods for determining whether a given formal argument is valid, whether two formulas are equivalent, and for solving related problems. In section 2.4.1 (page 40), we will try to explain why, in view of the above discussion, the conditional used in Boolean reasoning *must* inevitably be the material conditional.

PROBLEMS

1. Show by calculation that if $B = 0$ and $(B \Rightarrow L) \Rightarrow E = 1$ then $E = 1$.

2. Determine which of the following formulas are truth functionally equivalent to A and which are equivalent to $A \vee B$:

   (a) $(A \Rightarrow B) \Rightarrow A$

   (b) $(A \Rightarrow B) \Rightarrow B$

   (c) $-A \Rightarrow A$

3. Elimination of Parentheses

   (a) Eliminate parentheses in the following conditional, in accordance with the convention described in the next to last paragraph of this section:

$$\Big(A \wedge (B \vee C)\Big) \Rightarrow \Big((A \wedge B) \vee C\Big) \ .$$

(b) By the convention for eliminating parentheses in conditionals referred to in (a), the expression

$$A \vee (B \wedge C) \Rightarrow A \vee B$$

is an informal abbreviation for one and only one formula. What is the formula thus abbreviated when you restore parentheses?

2.3.2 Material Equivalence and Exclusive Disjunction

There are two other connectives we use from time to time. They are not strictly needed either because we can define them in terms of the connectives we already have, and we don't use them often enough to justify introducing special inference rules for them. We first introduce material equivalences.

The statement 'the water is boiling if it's 212°F or above' is the converse of 'the water is boiling only if it's 212°F or above'. If you want to say that both of these are true, you may say 'the water is boiling if and only if it's 212°F or above'. Instead of writing, say,

$$(A \Rightarrow B) \wedge (B \Rightarrow A)\ , \tag{2.13}$$

we simply write '$A \Leftrightarrow B$'.

Any formula $X \Leftrightarrow Y$ is a **biconditional** or a **material equivalence** which is true `iff` both $X \leq Y$ and $Y \leq X$. That is, $X \Leftrightarrow Y$ is true `iff` X and Y have the *same* truth value:

$$\Leftrightarrow \qquad X \Leftrightarrow Y = 1 \ \ \texttt{iff} \quad X = Y\ .$$

On the other hand, if you want to say that the water is boiling or it's cooler than 212°F (but not both), this is the same as saying 'the water is boiling if and only if it's not cooler than 212°F'. We may write '$B + C$' for its truth value. The formula $B + C$ is called an **exclusive disjunction**; it excludes the possibility that both disjuncts are true[7].

An exclusive disjunction $X + Y$ is true `iff` X and Y take *different* truth values:

$$+ \qquad X + Y = 1 \ \ \texttt{iff} \quad X \neq Y\ .$$

We may call $X + Y$ the **algebraic sum** of X and Y, to distinguish it from the Boolean sum. Both are the same as the arithmetical sum X plus Y when X and Y are truth values, not both of which are 1, but all three are distinct from each other when $X = Y = 1 : 1 \vee 1 = 1$, $1 + 1 = 0$ but 1 plus 1 is 2. In general, whether one and one is zero, one or two depends on how you define one and one.

Here are the matrices for $\Leftrightarrow$ (material equivalence) and $+$ (exclusive disjunction):

[7] In computer languages this is often symbolized as XOR, the exclusive OR

$\Leftrightarrow$	0	1
0	1	0
1	0	1

$+$	0	1
0	0	1
1	1	0

Note that we can define negation in terms of material implication and the Boolean constant ‘ 0 ’, since $X \Rightarrow 0$ is truth functionally equivalent to $-X$. We may also define negation in terms of exclusive disjunction and the Boolean constant ‘ 1 ’, since $X + 1$ is truth functionally equivalent to $-X$.

PROBLEMS

1. Prove the two claims made in the preceding paragraph.
2. Define regular "inclusive" disjunction in terms of exclusive disjunction and conjunction.

2.4 About The Truth Functional or Material Conditional*

Overview

There are good reasons for using the material conditional in Boolean logic, and its extension to first order logic and beyond. All such systems adequately represent mathematical reasoning, but not reasoning in informal or natural languages, which is a far more complicated affair.

Basically, in mathematical reasoning, it is routine in proofs to assume the principle that $\Gamma,\ X \models Y$ `iff` $\Gamma \models X \Rightarrow Y$. From this it follows that $X \Rightarrow Y$ must be the material conditional, as we show in section 2.4.2 (page 42). If $X \Rightarrow Y$ is the material conditional, the following assertions hold:

$$\begin{aligned} -X &\models X \Rightarrow Y \quad , \\ -(X \Rightarrow Y) &\models X \quad , \\ X \Rightarrow -Y &\models Y \Rightarrow -X \quad , \\ X \Rightarrow Y,\ Y \Rightarrow Z &\models X \Rightarrow Z \quad . \end{aligned} \tag{2.14}$$

Much ink has been spilled in defending the claim that they always hold in ordinary discourse, mostly by means of carefully chosen examples. Yet plausible counter- examples to such claims abound. We present some of these in section 2.5.1 (page 45), which are based on the assertions (2.14).

2.4.1 Why the Conditional of Boolean Reasoning is the Material Conditional

Let $X_1, \ldots X_k$, H and C be formulas which symbolize the Boolean statements $[X_1], \ldots, [X_k]$, $[H]$ and $[C]$, respectively. For the sake of convenience, we will

*This part may be skipped without loss of continuity.

write X for the conjunction $X_1 \wedge \ldots \wedge X_k$ of the X_i, where $i = 1, \ldots, n$. We may take the X_i to be the assumptions or equivalently X to be the assumption of a formal argument.

A common way of proving a conditional to follow from $[X]$ is to add another assumption $[H]$ (called a **hypothesis**) to $[X]$. If in a series of steps you succeed in deducing $[C]$, then the statement 'if $[H]$ then $[C]$' should follow from $[X]$ *without* the hypothesis $[H]$. This style of argument goes back at least to Euclid, and we call it "conditional proof" or `CP`.

Let $H \Rightarrow C$ be the truth value of 'if $[H]$ then $[C]$', where for the moment, (while taking all statements to be either true or false) we do *not* assume that $H \Rightarrow C$ symbolizes a material conditional, or determines any truth function at all.

