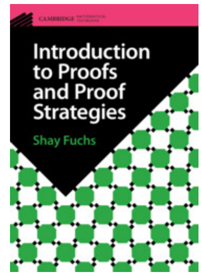
Review of [1]

Introduction to Proofs and Proof Strategies
Shay Fuchs
Cambridge University Press, 2023
USD 44.99, Paperback, 349 pages



Review by

David J. Littleboy
(djl@alum.mit.edu)
Technical translator, retired
Tokyo, Japan

1 Overview

This book is an undergraduate mathematics textbook that could (and should!) be used in undergraduate computer science curricula. It's that good.

First, the background. Mathematics education, from secondary through pre-calculus and through calculus itself is mainly concerned with solving problems by algebraic manipulations and is only secondarily concerned with foundational issues and formal proofs. Mathematics educators concerned with undergraduate pure math education worry that students who enjoy and are good at those sort of problems may have trouble dealing with foundational issues and reading and writing proofs. One approach to this is Lara Alcock's *How to Study for a Mathematics Degree* [1], which explains to the student reader what life will be like as an undergraduate mathematics students, and includes an extensive bibliography of work in this area as well as recommendations for further reading. While excellent, it prepares the student more psychologically than mathematically. Fuchs, on the other hand, prepares undergraduate students to handle the sorts of proofs they will be faced with in upper-level mathematics courses. As such, it also can serve that purpose for undergraduate computer science majors, who will be required to exhibit far greater mathematical sophistication than this reviewer (MS, Comp. Sci. '84) was.

More recent undergraduate computer science texts such as Lehman [4] and Gossett [2] do cover this material, although not as thoroughly. At the back of my mind is the concern that presenting this material as part of a larger book/course may implicitly deprecate it somewhat and fail to emphasize that this material applies to all of mathematics (including real and complex analysis) as well as the mathematics used in computer science. So it seems to me that dedicating a term to this material makes sense.

There are several reasons I like this book, but the main one is that it includes a large number of problems that drill the student thoroughly on the material covered. In addition, it is laser-focused on the necessary material while avoiding even mentioning the unnecessary. It may seem surprising that an undergraduate text that prepares students for number theory, abstract algebra, topology and the like does so without even mentioning those fields, but the main body of this book does exactly that. (To get ahead of myself, Part II of this book shows how the proof techniques presented are used in combinatorics, real analysis, complex numbers, and linear algebra. And includes a fair number of problems in these areas.) The tight focus and avoidance of extraneous material allows the author to keep this book to an extremely svelte 342 pages, while the main section itself is a mere 204 pages. In this day and age of the enormous tome textbook, this is a real pleasure.

2 What's in the Book?

This book consists of two parts, the first being the main material and the second the additional topics mentioned above to show the use of the material presented. The organization of the material presented in the first part is particularly well thought out. For example, one might think that in presenting the use of sets in mathematical proofs, relations would come before functions. But the author presents relations as the last chapter in Part I. This allows him to include longer discussions of equivalence relations and how modular arithmetic defines equivalence classes in that chapter.

Chapter 1 introduces proofs using the pre-calculus/high-school algebra potential math and computer science majors will be comfortable with: the quadratic formula, inequalities, various types of means. But now the student must read and write proofs of results. The text includes in-line exercises related to the material being presented, and provides solutions to all of these exercises at the end of the chapter. Both the exercises and the end-of-chapter problems range from simple to quite difficult. Chapter 2 introduces sets, functions, and the field axioms, although it puts off introducing injections, bijections and the like for another chapter. Here, the field axioms are introduced independently, that is, not as part of abstract algebra (i.e., groups, rings, fields). This strikes me as a good idea, since the point is to introduce the use of axioms in proofs.

The short Chapter 3 "Informal Logic and Proof Strategies" is the core of this book. The basic idea here is that proofs can be classified into one of three types: direct proof, proof by contrapositive, and proof by contradiction (leaving proof by induction for a later chapter). While [4] also lists proof by enumeration as a proof type, this book deals with that as a special case occurring in certain situations, since each resulting case is an independent proof. As an older reader, and one trained in AI of the 1970s and 1980s, I had hoped for more here, that is for a classification of the sorts of manipulations that get one from the claim to the result, e.g., completing the square, parity arguments (e.g., dominoes on a chess board). (Along these lines, Fuchs, however, explicitly notes where the induction step is used in inductive proofs.) But given the purpose of and main audience for this book, I think the author has the right idea: for students learning how to write proofs for the first time, this is the classification they need to know.

Fuchs, however, implicitly rejects the idea of classifying proofs further, arguing that beyond the major categories, proof is an art, not a science. He writes: "Is there also an algorithm for generating proofs? Unfortunately, no. In fact, I would rather say: Fortunately, no!". He goes on to argue that it is this creative and artistic nature of mathematics that makes it so attractive.

