CHAPTER

ONE

INTRODUCTION

1.1 Mathematical Proof

Although there is written evidence of mathematical activity in Egypt as early as 3000 BC, many scholars locate the birth of mathematics proper in ancient Greece around the sixth century BC, when deductive proof was first introduced. Aristotle credited Thales of Miletus with recognizing the importance of not just what we know but how we know it, and finding grounds for knowledge in the deductive method. Around 300 BC, Euclid codified a deductive approach to geometry in his treatise, the *Elements*. Through the centuries, Euclid's axiomatic style was held as a paradigm of rigorous argumentation, not just in mathematics, but in philosophy and the sciences as well.

Here is an example of an ordinary proof, in contemporary mathematical language. It establishes a fact that was known to the Pythagoreans.

Theorem. $\sqrt{2}$ is irrational, which is to say, it cannot be expressed as a fraction a/b, where a and b are integers.

Proof. Suppose $\sqrt{2} = a/b$ for some pair of integers a and b. By removing any common factors, we can assume a/b is in lowest terms, so that a and b have no factor in common. Then we have $a = \sqrt{2}b$, and squaring both sides, we have $a^2 = 2b^2$.

The last equation implies that a^2 is even, and since the square of an odd number is odd, a itself must be even as well. We therefore have $a = 2c$ for some integer c. Substituting this into the equation $a^2 = 2b^2$, we have $4c^2 = 2b^2$, and hence $2c^2 = b^2$. This means that b^2 is even, and so b is even as well.

The fact that a and b are both even contradicts the fact that a and b have no common factor. So the original assumption that $\sqrt{2} = a/b$ is false.

In the next example, we focus on the natural numbers,

$$\mathbb{N} = \{0, 1, 2, \ldots\}.$$

A natural number n greater than or equal to 2 is said to be *composite* if it can be written as a product $n = m \cdot k$ where neither m nor k is equal to 1, and *prime* otherwise. Notice that if $n = m \cdot k$ witnesses the fact that n is composite, then m and k are both smaller than n. Notice also that, by convention, 0 and 1 are considered neither prime nor composite.

Theorem. Every natural number greater than or equal to 2 can be written as a product of primes.

Proof. We proceed by induction on n. Let n be any natural number greater than 2. If n is prime, we are done; we can consider n itself as a product with one term. Otherwise, n is composite, and we can write $n = m \cdot k$ where m and k are smaller than n and greater than 1. By the inductive hypothesis, each of m and k can be written as a product of primes, say $m = p_1 \cdot p_2 \cdot \ldots \cdot p_u$ and $k = q_1 \cdot q_2 \cdot \ldots \cdot q_v$. But then we have

$$n = m \cdot k = p_1 \cdot p_2 \cdot \ldots \cdot p_u \cdot q_1 \cdot q_2 \cdot \ldots \cdot q_v,$$

a product of primes, as required.

Later, we will see that more is true: every natural number greater than 2 can be written as a product of primes in a unique way, a fact known as the *fundamental theorem of arithmetic*.

The first goal of this course is to teach you to write clear, readable mathematical proofs. We will do this by considering a number of examples, but also by taking a reflective point of view: we will carefully study the components of mathematical language and the structure of mathematical proofs, in order to gain a better understanding of how they work.

1.2 Symbolic Logic

Toward understanding how proofs work, it will be helpful to study a subject known as "symbolic logic," which provides an idealized model of mathematical language and proof. In the *Prior Analytics*, the ancient Greek philosopher set out to analyze patterns of reasoning, and developed the theory of the *syllogism*. Here is one instance of a syllogism:

Every man is an animal.

Every animal is mortal.

Therefore every man is mortal.

Aristotle observed that the correctness of this inference has nothing to do with the truth or falsity of the individual statements, but, rather, the general pattern:

Every A is B.

Every B is C.

Therefore every A is C.