If we take 'the conclusion $[C]$ follows from the assumptions $[X]$ and $[H]$', to mean that $[C]$ cannot be false when $[X]$ and $[H]$ are both true, and abbreviate this assertion as follows:

$$[X],\ [H] \models [C]\ ,$$

then we have the following principle, which we call **conditional proof** (`CP`):

CP `If` $[C]$ `follows from` $[X]$ `and` $[H]$`,`
$[H \Rightarrow C]$ `follows from` $[X]$`.`
`That is, if` $X, H \models C$`, then` $X \models H \Rightarrow C$`.`

Another principle of proof, known as **modus ponens** (`MP`), also goes back at least to Euclid and was stated explicitly by the Stoics shortly after:

MP $[C]$ `follows from` $[H]$ `and` $[H \Rightarrow C]$`.`
`That is,` H`,` $H \Rightarrow C \models C$`.`

It follows from `MP` that a conditional with a true antecedent and a false consequent cannot be true. Thus the bottom left entry in the matrix for $H \Rightarrow C$ must be 0:

$\Rightarrow$	0	1
0		
1	0	

Now it is trivial that $C, H \models C$. So by `CP`, $C \models H \Rightarrow C$, and a conditional with a true consequent cannot be false. Therefore, both entries in the right column must be 1:

$\Rightarrow$	0	1
0		1
1	0	1

It is also clear that $[C]$ cannot be false while $[H]$ and $[-H]$ are both true, (because $[H]$ and $[-H]$ can never both be true), so that $-H, H \models C$, and by `CP` again, $-H \models H \Rightarrow C$. Thus a conditional with a false antecedent must be true, and both entries in the top row must be 1:

$\Rightarrow$	0	1
0	1	1
1	0	1

We conclude that a conditional that meets the requirements `MP` and `CP` is truth functional, and is in fact the material conditional. Many objections have been raised against material implication, by appealing to ordinary language to produce counter-intuitive results, some of which we discuss in the next section. But none of these examples involve *Boolean* or *mathematical* reasoning, which is our real concern, and within that context, the material conditional works perfectly well.

PROBLEMS

1. Assume that $A \Rightarrow B$ is a truth functional conditional. Show that if $X, A \models B$, then $X \models A \Rightarrow B$. (Hint: show that any counterexample to the claim $X \models A \Rightarrow B$ is also a counterexample to the assertion that $X, A \models B$).

2. Show that both the assertion proved in the above problem, together with the assertion $A, A \Rightarrow B \models B$ imply:

$$X, A \models B \texttt{ iff } X \models A \Rightarrow B \ .$$

   (Hint: first show that if $\Phi \subseteq \Gamma$ and $\Phi \models X$, then $\Gamma \models X$).

3. Show that all of the following assertions are true:

$$-X \models X \Rightarrow Y \ , \tag{2.15}$$
$$-(X \Rightarrow Y) \models X \ , \tag{2.16}$$
$$X \Rightarrow -Y \models Y \Rightarrow -X \ , \tag{2.17}$$
$$X \Rightarrow Y, Y \Rightarrow Z \models X \Rightarrow Z \ . \tag{2.18}$$

2.4.2 The Material Conditional in Ordinary Reasoning: Some Pitfalls

The assertion (2.15-2.18) of problem 3 above work perfectly well in mathematical proofs, and consequently so does the resulting identification of conditionals with material conditionals. But by and large material implication does not capture the meanings which the conditional takes on in ordinary non-mathematical discourse. For example, if you interpret the following conditional as a truth functional implication, it is *true*:

```
If I jump off the Eiffel tower tomorrow,
I will not fall off.
```

Because I'm not going to jump off the Eiffel tower tomorrow, the antecedent of this conditional is false, and also because I'm not going to fall off the Eiffel tower tomorrow, the consequent is true. On both counts, the conditional is true.

This example is an instance of the "paradoxes" of material implication: it is supposed to be puzzling that the above conditional should be true just because it has a false antecedent, or just because it has a true consequent. But this is only "paradoxical" if you think that all ordinary language conditionals are material.

Since they aren't, it is not surprising that material conditionals don't always work when used in ordinary non- mathematical reasoning. Here are three examples of arguments which are valid if the conditional involved is taken as a material conditional; the first of these follows from the fact that any false material conditional must have a true antecedent (as well as a false consequent)[8]:

> `The statement 'if Napoleon was born in San Francisco, then he was born in Corsica' is false. Therefore, Napoleon was born in San Francisco and not in Corsica.`

The argument `if` $[X]$ `then not`$-[Y] \therefore [Y]$ `then not`$-[X]$ is valid for material conditionals and is a form of the principle of "contraposition". It is generally taken to be valid in ordinary discourse, but fails in this instance:

> `If it rained, it didn't rain hard. Therefore, if it rained hard, it didn't rain.`

And finally, here is another argument valid for material conditionals, and widely assumed valid in ordinary reasoning, and is called **hypothetical syllogism**:

> `If` $[X]$ `then` $[Y]$, `If` $[Y]$ `then` $[Z]$ $\therefore$ `If` $[X]$ `then` $[Z]$.

Yet this instance of hypothetical syllogism is fallacious:

> `If Clinton wins the election, Dole will retire to private life. If Dole dies before the election, Clinton will win it. Therefore, if Dole dies before the election, he will retire to private life.`

Of course, some ordinary language conditionals can be taken to be truth functional conditionals. But in ordinary discourse, there is generally some sort of connection between the antecedent and the consequent. For example, if the antecedent in some way causes the consequent, the consequent can't be true before the antecedent is true.

Thus, we might assent to 'if Dole dies before the election, he won't retire after it', but its contraposition 'if Dole retires after the election, he won't die before it' is less clear, for it looks as if what he does after the election is a cause of what happened before it. And we've only scratched the surface of such problems.

Consequently, mustering a few examples (often made up) of ordinary language conditionals which are arguably truth functional) doesn't prove much, for

[8] The second example is due to Adams (1988)[1], while variants of the first and third examples are in Jeffrey (1991)[38].

the exact meaning of the conditional when used in ordinary discourse is rarely made explicit, and so it's up to philosophers to tell you what you're *really* saying. But any convincing analysis of conditionals in ordinary language must run deeper, and the heart of the problem has yet to be found.

But that's not our problem. The object of our study is *Boolean* reasoning. And the *real* reason that material implication is used in Boolean reasoning is because it's convenient and useful to do so. When it is applied to describe reasoning (except for purposes of illustration), it describes mathematical but not ordinary language reasoning.

But consensus on such issues is rare, and if it is ever achieved, the question will cease to be philosophical. If not, "Critical Thinking", that pale reflection of Leibniz' universal characteristic and a staple of the college freshman curriculum, will never be a "universal characteristic", much less a calculus or a science, but will remain an art.

PROBLEMS

1. Show that the following arguments are valid, on the assumption that all conditionals are material conditionals:

   (a) If I will be cured if I have faith, then there is a god. I don't have faith. Therefore, there is a god.

   (b) If I will be cured only if I have faith, then there is a god. I won't be cured. Therefore, there is a god.

2.5 Boolean Syntax

Overview

Boolean *languages* determine what formulas there are in a given Boolean logic. In addition to the connectives and the parentheses (which are "punctuation symbols"), every Boolean language has among its symbols at least one Boolean variable or else Boolean constants ' 0 ' and ' 1 ' which denote the truth values 0 and 1. The Boolean variables range over the set $\{0, 1\}$ of truth values, which are the elements $\mathbf{B}_2$, the two element Boolean algebra.

The rules for constructing formulas (the *formation rules*) are *syntactical* (i.e. "grammatical") notions, and we present them in section 2.5.2 (page 47). The structure of formulas is important, because the validity and provability of a formal argument depends the structure of the assumptions and conclusion. For example, the argument $A \wedge (B \vee C) \therefore A$ is valid, but the argument $(A \wedge B) \vee C \therefore A$ is not, and this is because the assumptions '$A \wedge (B \vee C)$' and '$(A \wedge B) \vee C$' have different structures: the first one is a conjunction and the second a disjunction, which is true if $A = 0$ and $C = 1$.

