



Book review

Michael Spivey, *An Introduction to Logic Programming through Prolog*, Prentice Hall International Series in Computer Science, Englewood cliffs, NJ, 1996. ISBN 0-13-536047-1.

If you need to teach Logic Programming to an audience of people who are not very familiar with logic, and, nevertheless, you want to put an appropriate emphasis on the declarative reading of the programs and provide a rigorous logical approach, this is probably a good course book for you to use.

The book begins by introducing a subset of first-order logic: syntax, relations, truth tables and logical variables. Every concept is carefully and at length explained using examples taken from logic programming, and the reader is step-by-step brought to think about logic programs in a purely declarative way. The first operational concepts are introduced only in Chapter 5, and these operational concepts are inference rules and refutation. Unification and the SLD resolution principle are handled only in Chapters 6 and 7. This introductory part provides a very “declarative” and perhaps uncommon approach to the teaching of Logic Programming. Personally, I have found this part didactically sound and very convincing.

In what could be considered the second part of the book, the issue of PROLOG as a programming language is tackled. Here the approach is the standard one we are used to. In this section, it is explained how some prominent programming problems (among them: search, parsing and hardware simulation) can be solved using PROLOG. Some attention is dedicated to the problem of nontermination and to the choice of the search technique. In this part, the reader becomes aware of the most important features of the Logic Programming paradigm. To this end, one full chapter is dedicated to the unfold/fold transformations, which, in my personal view, is useful in that it allows to understand the strength of symbolic manipulation techniques.

From Chapter 14 onwards, the book concentrates on the operational aspects of PROLOG. It does so by describing the key implementation aspects of Pico-PROLOG, a dialect of PROLOG for which the author wrote an interpreter in about 2000 lines of PASCAL code. The carefully commented source code is reported at the end of the book. Despite the fact that from this point on the discussion becomes unavoidably more technical, the author manages to keep the book very readable and enjoyable, focusing on the key aspects, and leaving the technical details to the source code.

* Review copies of books which might be of interest to the readers of *Science of Computer Programming* should be sent to Prof. K. Apt (address: see inside front cover). Proceedings of conferences will not normally be reviewed.

The author is a fine and fluent writer; in my opinion this book is actually so easy to follow that it is quite suitable for self-teaching Logic Programming. At the same time the book manages to be formal and rigorous at tackling the cornerstone results of logic programming.

The book is particularly useful if the audience has knowledge of programming and little knowledge of mathematical logic. All the logical concepts that are needed to understand the logic programming paradigm are introduced and explained in the book itself.

Because of its style, this is definitely a course-book, which is meant to be studied in a consistent way from the beginning to the end. On the other hand, it is not meant to be used as a quick reference guide, and it is probably unsuitable for this latter purpose.

It must be said that in order to achieve a good balance between rigorousness and clarity, the author sometimes makes use of nonstandard notations, which, for people who already have some logic or logic programming background, might be confusing.

Concluding, this is definitely a peculiar book, with an approach to the teaching of logic programming which is novel to me. In my opinion, its strong point lies in the fact that it teaches well and in depth both the declarative and the operational readings of Logic Programming, and that it does so while remaining easy to follow. For the appropriate audience, this can be the suitable book for teaching logic programming.

Sandro Etalle
Department of Computer Science
University of Maastricht
The Netherlands