We can substitute various properties for A, B, and C; try substituting the properties of being a fish, being a unicorn, being a swimming creature, being a mythical creature, etc. The various statements that result may come out true or false, but all the instantiations will have the following crucial feature: if the two hypotheses come out true, then the conclusion comes out true as well. We express this by saying that the inference is *valid*.

Although the patterns of language addressed by Aristotle's theory of reasoning are limited, we have him to thank for a crucial insight: we can classify valid patterns of inference by their logical form, while abstracting away specific content. It is this fundamental observation that underlies the entire field of symbolic logic.

In the seventeenth century, Leibniz proposed the design of a *characteristica universalis*, a universal symbolic language in which one would express any assertion in a precise way, and a *calculus ratiocinatur*, a "calculus of thought" which would express the precise rules of reasoning. Leibniz himself took some steps to develop such a language and calculus, but much greater strides were made in the nineteenth century, through the work of Boole, Frege, Peirce, Schroeder, and others. Early in the twentieth century, these efforts blossomed into the field of mathematical logic.

If you consider the examples of proofs in the last section, you will notice that some terms and rules of inference are specific to the subject matter at hand, having to do with numbers and the properties of being prime, composite, even, odd, and so on. But there are other terms and rules of inference that are not domain specific, such as those related to the words "every," "some," "and," and "if … then." The goal of symbolic logic is to identify these core elements of reasoning and argumentation and explain how they work, as well as to explain how more domain-specific notions are introduced and used.

To that end, we will introduce symbols for key logical notions, including the following:

- $A \to B$, "if A then B"
- $A \wedge B$, "A and B"
- $A \vee B$, "A or B"
- $\neg A$, "not A"
- $\forall x\ A$, "for every x, A"
- $\exists x\ A$, "for some x, A"

We will then provide a formal proof system that will let us establish, deductively, that certain entailments between such statements are valid.

The proof system we will use is a version of *natural deduction*, a type of proof system introduced by Gerhard Gentzen in the 1930s to model informal styles of argument. In this system, the fundamental unit of judgment is the assertion that a statement, A, follows from a finite set of hypotheses, Γ. This is written as $\Gamma \vdash A$. If Γ and Δ are two finite sets of hypotheses, we will write Γ, Δ for the *union* of these two sets, that is, the set consisting of all the hypotheses in each. With these conventions, the rule for the conjunction symbol can be expressed as follows:

$$\dfrac{\Gamma \vdash A \qquad \Delta \vdash B}{\Gamma, \Delta \vdash A \wedge B}$$

This should be interpreted as saying: assuming A follows from the hypotheses Γ, and B follows from the hypotheses Δ, $A \wedge B$ follows from the hypotheses in both Γ and Δ.

We will see that one can write such proofs more compactly leaving the hypotheses implicit, so that the rule above is expressed as follows:

$$\dfrac{A \qquad B}{A \wedge B}$$

In this format, a snippet of the first proof in the previous section might be rendered as follows:

$$\dfrac{\dfrac{\dfrac{\dfrac{}{\neg \mathit{even}(b)} \qquad \dfrac{\forall x\ (\neg \mathit{even}(x) \to \neg \mathit{even}(x^2))}{\neg \mathit{even}(b) \to \neg \mathit{even}(b^2))}}{\neg \mathit{even}(b^2)} \qquad\qquad \mathit{even}(b^2)}{\bot}}{\mathit{even}(b)}$$

The complexity of such proofs can quickly grow out of hand, and complete proofs of even elementary mathematical facts can become quite long. Such systems are not designed for writing serious mathematics. Rather, they provide idealized models of mathematical inference, and insofar as they capture something of the structure of an informal proof, they enable us to study the properties of mathematical reasoning.

The second goal of this course is to help you understand natural deduction, as an example of a formal deductive system.

1.3 Interactive Theorem Proving

Early work in mathematical logic aimed to show that ordinary mathematical arguments could be modeled in symbolic calculi, at least in principle. As noted above, complexity issues limit the range of what can be accomplished in practice; even elementary mathematical arguments require long derivations that are hard to write and hard to read, and do little to promote understanding of the underlying mathematics.