2.5.1 What is a Boolean Language?

The way in which Boolean formulas are given truth values constitute the **semantics** of Boolean logic. Truth is inherently a semantical notion, while the grammar or **syntax** of formal Boolean logic consists of the rules whereby Boolean formulas are constructed. These determine the *structure* of formulas. Thus validity is a semantical notion, because the validity of an argument depends both on the structure of the assumptions and conclusion of the argument, and their truth values in all possible cases, while provability is a syntactical notion, because the provability of an argument depends only on the structure of the formulas involved.

In these three sections, we are concerned with the syntax of those *formal* languages which we'll call "Boolean languages". Every Boolean language has **atomic formulas**, which are either Boolean variables or Boolean constants. Those we consider either have no Boolean variables but both of the Boolean constants '0' and '1', or have at least one Boolean variable, and perhaps either of the Boolean constants '0' or '1' (or both). They also have both the left and right parentheses '(' and ')' (its "punctuation symbols"), and at least the connectives '$-$', '$\wedge$', '$\vee$' and '$\Rightarrow$'. The **logical symbols** of a Boolean language are its Boolean constants and its connectives, and its **non-logical** symbols are its Boolean variables. Its **alphabet** is the set of all its symbols, whether logical, non-logical or punctuation symbols.

The Boolean languages that we consider differ in the logical and non-logical symbols they have. We define Bv_n to be the set $\{`A_1', \ldots, `A_n'\}$ consisting of the first n Boolean variables, and Bv_∞ to be the denumerable set

$$Bv_1 \cup Bv_2 \cup \ldots = \{`A_1', `A_2', \ldots\} \; . \tag{2.19}$$

By L_0, we will mean any language which has no Boolean variables, but has both the Boolean constants '0' and '1', which we often informally abbreviate as '$\bot$' and '$\top$', respectively. And to avoid clutter, we informally replace the variables 'A_1', 'A_2', ... by arbitrary letters, or just by 'A', 'B', ..., if we have no particular symbolization in mind[9].

L_n is any Boolean language, whose set of Boolean variables is Bv_n. All such languages, together with L_0 languages, are **finitely generated**. Any L_∞ language is **infinitely generated**. We will not see any infinitely generated language until section 4.4.3 (page 46).

[9]We put single quotes around specific expressions of a formal language (but not around **metalinguistic** variables $X, Y, Z, \ldots$ ranging over expressions or metalinguistic compounds of them), when discussing or *mentioning* them, while the object that the expression denotes is given by the expression inside the quotes. Thus '0' is the **numeral** which names or denotes the number 0. So $\{0, 1\}$ is the set consisting of the numbers 0 and 1, while $\{`0', `1'\}$ is the set consisting of the numerals '0' and '1'. On the other hand, we adopt the usual convention of writing, for example, $A, -A \wedge B$ for $\{`A', `-A \wedge B'\}$ when referring to the formal argument $A, -A \wedge B \therefore B$, the set of assumptions of which is $\{`A', `-A \wedge B'\}$, and the conclusion of which is 'B'. The symbol 1 is *used* in the sentence "1 is a number", but mentioned in " '1' is a numeral". Single quotes or the lack of them in a sentence changes the meaning. For instance 'my name is arbitrary' means that it is arbitrary what my name is, while "my name is 'arbitrary'" says that 'arbitrary' is my name.

The non-atomic formulas of any Boolean language L are **truth functional compounds**. Of these, $-X$ is a **negation**, where X is a formula which is the **immediate subformula** of X, and the negation sign '$-$' is the **main connective** of $-X$. If Y is also a formula of L, $(X \wedge Y)$ is a **conjunction**, the **main connective** of which is the conjunction sign '$\wedge$' between X and Y, and whose **left** and **right immediate subformulas** are X and Y. Dually, $(X \vee Y)$ is a **disjunction**, whose **left** and **right immediate subformulas** are again X and Y, between which is the **main connective** of $(X \vee Y)$. The **main connective** of the **conditional** $(X \Rightarrow Y)$ is the conditional sign '$\Rightarrow$' between the X and the Y, and its left and right **immediate subformulas** are the antecedent X and the consequent Y.

To summarize, if $\star$ is any binary connective, the **truth functional compound** $(X \star Y)$ has X and Y as its **left** and **right immediate subformulas**, and the compound's **main connective** is between the two.

A **formula** of L is then any of its atomic formulas or truth functional compounds. If Af is the set of *atomic* formulas of a Boolean language, then we call $F(Af)$ the set of its formulas. A **Boolean term** of L is any L_0-formula, with the same connectives as L, while an **open Boolean term** of L_n (of L_∞) is a term of an L_n-language (of an L_∞ language) with the same connectives as L_0 (as L_∞) which has one or two Boolean constants. Any formula X is a subformula of itself, and any immediate subformula of X is a subformula of X, together with any subformula of a subformula of X.

Two things are worth pointing out. First, a connective can *occur* in more than one place in a formula. For instance, '$\wedge$' occurs in two places in the formula '$A \wedge (B \wedge C)$'. But it's main connective is the leftmost one, between its immediate subformulas 'A' and '$B \wedge C$'. Its *position* between them is what makes '$A \wedge (B \wedge C)$' a conjunction. Strictly speaking, then, we ought to call the main connective the "main connective occurrence" which is awkward and still not accurate. So we defer to common usage.

Secondly as you have noticed, every truth functional compound which has a binary main connective is enclosed in parentheses. By convention or common consent, we nearly always informally omit them, save in special cases which occur only in section 2.5.4 (page 52) and section 8.5.2 (page 161) below. The only function of outer parentheses is to simplify the definition of formulas.

PROBLEMS

1. Suppose that R is a set of formulas of a Boolean language L_2, no two members of which are equivalent. What is the largest number of formulas that R can have? (Hint: every formula of L_2 determines a truth function $f : \{0,1\}^2 \rightarrow \{0,1\}$, of which there are sixteen).

2. Can '$A \wedge B$' be a formula of L_3? Of L_1? Explain.

2.5.2 Boolean Formation Rules

The definition of formulas given above may at first glance seem circular, because it seems to define formulas in terms of formulas, but in fact it tacitly involves constructing formulas step by step, starting with atomic formulas. This process of construction follows syntactical rules which are precise, and have no exceptions. They are *much* simpler than the grammar of any natural language, and they allow an exact definition of what a Boolean formula is.

To this end, we first define expressions. An **expression** of a Boolean language L is then a string of one or more symbols of L. If E and F are expressions, so is EF, which is the string E followed by the string F. The formulas of L are given by the formation rules presented in the next section, so that the formulas of L are determined once its symbols are given. So we may identify a **Boolean language** L either with its alphabet of symbols or with its set Af of atomic formulas, or with $F(Af)$.

We may then state the rules thus: an expression of a Boolean language L is a **formula** `iff` its being so follows from the following **formation rules**:

(i) If E is an atomic formula, then E is a formula.

(ii) IF E is a formula, so is $-E$.

(iii) If E and F are formulas, so is $(E \star F)$, where $\star$ is a binary connective.

PROBLEMS

1. (a) What is the antecedent and what is the consequent of:

$$\Big(A \Rightarrow (B \Rightarrow C)\Big) \Rightarrow \Big((A \Rightarrow B) \Rightarrow (A \Rightarrow C)\Big) \ ? \tag{2.20}$$

How many left parentheses has the antecedent got? How many right parentheses? How many left and how many right parentheses has the consequent got?

