I Proofs

Introduction

This text explains how to use mathematical models and methods to analyze problems that arise in computer science. Proofs play a central role in this work because the authors share a belief with most mathematicians that proofs are essential for genuine understanding. Proofs also play a growing role in computer science; they are used to certify that software and hardware will *always* behave correctly, something that no amount of testing can do.

Simply put, a proof is a method of establishing truth. Like beauty, "truth" sometimes depends on the eye of the beholder, and it should not be surprising that what constitutes a proof differs among fields. For example, in the judicial system, *legal* truth is decided by a jury based on the allowable evidence presented at trial. In the business world, *authoritative* truth is specified by a trusted person or organization, or maybe just your boss. In fields such as physics or biology, *scientific* truth[1] is confirmed by experiment. In statistics, *probable* truth is established by statistical analysis of sample data.

Philosophical proof involves careful exposition and persuasion typically based on a series of small, plausible arguments. The best example begins with "Cogito ergo sum," a Latin sentence that translates as "I think, therefore I am." This phrase comes from the beginning of a 17th century essay by the mathematician/philosopher, René Descartes, and it is one of the most famous quotes in the world: do a web search for it, and you will be flooded with hits.

Deducing your existence from the fact that you're thinking about your existence is a pretty cool and persuasive-sounding idea. However, with just a few more lines

[1] Actually, only scientific *falsehood* can be demonstrated by an experiment—when the experiment fails to behave as predicted. But no amount of experiment can confirm that the *next* experiment won't fail. For this reason, scientists rarely speak of truth, but rather of *theories* that accurately predict past, and anticipated future, experiments.

of argument in this vein, Descartes goes on to conclude that there is an infinitely beneficent God. Whether or not you believe in an infinitely beneficent God, you'll probably agree that any very short "proof" of God's infinite beneficence is bound to be far-fetched. So even in masterful hands, this approach is not reliable.

Mathematics has its own specific notion of "proof."

Definition. A *mathematical proof* of a *proposition* is a chain of *logical deductions* leading to the proposition from a base set of *axioms*.

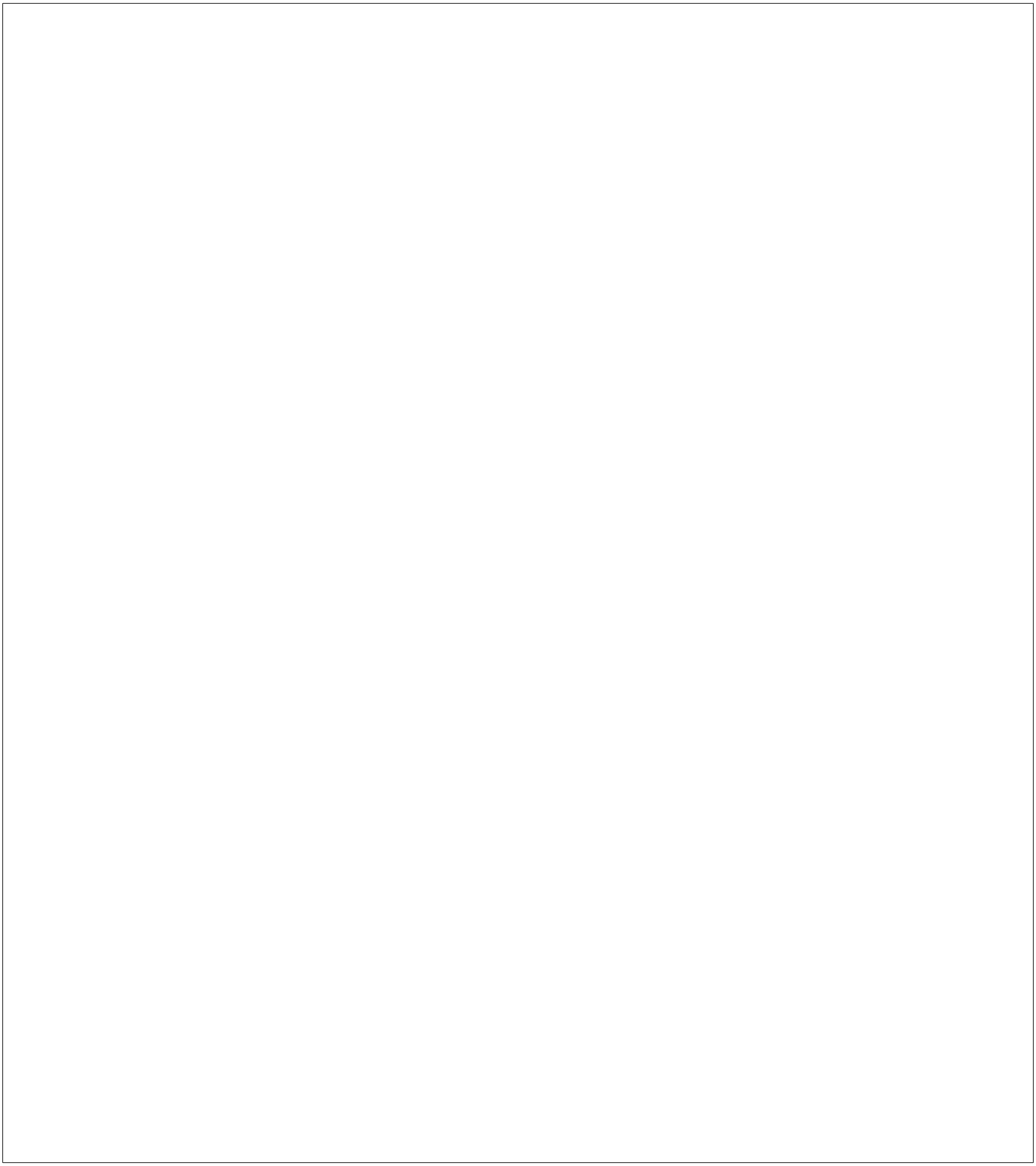
The three key ideas in this definition are highlighted: *proposition*, *logical deduction*, and *axiom*. Chapter **??** examines these three ideas along with some basic ways of organizing proofs. Chapter **??** introduces the Well Ordering Principle, a basic method of proof; later, Chapter **??** introduces the closely related proof method of induction.

If you're going to prove a proposition, you'd better have a precise understanding of what the proposition means. To avoid ambiguity and uncertain definitions in ordinary language, mathematicians use language very precisely, and they often express propositions using logical formulas; these are the subject of Chapter **??**.

The first three Chapters assume the reader is familiar with a few mathematical concepts like sets and functions. Chapters **??** and **??** offer a more careful look at such mathematical data types, examining in particular properties and methods for proving things about infinite sets. Chapter **??** goes on to examine recursively defined data types.

0.1 References

[**?**], [**?**], [**?**]



Index