Since the end of the twentieth century, however, the advent of computational proof assistants has begun to make complete formalization feasible. Working interactively with theorem proving software, users can construct formal derivations of complex theorems that can be stored and checked by computer. Automated methods can be used to fill in small gaps by hand, verify long calculations axiomatically, or fill in long chains of inferences deterministically. The reach of automation is currently fairly limited, however. The strategy used in interactive theorem proving is to ask users to provide just enough information for the system to be able to construct and check a formal derivation. This typically involves writing proofs in a sort of “programming language” that is designed with that purpose in mind. For example, here is a short proof in the *Lean* theorem prover:

```
section
variables (P Q : Prop)

theorem my_theorem : P ∧ Q → Q ∧ P :=
assume h : P ∧ Q,
have P, from and.left h,
have Q, from and.right h,
show Q ∧ P, from and.intro ‹Q› ‹P›

end
```

If you are reading the present text in online form, you will find a button above the formal “proof script” that says “try it!” Pressing the button opens the proof in an editor window and runs a version of Lean inside your browser to process the proof, turn it into an axiomatic derivation, and verify its correctness. You can experiment by varying the text in the editor; any errors will be noted in the window to the right.

Proofs in Lean can access a library of prior mathematical results, all verified down to axiomatic foundations. A goal of the field of interactive theorem proving is to reach the point where any contemporary theorem can be verified in this way. For example, here is a formal proof that the square root of two is irrational, following the model of the informal proof presented above:

```
import data.nat.prime
open nat

theorem sqrt_two_irrational {a b : ℕ} (co : gcd a b = 1) :
  a^2 ≠ 2 * b^2 :=
assume h : a^2 = 2 * b^2,
have 2 ∣ a^2,
  by simp [h],
have 2 ∣ a,
  from prime.dvd_of_dvd_pow prime_two this,
exists.elim this $
assume (c : nat) (aeq : a = 2 * c),
have 2 * (2 * c^2) = 2 * b^2,
  by simp [eq.symm h, aeq];
    simp [pow_succ', mul_comm, mul_assoc, mul_left_comm],
have 2 * c^2 = b^2,
  from mul_left_cancel' dec_trivial this,
have 2 ∣ b^2,
  by simp [eq.symm this],
```

(continues on next page)

(continued from previous page)

```
have 2 ∣ b,
  from prime.dvd_of_dvd_pow prime_two this,
have 2 ∣ gcd a b,
  from dvd_gcd ‹2 ∣ a› ‹2 ∣ b›,
have 2 ∣ (1 : ℕ),
  by simp * at *,
show false, from absurd ‹2 ∣ 1› dec_trivial
```

The third goal of this course is to teach you to write elementary proofs in Lean. The facts that we will ask you to prove in Lean will be more elementary than the informal proofs we will ask you to write, but our intent is that formal proofs will model and clarify the informal proof strategies we will teach you.

1.4 The Semantic Point of View

As we have presented the subject here, the goal of symbolic logic is to specify a language and rules of inference that enable us to get at the truth in a reliable way. The idea is that the symbols we choose denote objects and concepts that have a fixed meaning, and the rules of inference we adopt enable us to draw true conclusions from true hypotheses.

One can adopt another view of logic, however, as a system where some symbols have a fixed meaning, such as the symbols for "and," "or," and "not," and others have a meaning that is taken to vary. For example, the expression $P \wedge (Q \vee R)$, read "P and either Q or R," may be true or false *depending on the basic assertions that* P, Q, *and* R *stand for*. More precisely, the truth of the compound expression depends only on whether the component symbols denote expressions that are true or false. For example, if P, Q, and R stand for "seven is prime," "seven is even," and "seven is odd," respectively, then the expression is true. If we replace "seven" by "six," the statement is false. More generally, the expression comes out true whenever P is true and at least one of Q and R is true, and false otherwise.