(b) Indicate which occurrence of ‘ $\Rightarrow$ ’ is the main connective of:

$$\Big((A \Rightarrow B) \Rightarrow A\Big) \Rightarrow A \ . \tag{2.21}$$

How many left parentheses has the antecedent got? How many right parentheses? How many left and right parentheses has the consequent got?

2. What are the subformulas of formula (2.20)? What are the subformulas of the formula in (2.21)? What formulas are subformulas of *both* these formulas?

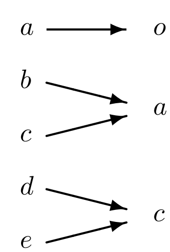
2.5.3 Trees and their Uses*

From this point on, we will find many uses for trees. We use analysis trees to test for equivalence, and valuation trees to show the truth values of the subformulas of a formula, given the truth values of its atomic subformulas. Valuation trees come from subformula trees, and game trees are valuation trees which have been "pruned". We discuss subformula trees and valuation trees in the next section, and the others in the rest of the chapter, and in the next we introduce equation trees and truth trees. For us, as for the National Forest Service, trees have multiple uses. But they "grow" in different ways and have different uses, much as redwood trees and oak trees grow differently and are used differently.

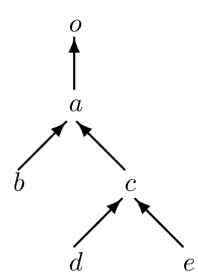
The trees we are about to discuss grow by syntactical rules, which are applied to the formulas or equations at given points or **nodes** in the tree to extend it further.

But what is a tree? There are many ways of defining a tree. Here is one way: a tree $\mathbf{T}$, the set of nodes of which is a non-empty set T, one of which is its **origin** o. If o is the only point in T, then $\mathbf{T} = T = \{o\}$. If $\mathbf{T}$ has more than one point, and is finite, it is a function $p : (T - \{o\}) \rightarrow T$, the field of which is T. Thus, o is in the range of p, but is not in its domain.

For instance, let p have the internal diagram,



which we may telescope into a **tree diagram** with the origin o on top:



The $f-$image of any node t distinct from o is the **predecessor** of t, i.e. $p(t)$ is the predecessor of t. A node which is the predecessor of more than one point in T is a **junction**, and a node which is not the predecessor of any point is

*This part may be skipped without loss of continuity.

an **end**. If t is a node which is not the origin, then it is a **successor** of $p(t)$. A node which has more than one successor is a **junction**, while a node that has no successors is an **end**. The origin of course is the only node that has no predecessor. A **path** through a tree $\mathbf{T} = p : (T - \{o\}) \to T$ is a left inverse of p. For instance, take the function $f : \{o, a, c\} \to (T - \{o\})$, the field of which is $\{o, a, c, d\}$, whose internal diagram displayed below in tree form with o on top is a path through $\mathbf{T}$ and is a left inverse of p :

As long as a tree is finite and its only node is o, we take it to consist of o alone. If it has more than one node, we may identify it with *any* function $p : (T - \{o\}) \to T$, the range of which contains o.

We also want to consider denumerable trees $\mathbf{T}$, where T is denumerable. But we only want to consider certain kinds of denumerable trees, and this requires certain restrictions on p. These restrictions are automatically satisfied if T is finite.

We require first that $\mathbf{T}$ be **finitely generated**, in the sense that every junction have finitely many successors. The other requirement is this: any subset of T that contains o and contains all successors of any of its members, is T itself.

A tree that has no junctions is a **sequence**, and is **finite** if it has an end, and is **infinite** otherwise.

The **natural number sequence** is an infinite sequence with origin 0. Its internal diagram is:

$$\begin{array}{ccc} 0 & \longrightarrow & 1 \\ 1 & \longrightarrow & 2 \\ 2 & \longrightarrow & 3 \\ & . & \\ & . & \\ & . & \end{array}$$

which we may abbreviate as: $0 \to 1 \to 2 \to \ldots$. When presented as a tree with the origin displayed at the top, it looks like this:

0
1
2
.
.
.

The predecessor function p of the natural number sequence **N** is a one-to-one correspondence from N^+ to N, and the inverse p^{-1} of **N** is called the **successor function** s of **N**. **Peano's Postulates** for **N** may then be stated thus:

1. 0 is in N.
2. If n is in N, so is $s(n)$.
3. 0 has no predecessor.
4. If $s(m) = s(n)$, then $m = n$.
5. Any subset of N which contains) and contains $s(x)$ whenever it contains x, is equal to N.

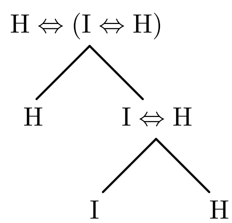
The trees we actually use are, for the most part **ordered trees**, in which the order, from left to right, of the successors of any junction is also given. For instance, the two ordered trees displayed below are not the same, even though they look the same and have the same nodes, because the order of the successors of the origin o in each of them is the reverse of the other[10].

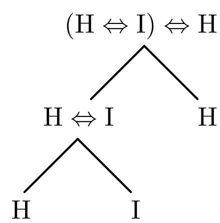
The trees we will discuss first are **formation trees** and **subformula trees**. Consider the biconditionals ' $H \Leftrightarrow (I \Leftrightarrow H)$ ' and ' $(H \Leftrightarrow I) \Leftrightarrow H$ '. The left immediate subformula of the first one is 'H', and its right immediate subformula is ' $I \Leftrightarrow H$ ', while the left immediate subformula of the second one is ' $H \Leftrightarrow I$ ', and its right immediate subformula is ' H '.

The left immediate subformula of ' $I \Leftrightarrow H$ ' is ' I ', and its right immediate subformula is ' H '. The left immediate subformula of ' $H \Leftrightarrow I$ ' is ' H ', and its right immediate subformula is ' I '. We start the subformula trees of the two biconditional formulas with the formulas themselves. Then we put their two left and right immediate subformulas directly below them on the left and right, respectively. We then repeat the process one more time. The result is a subformula tree for ' $(H \Leftrightarrow I) \Leftrightarrow I$ ' displayed on the left, and a subformula tree

[10]From now on, we leave out arrowheads from the arrows in tree diagrams, and leave out any line between two nodes, unless at least one of them is a junction.

for ‘ $(H \Leftrightarrow I) \Leftrightarrow I$ ’ displayed on the right, with the successors of each node in the tree displayed immediately below it. We leave out arrow heads:

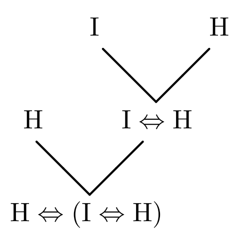


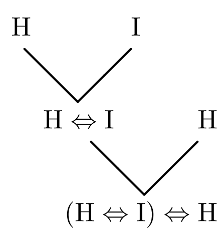


Now let’s provide *formation trees* for these two biconditionals, starting with the one on the left. By the formation rule (iii)(page 47), since ‘ I ’ and ‘ H ’ are formulas (by rule (i)), so is ‘ $I \Leftrightarrow H$ ’ *in that order*, with ‘ I ’ on the left and ‘ H ’ on the right of ‘ $I \Leftrightarrow H$ ’. Then since ‘ H ’ and ‘ $I \Leftrightarrow H$ ’ are both formulas, so is ‘ $H \Leftrightarrow (I \Leftrightarrow H)$ ’ by (iii) again, with ‘ H ’ on the left and ‘ I ’ on the right.