Although this book was developed and written before the explosion of generative AI, the argument for mathematics as art, in my opinion, does speak to the excessive enthusiasm of the generative AI researchers who believe that LLMs will be able to do and contribute to higher mathematics.

This chapter also exhibits the laser focus that I like about the book: the introduction to logic as used in proofs only introduces the necessary and key ideas and techniques. It's not an extensive, wide-ranging overview of the field of logic in general; it's the parts of logic you absolutely need to have an operational understanding of, and exercises are provided to make sure that you do acquire that operational understanding.

Aside: Despite its laser focus, this book goes into lovely detail on two problems whose treatment in other sources have irritated me. The first is the standard proof of the irrationality of the square root of 2. A common way of expressing this proof depends on evenness, which is because it's the square root of 2. This is not incorrect, but it doesn't work for the square root of 3 (or anything else for that matter). But Fuchs presents a more general proof and has several related problems. A computer science text I reviewed a while ago claimed that if N is the product of the first k primes, then $N + 1$ will be prime, which is incorrect, since it could be composite. Fuchs not only does not make this mistake, he goes into this in more detail, again, in the problems. It really is the problems that make this book.

The next four chapters cover mathematical induction, bijections and cardinality, integers and divisibility, and relations. As always, each chapter has text-related exercises with solutions, and a good number of end-of-chapter problems drilling the material. (Aside: In the margin next to a formula that appeared in the text I wrote "Prove me", since it wasn't immediately obvious. Proving that formula was one of the in-text exercises a few paragraphs down.)

I claim, perhaps unreasonably, that this text should be used in your computer science curriculum. The zeroth point is, of course, that other than using quite a few examples and problems involving pre-calculus algebra (since the author does have real analysis in his sights), the material is pretty much exactly what the student will need to do upper-level undergraduate computer science courses. And, again, in this day and age, being extremely comfortable with abstract algebra and being ready to handle arguments in number theory and linear algebra and other advanced mathematical areas will be necessary to deal with cryptography, AI (that is, SIMD algorithms), and quantum computing.

But my main point is that you are going to have to teach this material anyway, so doing it in an independent course would make sense. And this is a good way of doing that.

Are there better ways? Perhaps. MIT offers a math course for computer science majors that covers some of this material plus a lot more, and the textbook [4] is (or at least was) available as a free, 900-page, pdf download. The first 282 pages cover some of the material in this book with, of course, computer science material included. But there are fewer problems, and they don't include drill sorts of problems. A student who had taken a course based on Fuchs would be able to breeze through much of the first part of this text/course and focus on the CS-specific material. (Aside: this freebie textbook is also seriously great. But because it doesn't drill that thoroughly, it's going to be hard for many students.)

3 The Writing

The author, in addition to teaching mathematics at the university level, has credentials and experience in secondary education teaching, and it shows in the writing. It is respectful of the reader yet points out issues that the reader may find difficult or confusing. Unlike another text I recently reviewed, in which there was something that irritated or angered on nearly every page, the writing here doesn't intrude or irritate. That may sound like faint praise, but it's not: the writing is exactly what is needed.

4 Criticisms

Nothing in this world is perfect, and reviews are supposed to be critical, so here goes. First, I'd like there to have been more problems that involve summation and product notation. Upper-level computer science texts such as [3] can be really dense of these, and more practice would be useful were this text to be used in a computer science context. Second, no solutions are provided for the end-of chapter drill problems, only solutions to the example problems used in the text are given. A few solved problems at the end of each chapter would make this book more useful for self study. Finally, the index is a tad sparse. Computer science is mentioned in a couple of places, but doesn't appear in the index. Also, I wanted to look up the problems on proving irrationality, but "irrational" didn't index them.

This book does not include a bibliography, but I don't think it needs to. A listing of the author's favorite mathematics books and textbooks, or a suggested further readings list, would be nice to have, but isn't necessary. (Again, the bibliography and further reading recommendations in Alcock [1] are excellent.)

5 Conclusions

To my eye, ear, and sensibility, this book succeeds in doing what it sets out to do: to teach students the fundamentals of actually doing mathematics at a more advanced and abstract level than they have seen. Its organization of topics works well and I can't find fault with the material covered. (Aside: It is designed for a course taught at the first year undergraduate level, presumably in parallel with first year calculus, and as such, avoids the use of examples from calculus, only mentioning it in two or three places.) The large number of problems means that the students will have extensive experience with these new concepts by the end of a course. Since all of the concepts presented are used in computer science, it could provide the basis for a mathematics for a computer science course as well.

References

[1] Alcock, L. *How to Study for a Mathematics Degree*, Oxford University Press, 2013.

[2] Gossett, E., *Discrete Mathematics with Proof*, Wiley, 2nd. ed., 2009.

[3] Graham, R. L., Knuth, D. E., Patashnik, O. *Concrete Mathematics*, Addison-Wesley, 2nd. ed., 1994.

[4] Lehman, E., Leighton, F.T., Meyer, A. R. *Mathematics for Computer Science*, Creative Commons, 2015.