From this perspective, logic is not so much a language for asserting truth, but a language for describing possible states of affairs. In other words, logic provides a specification language, with expressions that can be true or false depending on how we interpret the symbols that are allowed to vary. For example, if we fix the meaning of the basic predicates, the statement "there is a red block between two blue blocks" may be true or false of a given "world" of blocks, and we can take the expression to describe the set of worlds in which it is true. Such a view of logic is important in computer science, where we use logical expressions to select entries from a database matching certain criteria, to specify properties of hardware and software systems, or to assert constraints that we would like a constraint solver to satisfy.

There are important connections between the syntactic / deductive point of view on the one hand, and the semantic / model-theoretic point of view on the other. We will explore some of these along the way. For example, we will see that it is possible to view the "valid" assertions as those that are true under all possible interpretations of the non-fixed symbols, and the "valid" inferences as those that maintain truth in all possible states and affairs. From this point of view, a deductive system should only allow us to derive valid assertions and entailments, a property known as *soundness*. If a deductive system is strong enough to allow us to verify *all* valid assertions and entailments, it is said to be *complete*.

The fourth goal of this course is to convey the semantic view of logic, and to lead you to understand how logical expressions can be used to specify states of affairs.

1.5 Goals Summarized

To summarize, these are the goals of this course:

- You should learn to write clear, "literate," mathematical proofs.
- You should become comfortable with symbolic logic and the formal modeling of deductive proof.
- You should learn how to use an interactive proof assistant.
- You should understand how to use logic as a precise language for making claims about systems of objects and the relationships between them, and specifying certain states of affairs.

Let us take a moment to consider the relationship between some of these goals. It is important not to confuse the first three. We are dealing with three kinds of mathematical language: ordinary mathematical language, the symbolic representations of mathematical logic, and computational implementations in interactive proof assistants. These are very different things!

Symbolic logic is not meant to replace ordinary mathematical language, and you should not use symbols like $\wedge$ and $\vee$ in ordinary mathematical proofs any more than you would use them in place of the words "and" and "or" in letters home to your parents. Natural languages provide nuances of expression that can convey levels of meaning and understanding that go beyond pattern matching to verify correctness. At the same time, modeling mathematical language with symbolic expressions provides a level of precision that makes it possible to turn mathematical language itself into an object of study. Each has its place, and we hope to get you to appreciate the value of each without confusing the two.

The proof languages used by interactive theorem provers lie somewhere between the two extremes. On the one hand, they have to be specified with enough precision for a computer to process them and act appropriately; on the other hand, they aim to capture some of the higher-level nuances and features of informal language in a way that enables us to write more complex arguments and proofs. Rooted in symbolic logic and designed with ordinary mathematical language in mind, they aim to bridge the gap between the two.

This book also aims to show you how mathematics is built up from fundamental concepts. Logic provides the rules of the game, and then we work our way up from properties of sets, relations, functions, and the natural numbers to elementary number theory, combinatorics, and properties of the real numbers. The last chapter rounds out the story with a discussion of axiomatic foundations.

1.6 About this Textbook

Both this online textbook and the *Lean* theorem prover are new and ongoing projects. You can learn more about Lean from its project page, the Leann community pages, and the online textbook, Theorem Proving in Lean.

We are grateful for feedback and corrections from a number of people, including Bruno Cuconato, William DeMeo, Tobias Grosser, Lyle Kopnicky, Alexandre Rademaker, Matt Rice, and Jason Siefken.

CHAPTER

TWO

PROPOSITIONAL LOGIC

2.1 A Puzzle

The following puzzle, titled “Malice and Alice,” is from George J. Summers’ *Logical Deduction Puzzles*.