As for the right biconditional, we note that by (iii), since ‘ H ’ and ‘ I ’ are both formulas (by rule (i)), so is ‘ $H \Leftrightarrow I$ ’ with ‘ H ’ on the left side and ‘ I ’ on the right. Then since ‘ H ’ is also a formula, so is ‘ $(H \Leftrightarrow I) \Leftrightarrow H$ ’, with ‘ $H \Leftrightarrow I$ ’ on the left and ‘ H ’ on the right of the biconditional.

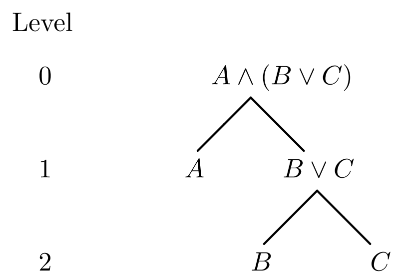
So we have a **formation trees** for ‘ $H \Leftrightarrow (I \Leftrightarrow H)$ ’ displayed on the left, and one for ‘ $(H \Leftrightarrow I) \Leftrightarrow H$ ’ displayed on the right:



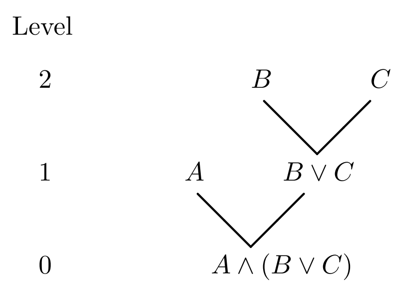


The formation tree for ‘ $H \Leftrightarrow (I \Leftrightarrow H)$ ’ is the same *tree* as subformula tree, and the same is true of the its subformula and formation trees for ‘ $(H \Leftrightarrow I) \Leftrightarrow H$ ’, even though they are constructed by different rules and displayed by different diagrams. But you can turn the tree diagram for the subformula tree for ‘ $H \Leftrightarrow (I \Leftrightarrow H)$ ’ into the diagram for its formation tree and *vice versa* just by turning it upside down and looking at it in the mirror, and the same holds for ‘ $(H \Leftrightarrow I) \Leftrightarrow H$ ’.

Of course this only works if the formula you chose to construct its formation and subformula tree and formation tree looks the same when written out, if you turn it upside-down and look at it in the mirror. Otherwise we need a more general argument. Let the **level** of a node d in a tree be the number of arrow in the path to which d belongs, leading from d to the origin o. So if you take the subformula tree for ‘ $A \wedge (B \vee C)$ ’ which we display along with the level of each formula, with the origin on top at level 0,



and then reverse the order of the levels, with the origin at the bottom, you will get the tree diagram for the formation tree. Each formula will be at the same level as before, and the left and right immediate subformulas of a formula at level n will still be its left and right successors at level $n + 1$:



2.5.4 The Unique Readability Theorem and the Uniqueness of Substitution*

The *unique readability theorem* says that every formula has one and only one formation tree, which therefore displays its structure uniquely. As a result, if the truth values assigned to the atomic formulas at the top of the formation tree for a formula X are given, the truth value of X is uniquely determined.

To prove the unique readability theorem, we must show that there is an algorithm which invariably finds the main connective of any truth functional compound X, along with its immediate subformula, if the main connective is the unary connective '$-$', or its immediate left and right subformulas, if the main connective is a binary connective. With this information in hand, the construction of the subformula for X is routine, and it is clear that this subformula tree, which is also its formation tree, is unique.

To this end, we need some new definitions involving expressions. We say that the expressions E **begins** EF and F **ends** EF, and that every expression both begins and ends itself. If s is some symbol, we say that s **follows** E in the expression G `iff` G is EsF or Es. And finally, X is the **shortest** expression

*This part may be skipped without loss of continuity.

that begins E and has a given property `iff` X begins every expression which begins E and has that property.

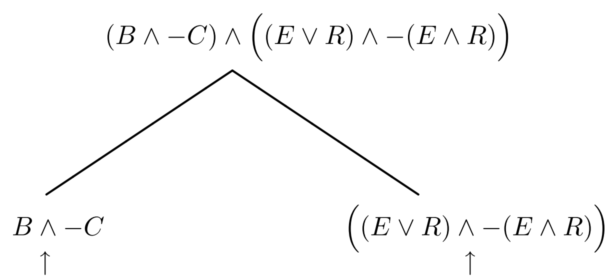
Now for the algorithm. If X is a formula which does not begin with a left parenthesis, then either it is atomic, or it has the form $-Y$. If it is atomic, it has no connectives and no immediate subformulas. Otherwise, the initial negation sign is its main connective, and Y is its immediate subformula.

If X begins with a left parenthesis, and so has the form (G), let E be the shortest expression which begins G, contains no left parentheses and ends with an atomic formula, or begins G and has as many right as left parentheses. Then the symbol that follows E will be a binary connective $\star$, so that $G = E \star F$ and E and F will be the left and right immediate subformulas of X.

As an example, consider again the formula `U` (page 33), where we obey the convention of omitting outer parentheses from formulas, for ease in constructing the subformula tree. At each step in what follows, we shall point to the main connective with an arrow $\uparrow$. Consider,

$$\begin{array}{ccc} (B \wedge -C) & \wedge & \Big((E \vee R) \wedge -(E \wedge R)\Big) \\ & \uparrow & \end{array} \tag{2.22}$$

Here ' $B \wedge -C$ ' and ' $(E \vee R) \wedge -(E \wedge R)$ ' are the left and right immediate subformulas of `U` respectively, so we have,



Here we have identified the main connectives of '$B \wedge -C$' and '$(E \vee R) \wedge -(E \wedge R)$'. We continue until each formula at an end of the tree is atomic. We then have a subformula tree for U [formula (2.5), page 34]. The complete tree is displayed in Figure 2.2 (page 54).

This tree is the one and only one subformula tree for U. Therefore, when the truth value of each Boolean value in U given, the truth value of each formula in the formation tree for U is uniquely determined. For instance if $B = 1, C = 0, E = 1$ and $R = 0$, we found by calculation in eq. (2.7) (page 34) that $U = 1$. The calculation can be depicted for the following **valuation tree** for U (Figure 2.3, page 55), in which the truth value of any formula in the formation tree is calculated from the truth values of the formula or pair of formulas immediately below it.

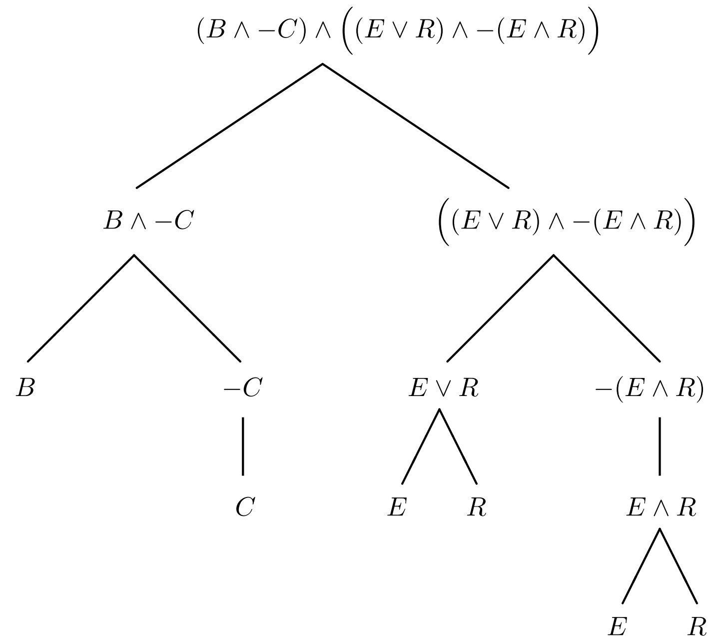


Figure 2.2: Subformula Tree for U

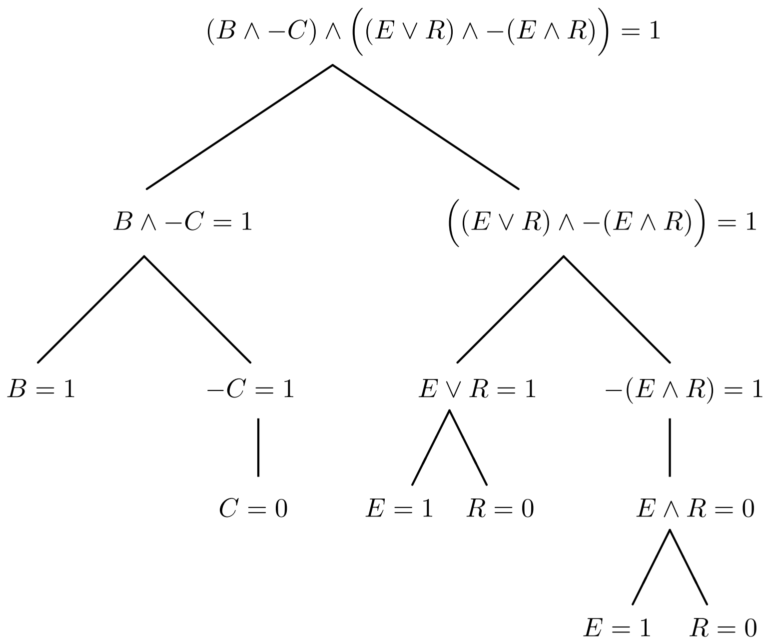


Figure 2.3: Valuation Tree for U

PROBLEMS

1. Provide a subformula tree for the formula,

$$\big(A \Rightarrow (B \Rightarrow C)\big) \Rightarrow \big((A \Rightarrow B) \Rightarrow C\big) \ .$$

2. Provide a valuation tree for the formula in the previous question, under the valuation τ for which,

$$\tau(\text{‘}\, A\,\text{’}) = \tau(\text{‘}\, B\,\text{’}) = \tau(\text{‘}\, C\,\text{’}) = 0 \ .$$

3. Identify the main connective of the formula,

$$\Big(A \Rightarrow ((A \Rightarrow A) \Rightarrow A)\Big) \Rightarrow \Big(\big(A \Rightarrow (A \Rightarrow A)\big) \Rightarrow (A \Rightarrow A)\Big) \ .$$

4. Show that every formula of a Boolean language has as many right as left parentheses.

2.6 Game-Theoretical Semantics*

Overview

Game-theoretical methods for determining truth were first introduced by Hintikka (see for example Hintikka (1979)[37]), opening a new field of study in logic, and may provide more insight into philosophical analysis than does Boolean logic, or Boolean first order logic. A given formula is defined to be true `iff` I have a winning strategy in a two person game between myself and "Nature", played on a tree derived from a valuation tree.

2.6.1 Boolean Games

Calculating truth values of formulas is easier than doing arithmetic, because there are only two numbers, 0 and 1 in the calculation. But instead of a direct calculation to find the truth value of a formula, we can also make a game of it, which is fun, and besides it has interesting applications in logic. To see how this works, we consider some examples.

EXAMPLES

Given that $A = 1$, $B = 2$ and $C = 0$, find the truth values of the formulas:

Example	Formula X	Term	$X[1/A, 1/B, 0/C]$
1.	$A \vee (B \wedge C)$	(i)	$1 \vee (1 \wedge 0)$
2.	$(A \vee B) \wedge C$	(ii)	$(1 \vee 1) \wedge 0$
3.	$(A \vee B) \wedge -(A \wedge B)$	(iii)	$(1 \vee 1) \wedge -(1 \wedge 1)$
4.	$(A \vee C) \wedge -(B \wedge C)$	(iv)	$(1 \vee 0) \wedge -(1 \wedge 0)$

Of course we can find the answers in the usual way by calculating the value of $X[1/A, 1/B, 0/C]$ in each case. Instead, for each formula X in examples 1 - 4 below, we associate a **game tree** with each of the Boolean terms $X[1/A, 1/B, 0/C]$ in (i) - (iv) above. A game tree for a formula X for a given valuation τ is a subformula tree for X, in which every non-atomic subformula has been replaced by its main connective, while every atomic subformula at each end has been replaced by the Boolean constant that denotes its truth value. Alternatively, it is the subterm tree for the term that results from X by substituting for each Boolean variable in X the constant denoting its given truth value, and then deleting all but the main connective of each non-atomic subterm.

Example 1. Let's start with the term (i) which we associate a game tree of Figure 2.4. The game is between you and Nature. It starts by putting a coin at the top of the tree. Get out a coin.

You win and Nature loses if the coin ends up on a 1. Otherwise, you lose and Nature wins. These games have two rules for moving the coin, which can be moved from one position or point in the tree to another directly below it:

*This part may be skipped without loss of continuity.

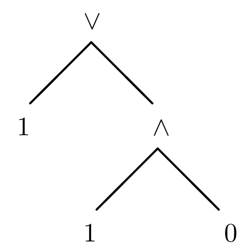


Figure 2.4: Game Tree for $1 \vee (1 \wedge 0)$ (Example 1)

R1: `If the coin is on ∧,` **Nature** `chooses whether to move it down one place to the left or right.`

R2: `If the coin is on ∨,` **you** `choose whether to move it down one place to the left or right.`

Eventually the coin will land on a 1 or 0 at the bottom of the tree, ending the game. If it lands on a 1, you win and Nature loses. If it lands on 0, you loose and Nature wins.

You have a **winning strategy** if no matter what moves Nature chooses, you can move the coin so as to win the game. The formula is true for the given assignment `iff` you have a winning strategy. (Of course you can lose the game even if you have a winning strategy, if you make a wrong move).

For the game of Example 1 above, you have a winning strategy: you have the first move, and you move the coin down to the left to win. (Of course, if you move the coin down to the right, and then Nature moves the coin down to the right again, you lose).

Example 2. Now lets try the game tree for the term (ii), diagrammed in Figure 2.5 below. Here you don't have a winning strategy, because it's Nature's first move, and she can choose to move the coin to the right and win.

Next we consider games, the game tree of which contains negation signs. Whenever the coin is on a negation sign, you and Nature exchange roles. If the coin is then immediately above an end, it is moved either to a ' 0 ' or a ' 1 '. If you are playing Nature and the coin lands on a ' 0 ' then you as Nature have won, while if it lands on a ' 1 ', you lose. The reverse applies if you are playing yourself. If the coin is immediately above an ' ∧ ', while you are playing Nature, it's your turn, while if it's directly above an ' ∨ ', it's Nature's turn. The reverse holds if you're playing yourself (and Nature is playing herself). If the coin is immediately above a negation sign, it is moved down to it, and you and Nature again switch roles.