Alice, Alice’s husband, their son, their daughter, and Alice’s brother were involved in a murder. One of the five killed one of the other four. The following facts refer to the five people mentioned:

1. A man and a woman were together in a bar at the time of the murder.
2. The victim and the killer were together on a beach at the time of the murder.
3. One of Alice’s two children was alone at the time of the murder.
4. Alice and her husband were not together at the time of the murder.
5. The victim’s twin was not the killer.
6. The killer was younger than the victim.

Which one of the five was the victim?

Take some time to try to work out a solution. (You should assume that the victim’s twin is one of the five people mentioned.) Summers’ book offers the following hint: “First find the locations of two pairs of people at the time of the murder, and then determine who the killer and the victim were so that no condition is contradicted.”

2.2 A Solution

If you have worked on the puzzle, you may have noticed a few things. First, it is helpful to draw a diagram, and to be systematic about searching for an answer. The number of characters, locations, and attributes is finite, so that there are only finitely many possible “states of affairs” that need to be considered. The numbers are also small enough so that systematic search through all the possibilities, though tedious, will eventually get you to the right answer. This is a special feature of logic puzzles like this; you would not expect to show, for example, that every even number greater than two can be written as a sum of primes by running through all the possibilities.

Another thing that you may have noticed is that the question seems to presuppose that there is a unique answer to the question, which is to say, over all the states of affairs that meet the list of conditions, there is only one person who can possibly be the killer. *A priori*, without that assumption, there is a difference between finding *some* person who could have been the victim and showing that that person *had* to be the victim. In other words, there is a difference between exhibiting some state of affairs that meets the criteria and demonstrating conclusively that no other solution is possible.

The published solution in the book not only produces a state of affairs that meets the criterion, but at the same time proves that this is the only one that does so. It is quoted below, in full.

From (1), (2), and (3), the roles of the five people were as follows: Man and Woman in the bar, Killer and Victim on the beach, and Child alone.

Then, from (4), either Alice's husband was in the bar and Alice was on the beach, or Alice was in the bar and Alice's husband was on the beach.

If Alice's husband was in the bar, the woman he was with was his daughter, the child who was alone was his son, and Alice and her brother were on the beach. Then either Alice or her brother was the victim; so the other was the killer. But, from (5), the victim had a twin, and this twin was innocent. Since Alice and her brother could only be twins to each other, this situation is impossible. Therefore Alice's husband was not in the bar.

So Alice was in the bar. If Alice was in the bar, she was with her brother or her son.

If Alice was with her brother, her husband was on the beach with one of the two children. From (5), the victim could not be her husband, because none of the others could be his twin; so the killer was her husband and the victim was the child he was with. But this situation is impossible, because it contradicts (6). Therefore, Alice was not with her brother in the bar.

So Alice was with her son in the bar. Then the child who was alone was her daughter. Therefore, Alice's husband was with Alice's brother on the beach. From previous reasoning, the victim could not be Alice's husband. But the victim could be Alice's brother because Alice could be his twin.

So *Alice's brother was the victim* and Alice's husband was the killer.

This argument relies on some "extralogical" elements, for example, that a father cannot be younger than his child, and that a parent and his or her child cannot be twins. But the argument also involves a number of common logical terms and associated patterns of inference. In the next section, we will focus on some of the key logical terms occurring in the argument above, words like "and," "or," "not," and "if … then."

Our goal is to give an account of the patterns of inference that govern the use of those terms. To that end, using the methods of symbolic logic, we will introduce variables A, B, C, … to stand for fundamental statements, or *propositions*, and symbols $\wedge$, $\vee$, $\neg$, and $\to$ to stand for "and," "or," "not," and "if … then … ," respectively. Doing so will let us focus on the way that compound statements are built up from basic ones using the logical terms, while abstracting away from the specific content. We will also adopt a stylized notation for representing inferences as *rules*: the inscription

$$\frac{A \qquad B}{C}$$

indicates that statement C is a *logical consequence* of A and B.