The eventual outcome is this: when the coin lands on a negation sign, and there is no negation sign immediately below it, and you have just switched rules

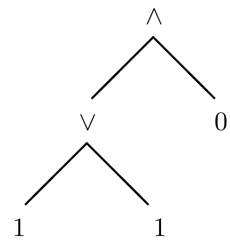


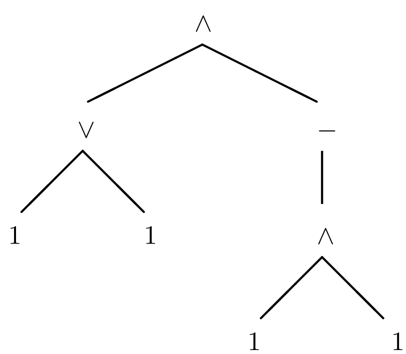
Figure 2.5: Game Tree for $(1 \vee 1) \wedge 0$ (Example 2)

an odd number of consecutive times, then you and Nature now play the roles opposite to the roles you and Nature assumed before the coin landed on the first negation sign. If you have just switched roles an even number of consecutive times, you revert to the role you were playing before the coin landed on the first negation sign. Of course if the coin is on the top of the tree and is on a negation sign, you play Nature and she plays you.

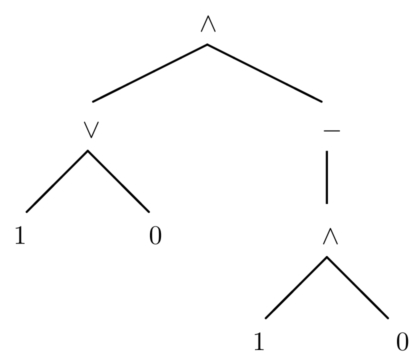
You have a **winning strategy** if you can win no matter what moves Nature makes, **whoever is playing her role**.

R3: `If the coin is on −, you and Nature assume roles opposite to what each of you had in the previous play. (If there is no previous play, you play Nature and she plays you).`

Examples 3. and 4. Finally lets consider terms (iii) and (iv), in which negation does enter in:



Example 3: Game Tree for $(1 \vee 1) \wedge -(1 \wedge 1)$



Example 4: Game Tree for $(1 \vee 0) \wedge -(1 \wedge 0)$

In Example 3, you have no winning strategy, for Nature has the first move. If she moves to the right, and lands on −, you play Nature, and the coin moves to ‘ $\wedge$ ’ directly below. Then no matter what move you (as Nature) choose, the coin lands on a 1, so Nature loses (so *you* loose, in your role as Nature). But in Example 4, if the coin is on the lower ‘ $\wedge$ ’ directly below the negation sign, you as Nature can move to the right to win (because Nature wins by landing on a ‘ 0 ’). And if on the first move, Nature goes to the left and lands on ‘ $\vee$ ’, then it’s your move, so you can move to the left and win.

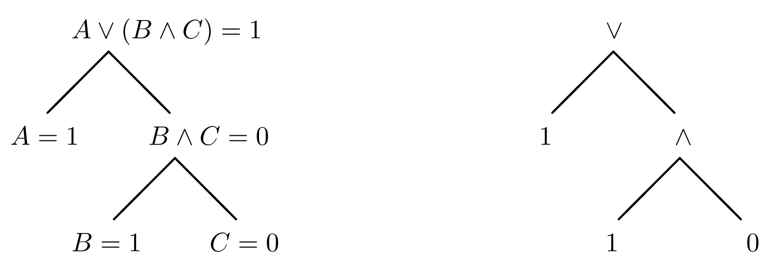
2.6.2 How to Win a Boolean Game when you have a Winning Strategy

Why is a formula X true when a truth value assignment σ is given, for instance when as in this case, $\sigma(‘ A ’) = 1$, $\sigma(‘ B ’) = 1$ and $\sigma(‘ C ’) = 0$ `iff` you have a winning strategy for the game for $X[1/A, 1/B, 0/C]$?

Well, to each game tree for $X[1/A, 1/B, 0/C]$ there corresponds a unique τ-valuation tree $\mathbf{T}_{\tau(X)}$ for X. In fact we can get the game tree for $X[1/A, 1/B, 0/C]$ from the valuation tree for X by replacing every equation in it with a non-atomic formula on the left by its main connective, and every equation with a Boolean variable on the left (it will be the bottom of the tree) its assigned truth value on the right hand side.

Example 1.

First consider games in which the negation sign does not occur in the game tree. So let’s take the valuation tree and the corresponding game tree for term (i),



Valuation Tree Game Tree

You are in a **safe position** in the game, if the coin is on a point in the game tree which corresponds to an equation of the form $Y = 1$ in the valuation tree. Otherwise, you are in an **unsafe position**.

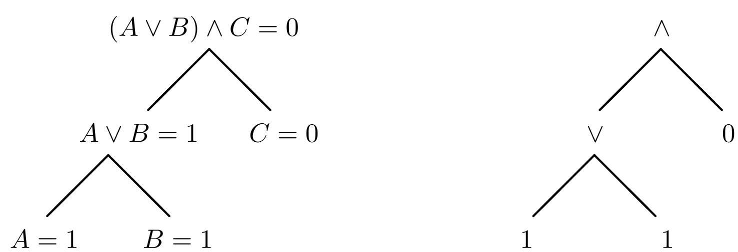
If you are on a safe position, you can always move to a safe position if it’s your move, while if it’s Nature’s move, she has no choice but to move to another safe position. So, if the game starts out with the coin in a safe position, you can always keep it in a safe position, and win.

Why? Because if you start out on a safe position and it’s Nature’s move, you must be on a ‘ $\wedge$ ’, so both positions under you are safe, and Nature has no choice but to move to another safe position. If on the other hand you are on a safe ' $\vee$ ' position, at least one of the positions immediately below you is safe, and since it's your move, you can go to a safe position. So if the coin starts at the top of the tree on a safe position, you can always move it to a safe position when it's your turn, and Nature has no choice but to move to a safe position when it's her turn, so that you eventually win.

In this example, at the start of the game you are in a safe position, and *therefore* you have a winning strategy: at the start of the game, it's your turn, so there is a safe position below you which you move to and in this game you land on a safe endpoint and win.

Example 2.

Now consider term (ii), with this valuation tree and corresponding game tree:



Valuation Tree Game Tree

Here the tables are turned: at the start of the game, you are on an unsafe position. When you're in an unsafe position and its your turn, both positions immediately below you are unsafe, so you have no choice but to move to an unsafe position. If it's Nature's turn, at least one of the positions immediately below her is unsafe, so she can move the coin to another unsafe position. In this case, it's her turn, and she can move down to the right to a terminal unsafe position and win. In short, the game started out with the coin on an unsafe position, *therefore* you have no winning strategy.

Examples 3. and 4.

Now let's consider what happens when you and Nature are in opposite roles. What we've said still goes. If you were in a safe spot before you switched roles, then you moved the coin through an odd number of negation signs, you'd land on an unsafe position for Nature, who is now assuming *your* role, but *safe* for you, in your role of Nature.

So you, as Nature, can keep it on a false formula and win. On the other hand, if you're on a true formula while you're playing Nature, then Nature, as you, can keep it on a false formula and win, so you lose.

So a safe formula is now a **true subformula** of X in the subformula tree for X, provided at that point in the corresponding game tree, you're playing yourself, while it is a **false subformula** if at that point you're playing Nature. We present the valuation trees for examples 3 and 4, together with their corresponding game trees:

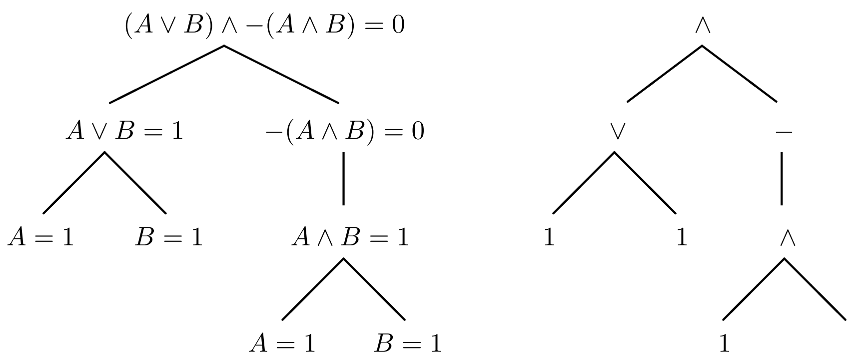


Figure 2.6: Example 3 Valuation and Game Trees

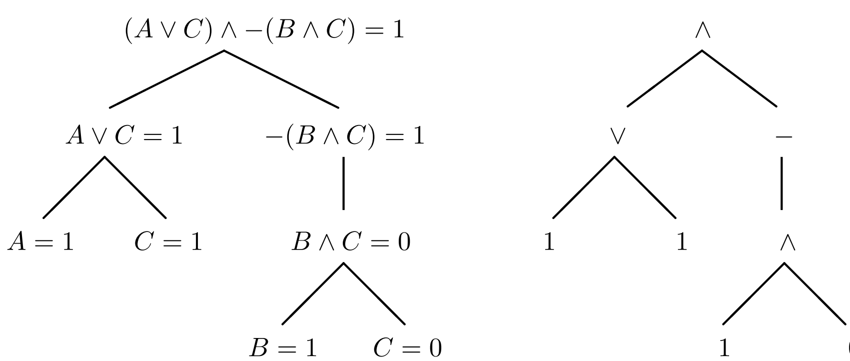


Figure 2.7: Example 4 Valuation and Game Trees

In both cases, it's Nature's choice first. You have no winning strategy in the first case (Example 3) since Nature can move you to an unsafe position in your role as Nature, and keep you in an unsafe position to lose, but in Example 4, if she moves you to '$-$', the coin goes to a safe position (for you, in your new role), so you can as Nature move it to a safe position (for you as Nature) to win.

PROBLEMS

1. Let $\sigma(`A\,') = \sigma(`B\,') = \sigma(`C\,') = 0$.
   (a) Under the valuation τ which extends σ, provide a game tree for,
   $$\big(A \Rightarrow (B \Rightarrow C)\big) \Rightarrow \big((A \Rightarrow B) \Rightarrow C\big) \ .$$
   (b) Determine whether you have a winning strategy for this game, and explain your answer.

2.7 How to Believe that you will Believe a Falsehood

2.7.1 On the Island of Knights and Knaves*

In some remote stretch of the Pacific Ocean is the Island of Knights and Knaves, discovered only recently by that intrepid explorer, anthropologist and logician, the renowned Raymond Smullyan (as reported in Smullyan (1985)[85] or (1987)[86]).

Every inhabitant of the island is either a knight or a knave (but not both). The most well known distinction between the knights and knaves of this island is that the knights *always* tell the truth, while the knaves *always* lie. Yet there is nothing about a native's appearance or demeanor that can tell you whether she (or he) is a knight or a knave. To summarize, the *Rules of the Island* are:

A `Every inhabitant of the island is either a knight or a knave (but not both).`

B `Knights always tell the truth.`

C `Knaves always lie`[11]`.`

Of course, no native of the Island of Knights and Knaves can claim to be a knave–all will claim that they are knights. But if you ask a native whether two times two is four, you can tell by her answer whether she is a knight or a knave. Or, if you ask the following question, you can also tell: "if I were to ask you if you are a knight, would you answer 'yes' or 'no"'? Can you see why?

Now suppose you come across two natives, and you ask which is a knight and which is a knave, but one of them merely replies "we are both knaves". A

*This part may be skipped without loss of continuity.

[11] On the other hand, somewhere in the vast Atlantic, there is an island called Parris Island. The rules of this island are:

1. Every inhabitant of the island who is recognized as human is either an officer or an enlisted man (but not both).
2. Officers always tell the truth.
3. Rule for enlisted men : never tell an officer the truth.
4. Rule for officers : never believe an enlisted man.

knight cannot say that, so the speaker must be a knave. The other native must then be a knight, lest the knave be telling the truth.

A more systematic way of solving such problems is this: if a native asserts $[X]$, and if $[K]$ is true `iff` she is a knight, then $[K \Leftrightarrow X]$ must be true, i.e.the formula

$$K \Leftrightarrow X \ , \tag{2.23}$$

must be true. Reason: if the speaker is a knight and asserts $[X]$, then $[X]$ must true (because the speaker said so, and knights always tell the truth), while if $[X]$ is true, then the speaker must be a knight (because the speaker said so, and knaves never tell the truth).

So, for example, if one of two natives says "we are both knaves", and $[K]$ is true `iff` the speaker is a knight, while $[L]$ is true `iff` the other native is a knight, then

$$K \Leftrightarrow -K \wedge -L \ , \tag{2.24}$$

is true, and is equivalent to $-K \wedge L$. (Check it out!)

But suppose that the speaker says instead "you will never believe that I am a knight." What should **you** believe? (See the end of section 8.5.4, page 161 on Gödel's discoveries for clues).

PROBLEMS

1. Explain how you can tell, by a native's answer to your question,

   > "If I asked you if you are a knight, would you answer 'yes'?"

   whether that native is a knight or a knave.

2. A visitor to the Island of Knights and Knaves addresses two natives of the island, and asks which one is a knight and which one is a knave. One of them replies "we are not both knights". Is the one who replies a knight or a knave? Is the other one a knight or a knave? Explain your answer. (Do not answer "yes" to either question. We want the status of each native).

3. A theology student, who believes all logical consequences of anything he believes, and also believes everything his professor tells him, asks the professor whether God exists. His professor replies "God exists if and only if you will never believe that God exists". Show that the student believes that he will sooner or later believe a falsehood.

4. During a realistic combat exercise (in which the officers never carry evidence of rank, because in combat, officers cannot afford to be recognized as such), an inhabitant of Parris Island, who is recognized as human, is taken prisoner. The interrogator, whom the prisoner recognizes as an officer, seeks to determine whether the prisoner is an officer. But the prisoner will only say:

   > "You will never believe that I am an officer."

Assume that the rules of Parris Island hold (see footnote 11, page 62). What is the interrogator to believe? Can he ever believe the whole truth?