2.3 Rules of Inference

2.3.1 Implication

The first pattern of inference we will discuss, involving the "if … then …" construct, can be hard to discern. Its use is largely implicit in the solution above. The inference in the fourth paragraph, spelled out in greater detail, runs as follows:

If Alice was in the bar, Alice was with her brother or her son.

Alice was in the bar.

Alice was with her brother or son.

This rule is sometimes known as *modus ponens*, or “implication elimination,” since it tells us how to use an implication in an argument. As a rule, it is expressed as follows:

$$\frac{A \to B \qquad A}{B}\ {\to}\mathrm{E}$$

Read this as saying that if you have a proof of $A \to B$, possibly from some hypotheses, and a proof of A, possibly from hypotheses, then combining these yields a proof of B, from the hypotheses in both subproofs.

The rule for deriving an “if … then” statement is more subtle. Consider the beginning of the third paragraph, which argues that if Alice’s husband was in the bar, then Alice or her brother was the victim. Abstracting away some of the details, the argument has the following form:

Suppose Alice’s husband was in the bar.

Then …

Then …

Then Alice or her brother was the victim.

Thus, if Alice’s husband was in the bar, then Alice or her brother was the victim.

This is a form of *hypothetical reasoning*. On the supposition that A holds, we argue that B holds as well. If we are successful, we have shown that A implies B, without supposing A. In other words, the temporary assumption that A holds is “canceled” by making it explicit in the conclusion.

$$\begin{array}{c} \overline{A}^{\;1} \\ \vdots \\ \dfrac{B}{A \to B}\ 1\ {\to}\mathrm{I} \end{array}$$

The hypothesis is given the label 1; when the introduction rule is applied, the label 1 indicates the relevant hypothesis. The line over the hypothesis indicates that the assumption has been “canceled” by the introduction rule.

2.3.2 Conjunction

As was the case for implication, other logical connectives are generally characterized by their *introduction* and *elimination* rules. An introduction rule shows how to establish a claim involving the connective, while an elimination rule shows how to use such a statement that contains the connective to derive others.

Let us consider, for example, the case of conjunction, that is, the word “and.” Informally, we establish a conjunction by establishing each conjunct. For example, informally we might argue:

Alice’s brother was the victim.

Alice’s husband was the killer.

Therefore Alice’s brother was the victim and Alice’s husband was the killer.

The inference seems almost too obvious to state explicitly, since the word “and” simply combines the two assertions into one. Informal proofs often downplay the distinction. In symbolic logic, the rule reads as follows:

$$\frac{A \qquad B}{A \wedge B}\ \wedge\mathrm{I}$$

The two elimination rules allow us to extract the two components:

Alice's husband was in the bar and Alice was on the beach.

So Alice's husband was in the bar.

Or:

Alice's husband was in the bar and Alice was on the beach.

So Alice was on the beach.

In symbols, these patterns are rendered as follows:

$$\frac{A \wedge B}{A}\ \wedge\mathrm{E_l} \qquad \frac{A \wedge B}{B}\ \wedge\mathrm{E_r}$$

Here the l and r stand for "left" and "right".

2.3.3 Negation and Falsity

In logical terms, showing "not A" amounts to showing that A leads to a contradiction. For example:

Suppose Alice's husband was in the bar.

…

This situation is impossible.

Therefore Alice's husband was not in the bar.

This is another form of hypothetical reasoning, similar to that used in establishing an "if … then" statement: we temporarily assume A, show that leads to a contradiction, and conclude that "not A" holds. In symbols, the rule reads as follows:

$$\begin{array}{c} \overline{A}^{\,1} \\ \vdots \\ \dfrac{\bot}{\neg A}\ 1\ \ \neg\mathrm{I} \end{array}$$

The elimination rule is dual to these. It expresses that if we have both "A" and "not A," then we have a contradiction. This pattern is illustrated in the informal argument below, which is implicit in the fourth paragraph of the solution to "Malice and Alice."

The killer was Alice's husband and the victim was the child he was with.

So the killer was not younger than his victim.

But according to (6), the killer was younger than his victim.